



TESSELAÇÕES EM GRAFOS E SUAS APLICAÇÕES EM COMPUTAÇÃO QUÂNTICA

Alexandre Santiago de Abreu

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia de Sistemas e Computação.

Orientadores: Franklin de Lima Marquezino
Luis Antonio Brasil Kowada

Rio de Janeiro
Março de 2017

TESSELAÇÕES EM GRAFOS E SUAS APLICAÇÕES EM COMPUTAÇÃO
QUÂNTICA

Alexandre Santiago de Abreu

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

Prof. Franklin de Lima Marquezino, D.Sc.

Prof. Luis Antonio Brasil Kowada, D.Sc.

Prof. Daniel Ratton Figueiredo, Ph.D.

Prof. Mitre Costa Dourado, D.Sc.

Prof. Renato Portugal, D.Sc.

RIO DE JANEIRO, RJ – BRASIL

MARÇO DE 2017

Abreu, Alexandre Santiago de

Tesselações em grafos e suas aplicações em Computação Quântica/Alexandre Santiago de Abreu. – Rio de Janeiro: UFRJ/COPPE, 2017.

X, 48 p.: il.; 29, 7cm.

Orientadores: Franklin de Lima Marquezino

Luis Antonio Brasil Kowada

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia de Sistemas e Computação, 2017.

Referências Bibliográficas: p. 44 – 47.

1. Tesselações em Grafos.
 2. Algoritmo Quântico.
 3. Element Distinctness.
 4. Staggered Quantum Walk Model.
- I. Marquezino, Franklin de Lima *et al.*
II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

Dedico ao soberano Deus.

Agradecimentos

A Deus, por tudo o que Ele fez, faz, e fará em minha vida.

Ao meus pais por todo o apoio, incentivo, e suporte não apenas na minha vida acadêmica, mas durante todo o tempo.

A minha namorada e amiga, Amanda, por toda compreensão, apoio, e incentivo em todos os instantes.

Aos meus orientadores, Franklin e Luis, por todos os conhecimentos e ensinamentos transmitidos ao longo dos anos em que trabalhamos juntos.

Aos professores Renato Portugal (LNCC) e Celina Figueiredo (PESC/UFRJ), por discussões importantíssimas que contribuíram com este trabalho.

Aos amigos, Luis Felipe Ignácio, Alexander Melo, e Daniel Posner, por se disporem a ler manuscritos e dar opiniões muito válidas que contribuíram com o trabalho.

Aos demais amigos do Laboratório de Algoritmos e Combinatória (LAB-AC), por sempre estarem dispostos a respeito dos assuntos abordados neste trabalho.

Aos demais professores que passaram ao longo da minha história, por todo o conhecimento fornecido.

Ao CNPq pelo apoio financeiro.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

TESSELAÇÕES EM GRAFOS E SUAS APLICAÇÕES EM COMPUTAÇÃO QUÂNTICA

Alexandre Santiago de Abreu

Março/2017

Orientadores: Franklin de Lima Marquezino

Luis Antonio Brasil Kowada

Programa: Engenharia de Sistemas e Computação

O uso de passeios aleatórios na computação clássica tem resultado em boas soluções para problemas em diversas áreas. Seu correspondente quântico tem sido uma ferramenta eficiente no desenvolvimento de algoritmos quânticos. Um exemplo de grande importância teórica usando tal artifício é o algoritmo de Ambainis para o problema de *Element k -Distinctness*, que responde se em uma lista de N elementos existem k elementos de mesmo valor. A abordagem de Ambainis reduz o problema a encontrar um vértice marcado em um grafo de Johnson, bipartido, requerendo $O(N^{k/k+1})$ passos e sendo melhor do que qualquer abordagem clássica já desenvolvida. Esse procedimento foi mais tarde generalizado por Szegedy em um novo modelo de passeios quânticos que consiste em executar um passeio através das arestas de um grafo bipartido. Recentemente, Portugal *et al.* desenvolveram o modelo Escalonado, que inclui o modelo de Szegedy como um caso particular. Junto a este modelo, surgiu o conceito de *tesselações em grafos*, importante para a geração dos operadores de evolução unitários para o modelo Escalonado. Portugal ainda mostrou que todos os grafos 2-tesseláveis possuem grafos clique 2-coloríveis. Não é possível afirmar que um grafo cuja cobertura mínima por tesselações $T(G) > 2$ possui um grafo clique cujo número cromático seja igual a $T(G)$. Neste trabalho apresentamos o limite superior para o problema de cobertura mínima por tesselações, além de famílias de grafos cujo número de tesselação é menor ou igual a este limite. Apresentamos também uma reformulação do algoritmo para o problema de *Element k -Distinctness*, utilizando o modelo Escalonado.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

TESSELLATIONS ON GRAPHS AND THEIR APPLICATIONS IN QUANTUM COMPUTING

Alexandre Santiago de Abreu

March/2017

Advisors: Franklin de Lima Marquezino
Luis Antonio Brasil Kowada

Department: Systems Engineering and Computer Science

The usage of random walks in classical computing has resulted in good solutions to problems in many areas. Its quantum counterpart has been an efficient tool in the development of quantum algorithms. An example of great theoretical importance using this approach is Ambainis' algorithm for the Element k -Distinctness problem, which answers if in a list of N elements there are k elements of same value. Ambainis' approach reduces the problem to finding a marked vertex in a bipartite Johnson graph, requiring $O(N^{k/k+1})$ steps, which is better than any known classical approach. This procedure was later generalized by Szegedy in a new model of quantum walks consisting on a walk through the edges of a bipartite graph. Recently, Portugal *et al.* developed the Staggered model, which includes the Szegedy's model as a particular case. The Staggered model introduced the concept of *graph tessellations*, which is important for the definition of the unitary evolution operators. Portugal also proved that all 2-tessellable graphs have a 2-colorable clique graph. It is not possible to state that a graph whose the minimum cover by tessellations $T(G) > 2$ has a clique graph whose chromatic number is equal $T(G)$. In this work we present the upper bound for the problem of tessellation cover, in addition to families of graphs whose tessellation number is less or equal than this bound. We also present a reformulation of the algorithm for the Element k -Distinctness problem using the Staggered model.

Sumário

Lista de Figuras	ix
1 Introdução	1
2 Noções Preliminares	4
2.1 Tesselações em Grafos	4
2.2 Computação Quântica	8
2.3 Passeio Quântico no modelo de Szegedy	10
2.4 Passeio Quântico no modelo Escalonado	14
3 Resultados em Teoria de Grafos	18
3.1 Limite Superior para o Problema de Cobertura Mínima de Grafos por Tesselações	18
3.2 Classes de Grafos com Número de Tesselação Definido	19
3.3 Algumas Conjecturas	21
4 Resultados em Computação Quântica	24
4.1 O problema de <i>Element k-Distinctness</i> e a abordagem de Ambainis .	24
4.2 Algoritmo Quântico para o Problema de <i>Element k-Distinctness</i> no Modelo Escalonado	27
4.2.1 Corretude e Complexidade do Algoritmo de <i>Element k-</i> <i>Distinctness</i>	37
5 Conclusão	42
Referências Bibliográficas	44
Índice Remissivo	48

Lista de Figuras

2.1	Exemplos de tesselações em um mesmo grafo. Cada polígono é uma clique disjunta e a união destes polígonos contém todos os vértices do grafo.	5
2.2	Cobertura formada pela união das três tesselações apresentadas na Figura 2.1. Esta união cobre todo o grafo G e não é possível cobrir todas as arestas com duas tesselações, portanto o número de tesselação deste grafo é 3.	5
2.3	Grafo clique do grafo apresentado pela Figura 2.2. Este grafo é um grafo completo de tamanho 3, visto que todas as 3 cliques do grafo que o originou possuem interseção de vértices não nula uma com as outras.	6
2.4	Grafo A , 2-tesselável, e seu grafo clique $K(A)$, 2-colorível, ou seja, bipartido. Cada cor em $K(A)$ representa uma tesselação em A	7
2.5	Grafo B , conhecido como <i>Hajós Graph</i> , que possui número de tesselação $T(G) = 3$, e seu grafo clique $K(B)$ que é 4-colorível, e que não pode ser colorido com menos que 4 cores. Esta figura exemplifica um caso em que o número de tesselação é inferior ao número cromático do grafo clique.	7
2.6	Esfera de Bloch, usada para representar o estado de um único qubit. Nesta representação, qualquer vetor que parta do centro da esfera e toque a casca da mesma representa um estado quântico para um qubit.	9
2.7	Grafo bipartido onde $S = \{1, 2, 3\}$ e $T = \{1, 2, 3\}$	12
2.8	Exemplo de um grafo de 4 vértices e seu grafo bipartido gerado a partir do processo de duplicação.	14
2.9	Uma tesselação no grafo G com polígonos disjuntos e que cobre todos os nós do grafo, como definido pela Definição 2.1.	15
2.10	Uma tesselação no grafo G , que cobre todos os vértices do grafo e seus polígonos têm interseção vazia entre si.	16
3.1	Grafo roda 6-wheel, que possui número de tesselação igual a 3, e seu grafo clique representado por um K_6 , que é 6-colorível.	20

3.2	Grafo cujo número de tesselação é 3, e seu grafo clique representado por um K_3 , que é 3-colorível.	20
3.3	Exemplos de grafos da família de triângulos de Sierpinski criados a partir de uma geração fractal. Todos possuem número de tesselação igual a 3. Nestas figuras foram representados apenas os polígonos que cobrem cliques maximais do grafo por questões de legibilidade.	23
4.1	Grafo de Johnson bipartido para um conjunto $[N] = \{1, 2, 3, 4\}$ e um valor de $k = 2$. Para tal, temos que a lista de elementos é $ell = \{1, 2, 1, 4\}$	26
4.2	Grafo de Johnson bipartido G , utilizado por Ambainis em sua abordagem, e seu grafo linha $L(G)$, que é exatamente o grafo construído pela Definição 4.1, ambos para $[N] = \{1, 2, 3, 4\}$, onde temos $r = 2$ e $k = 2$	31
4.3	Grafo construído segundo a Definição 4.1 e tesselado segundo a Definição 4.8 para $N = 4$ tal que $[N] = \{1, 2, 3, 4\}$ e $k = 2$	36
4.4	Grafo construído segundo a Definição 4.1 e tesselado segundo a Definição 4.8 após a aplicação do operador U_0 sobre o estado quântico $ \Psi_0\rangle$	37
4.5	Grafo construído segundo as Definição 4.1 e Definição 4.8 após a aplicação do operador U_1 no estado quântico $ \Psi_1\rangle$	37

Capítulo 1

Introdução

A caminhada clássica, ou passeio aleatório (*random walk*), consiste em modelar matematicamente o passeio de uma partícula, que chamamos de caminhante, através de sucessivos passos aleatórios sobre os vértices de um grafo. Esse conceito tem sido muito importante no desenvolvimento de algoritmos na computação clássica, sendo aplicada para solucionar problemas em diversas áreas, como por exemplo Ciência da Computação [22], Física [26, 40], Psicologia [33] e Biologia [14].

Em um passeio clássico em tempo discreto temos um operador estocástico que é aplicado a cada unidade de tempo ao estado do caminhante, fazendo com que a distribuição de probabilidade da posição do caminhante seja alterada a cada passo. De modo geral, temos um operador que atua de modo similar ao lançamento de uma moeda, cujo resultado irá indicar qual direção o caminhante irá tomar.

Feynman [20] foi pioneiro a mostrar que sistemas físicos quânticos poderiam ser usados para implementar um novo paradigma computacional, possivelmente mais eficiente que o modelo clássico. De fato, antes mesmo de os primeiros hardwares quânticos terem sido criados, algoritmos quânticos mais eficientes que seus correspondentes clássicos já haviam sido desenvolvidos. Podemos elencar algoritmos quânticos notáveis, como o algoritmo de Grover [23], que encontra um elemento marcado em uma lista com N elementos em $O(\sqrt{N})$ passos, alcançando um *speed up* quadrático em relação ao tempo requerido pela melhor abordagem clássica. Outro algoritmo importante historicamente é o algoritmo para fatoração de inteiros em números primos apresentado por Shor [42], que soluciona o problema em tempo polinomial, enquanto que na computação clássica não se conhece ainda solução eficiente.

Os conceitos básicos para a compreensão da computação quântica são a álgebra linear, que define toda a base matemática deste paradigma, e os quatro postulados da mecânica quântica, que definem o sistema físico, a evolução temporal do sistema, a composição do sistema e a obtenção de respostas das computações executadas.

Aharonov *et al.* [7] definiram um conceito análogo ao passeio aleatório, conhe-

cido como passeio quântico, sendo amplamente utilizado no desenvolvimento de algoritmos quânticos [10]. Além das aplicações diretamente ao desenvolvimento de algoritmos quânticos, um resultado importante apresentado por Childs *et al.* [17] revela que o passeio quântico é um modelo universal para a computação quântica. Este resultado implica que a implementação de passeios quânticos pode também contribuir para o desenvolvimento de hardwares quânticos em geral. A definição de passeio quântico é similar ao seu correspondente clássico, no entanto, o estado que representa o caminhante, bem como os operadores de evolução do passeio, devem estar embasados nos quatro postulados da mecânica quântica. Em passeios de tempo discreto, um dos modelos mais conhecidos é o passeio quântico com moeda (*coined model*), no qual o operador de evolução divide-se em operador moeda, que indica a direção que será tomada pelo caminhante, e o operador deslocamento, que efetivamente movimenta o caminhante pelo grafo. Historicamente podemos citar ainda o modelo de Szegedy [43], que executa um passeio quântico sem moeda, e mais recentemente o modelo Escalonado (*Staggered Model*) [35, 37, 38], que generaliza todo o modelo de Szegedy e uma grande parte do modelo com moeda. Há indícios de que o modelo Escalonado ainda pode ser mais adequado que os demais para implementações físicas [36].

Devido a sua crescente importância para a computação quântica, o estudo de passeios quânticos tem se tornado alvo de pesquisa desde abordagens mais teóricas e desenvolvimento de algoritmos [6, 8, 9, 11–13, 15, 19, 25, 30, 34, 41] até abordagens que envolvem o desenvolvimento de ferramentas clássicas de simulação [4, 18, 28, 31]. Um exemplo de uma abordagem bem sucedida na utilização do conceito de passeios quânticos é o algoritmo para o problema de *Element k -Distinctness* [8]. Esta abordagem, como será visto ao longo deste trabalho, resolve tal problema em tempo $O(N^{k/k+1})$ (onde k representa o número de repetições de algum elemento da lista) sendo ótimo para o caso em que $k = 2$, como provado por Aaronson & Shi [1].

Neste trabalho, apresentamos uma nova abordagem para o problema de *Element k -distinctness* [2], utilizando o modelo Escalonado. Ainda apresentamos resultados em Teoria de Grafos para o novo parâmetro de grafos que surgiu junto ao modelo Escalonado, conhecido como número de tesselação. Uma tesselação representa uma partição de vértices do grafo em cliques disjuntas, enquanto uma cobertura por tesselações representa um conjunto de tesselações cuja união cobre todas as arestas do grafo. Neste contexto, propomos o limite superior justo para o problema de cobertura mínima por tesselações, que consiste em encontrar o menor número de tesselações possíveis cuja união cubra todas as arestas do grafo. Além disso, apresentamos duas famílias de grafos onde cada uma delas possui o tamanho da cobertura mínima de forma bem definida, sendo que uma delas possui o tamanho de sua cobertura mínima exatamente igual ao limite superior. Apresentamos ainda

algumas conjecturas, sendo duas delas em relação ao número de tesselação para os grafos livres de triângulos e para os grafos da família de triângulos de Sierpinski [27] para qualquer geração fractal finita $g \geq 1$, enquanto a outra conjectura trata da NP-completude do problema.

O trabalho é dividido como segue. No Capítulo 2 apresentamos os conceitos preliminares que são necessários para o entendimento do trabalho, como o conceito de tesselações em grafos e computação quântica. No Capítulo 3 apresentamos os resultados obtidos na área de Teoria de Grafos para o problema de tesselações, como o limite superior do problema, além de duas classes de grafos cujo número de tesselação está bem definido. Também apresentamos conjecturas que estão sendo analisadas nos trabalhos em andamento. No Capítulo 4 apresentamos o algoritmo que foi desenvolvido para o problema de *Element Distinctness* utilizando o modelo Escalonado. Finalmente, no Capítulo 5 apresentamos as considerações finais.

Capítulo 2

Noções Preliminares

2.1 Tesselações em Grafos

O conceito de tesselações em grafos surgiu recentemente junto à apresentação de um novo modelo de passeio quântico, conhecido como modelo Escalonado [35, 37, 38], apresentado na Seção. 2.4.

Uma tesselação em um grafo G é uma família de cliques de G de forma que tais cliques sejam disjuntas e a união das mesmas cubra todos os vértices do grafo. Neste contexto, cada clique disjunta que compõe tal tesselação é chamada de *polígono*.

Definição 2.1 *Seja $C = \{c_1, \dots, c_{\mathcal{T}}\}$ um conjunto de cliques disjuntas duas a duas de G . Se a união das cliques de C cobrir todos os vértices de G , dizemos que C é uma **tesselação** do grafo G . No contexto do modelo Escalonado de passeio quântico, chamamos cada clique de C de **polígono**.*

Como definido acima, uma tesselação cobre todos os vértices do grafo G , no entanto, tal tesselação não necessariamente irá cobrir todas as arestas de G . Neste contexto, dizemos que uma aresta e está coberta por uma tesselação se o par de vértices incidentes a e está dentro de algum polígono presente em tal tesselação.

Definição 2.2 *Uma **cobertura por tesselações** é um conjunto de tesselações cuja união cobre todas as arestas do grafo.*

O conceito de cobertura é usado habitualmente para cobertura de um grafo por vértice, mas como não iremos tratar de cobertura por vértices neste trabalho, para não carregar muito as definições, chamaremos cobertura por tesselações simplesmente como cobertura.

Definição 2.3 *Sejam $t_1, t_2, \dots, t_{\mathcal{T}}$ tesselações de G . Dizemos que G possui seu **número de tesselação** $T(G) = \mathcal{T}$ se e somente se $\{t_1, t_2, \dots, t_{\mathcal{T}}\}$ é um menor conjunto de tesselações em G tal que $t_1 \cup t_2 \cup \dots \cup t_{\mathcal{T}}$ cobre todas as arestas de G .*

Definição 2.4 Dizemos que um grafo G é \mathcal{T} -tesselável, se $T(G) \leq \mathcal{T}$.

Exemplo 2.5 Na Figura 2.1 apresentamos algumas tesselações em um mesmo grafo. As cliques coloridas representam as cliques disjuntas, isto é, os polígonos de suas respectivas tesselações. Pode-se notar que não há interseções entre polígonos de mesma cor e que os mesmos cobrem todos os vértices do grafo em questão.

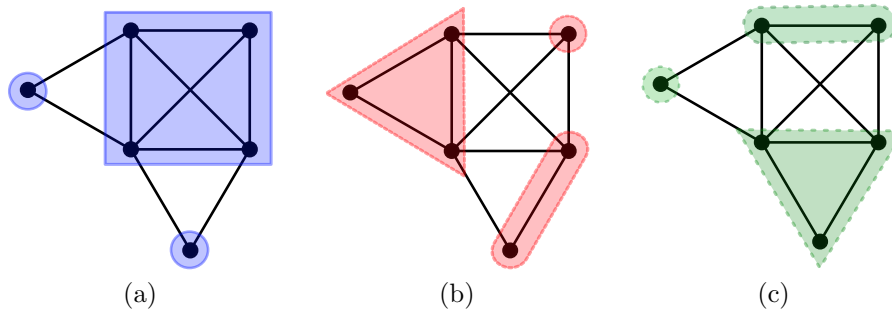


Figura 2.1: Exemplos de tesselações em um mesmo grafo. Cada polígono é uma clique disjunta e a união destes polígonos contém todos os vértices do grafo.

Para cada grafo podemos ter mais de uma tesselação possível, formada por diferentes polígonos, como visto na Figura 2.1.

Exemplo 2.6 Na Figura 2.2 apresentamos a união das tesselações apresentadas na Figura 2.1. Pode-se notar que tal união cobre todo o grafo G . Note que a remoção de qualquer uma das tesselações utilizadas faz com que pelo menos uma aresta de G fique descoberta. Este conjunto de tesselações formam uma cobertura mínima visto que não é possível tesselar este grafo com um número menor de tesselações. Sendo assim, o número de tesselação para este grafo é 3.

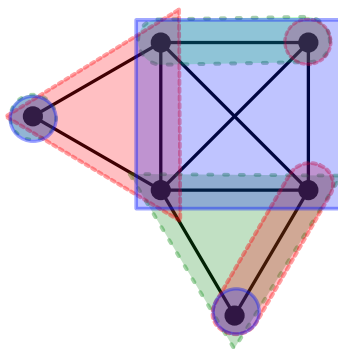


Figura 2.2: Cobertura formada pela união das três tesselações apresentadas na Figura 2.1. Esta união cobre todo o grafo G e não é possível cobrir todas as arestas com duas tesselações, portanto o número de tesselação deste grafo é 3.

Definição 2.7 Uma clique em um grafo G é todo subgrafo H tal que H é completo.

Definição 2.8 *Seja o grafo G com o conjunto de cliques maximais $C = \{c_1, c_2, \dots, c_k\}$. Dizemos que $K(G)$ é o grafo clique de G se existe um mapeamento $U : c_i \rightarrow v_i, \forall c_i \in C$ e $\forall v_i \in V(K(G))$ para $i \in \{1, \dots, k\}$. Existirá a aresta $(v_i, v_j) \in E(K(G))$ se e somente se as cliques c_i e c_j possuírem vértices em comum no grafo G .*

Exemplo 2.9 *O grafo da Figura 2.2 possui o grafo clique representado pela Figura 2.3, que é um grafo completo de tamanho 3, já que o grafo que o gerou possui 3 cliques maximais, todas elas possuindo vértices em comum uma com as outras. Portanto, cada vértice do grafo clique possuirá uma aresta para os demais vértices.*

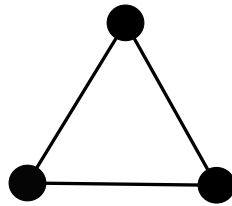


Figura 2.3: Grafo clique do grafo apresentado pela Figura 2.2. Este grafo é um grafo completo de tamanho 3, visto que todas as 3 cliques do grafo que o originou possuem interseção de vértices não nula uma com as outras.

No Capítulo 3 veremos que uma cobertura mínima, i.e., o número de tesselação $T(G)$ de um grafo, está sempre limitada superiormente pelo número cromático¹ de seu grafo clique. Um grafo clique de G , chamado $K(G)$, é um grafo cujos vértices representam cliques maximais de G , enquanto as arestas representam interseções entre tais cliques em G [44].

O conceito de tesselações em grafos está intimamente relacionado com o conceito de coloração de vértices em grafos. Portugal [38] mostrou que todo grafo G 2-tesselável possuiu um grafo clique $K(G)$ que é 2-colorível, ou seja, bipartido. De fato, toda clique maximal do grafo G será representada por um vértice no grafo clique $K(G)$. Desse modo, dois vértices adjacentes no grafo clique indicam que, no grafo original, as cliques representadas por eles possuem interseção entre si, portanto, estando cada uma em tesselações distintas. Pelo fato de que cliques em uma mesma tesselação são disjuntas, todas as cliques maximais presentes em uma mesma tesselação serão representadas, no grafo clique, por vértices que compõem um conjunto independente². Como $K(G)$ é 2-colorível e, portanto, bipartido, cada partição do grafo clique representa cliques maximais —nesse caso, também disjuntas— de uma mesma tesselação em G .

¹Número cromático representa a quantidade de cores mínima para colorir os vértices de algum grafo sem que vértices adjacentes tenham a mesma cor.

²Conjunto independente em um grafo G é um conjunto de vértices não-adjacentes do grafo.

Proposição 2.10 [38] *Um grafo é 2-tesselável se e somente se seu grafo clique é 2-colorível.*

A Proposição 2.10 prova que se um grafo possui seu número de tesselação $T(G) = 2$, então seu grafo clique $K(G)$ é bipartido. Por outro lado, se G possui seu número de tesselação $T(G) = \mathcal{T}$, tal que $\mathcal{T} > 2$, não se pode afirmar que seu grafo clique possui um número cromático $\chi(K(G)) = \mathcal{T}$. A seguir, temos dois exemplos que ilustram o que foi dito acima. Primeiro temos o grafo G que é 2-tesselável cujo grafo clique associado é bipartido. Em seguida, temos o grafo de *Hajós*, que possui número de tesselação $T(G) = 3$, porém seu grafo clique não é 3-colorível.

Exemplo 2.11 *A Figura 2.4 exemplifica um grafo A , 2-tesselável, e seu grafo clique $K(A)$, 2-colorível. Na Figura 2.5 temos um grafo B , conhecido como grafo de *Hajós*. Tal grafo possui um número de tesselação igual a 3, no entanto, seu grafo clique $K(B)$ não é 3-colorível.*

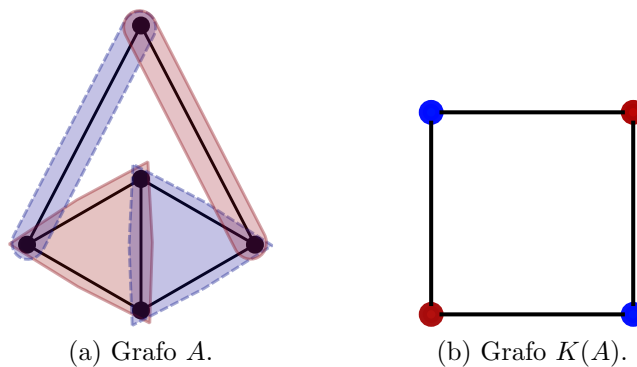


Figura 2.4: Grafo A , 2-tesselável, e seu grafo clique $K(A)$, 2-colorível, ou seja, bipartido. Cada cor em $K(A)$ representa uma tesselação em A .

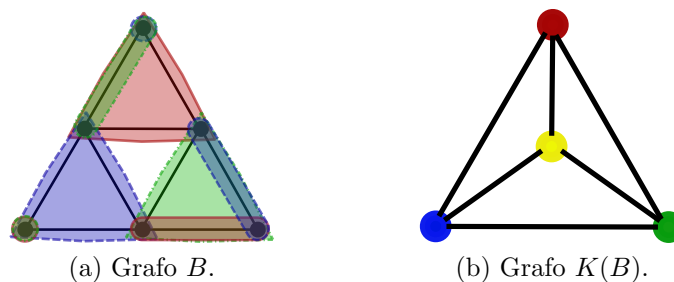


Figura 2.5: Grafo B , conhecido como *Hajós Graph*, que possui número de tesselação $T(G) = 3$, e seu grafo clique $K(B)$ que é 4-colorível, e que não pode ser colorido com menos que 4 cores. Esta figura exemplifica um caso em que o número de tesselação é inferior ao número cromático do grafo clique.

2.2 Computação Quântica

A computação quântica faz uso dos conceitos de Álgebra Linear [29] e da Mecânica Quântica [32], que juntas regem o funcionamento de um sistema quântico. Cada estado da computação é definido por um vetor unitário no espaço vetorial³ complexo de Hilbert \mathcal{H} que evolui após sucessivas aplicações de operadores matriciais que também devem ser unitários.

Antes de introduzirmos de fato os conceitos utilizados na computação quântica, é conveniente que apresentemos a notação de Dirac, que é de uso comum entre os estudiosos da área. Nesta notação, todo vetor coluna \vec{v} é representado por $|v\rangle$, enquanto todo vetor linha \vec{u} é representado por $\langle u|$. Logo, sempre que desejamos escrever um produto escalar, também conhecido como produto interno [29], entre os vetores v e u escrevemos como $\langle v|u\rangle$, ao passo que todo produto vetorial, também conhecido como produto externo, entre dois vetores v e u escrevemos como $|v\rangle\langle u|$.

O estado quântico mais trivial que podemos enunciar é o bit quântico, ou qubit⁴. Tal estado é representado por um vetor unitário no espaço de Hilbert \mathcal{H}^2 tendo representação na base computacional como

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (2.1)$$

e

$$|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \quad (2.2)$$

Uma forma de visualizar um único qubit é através da esfera de raio unitário conhecida como esfera de Bloch, apresentada pela Figura 2.6. Nesta representação temos que qualquer vetor partindo do centro da esfera e tocando a casca da mesma representa um estado quântico possível para um único qubit.

Os quatro postulados da mecânica quântica definem o funcionamento do sistema, bem como as condições para o mesmo. O **primeiro postulado** descreve o espaço de estados do sistema. Todo sistema físico isolado tem associado a ele um espaço vetorial complexo, conhecido como espaço de estados do sistema. Tomando como exemplo o nosso qubit, podemos escrevê-lo de maneira genérica como

$$\alpha|0\rangle + \beta|1\rangle \quad (2.3)$$

onde $\alpha, \beta \in \mathbb{C}$ representam as amplitudes quânticas dos estados $|0\rangle$ e $|1\rangle$, respectivamente. Tais amplitudes devem respeitar a propriedade de $|\alpha|^2 + |\beta|^2 = 1$,

³Um espaço vetorial é um conjunto de vetores onde define-se operações de adição e multiplicação entre tais vetores.

⁴Na literatura também podemos encontrar o termo *q-bit* para nomear o bit quântico.

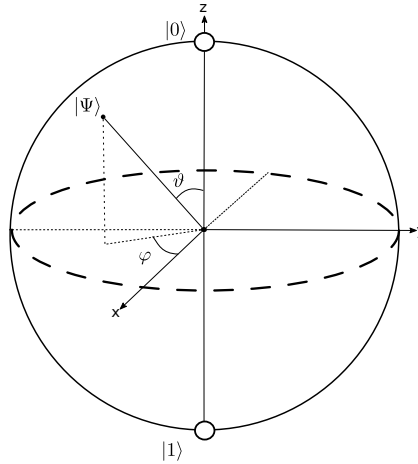


Figura 2.6: Esfera de Bloch, usada para representar o estado de um único qubit. Nesta representação, qualquer vetor que parta do centro da esfera e toque a casca da mesma representa um estado quântico para um qubit.

pois o módulo ao quadrado de cada amplitude representa a probabilidade do estado quântico colapsar para um determinado estado (como veremos mais adiante no quarto postulado).

O **segundo postulado** descreve a evolução do sistema, que é dada através de sucessivas aplicações de operadores unitários em um estado inicial. Tais operadores devem ser unitários, isto é, $UU^\dagger = U^\dagger U = I$, onde I representa a matriz Identidade, dessa forma são reversíveis e determinísticos. Ser reversível significa que dado um estado $|\Psi_1\rangle$, podemos obter o estado imediatamente anterior apenas aplicando o transposto conjugado do operador de evolução, sendo assim temos $|\Psi_0\rangle = U^\dagger|\Psi_1\rangle$. Ser determinístico significa que sabendo o estado atual e o operador de evolução do sistema, somos capazes de descrever estados futuros, sendo assim, temos que $|\Psi_T\rangle = U^T|\Psi_0\rangle$.

O **terceiro postulado** aborda a composição de estados, isto é, quando o sistema quântico possui mais de um qubit em seu estado corrente. Para abordar este assunto precisamos primeiro descrever o produto de Kronecker, ou produto tensorial. Tal produto consiste em multiplicar cada elemento do primeiro fator por cada elemento do segundo fator, como exemplificado na Equação (2.4), sendo assim uma operação

não comutativa.

$$|\psi\theta\rangle = |\psi, \theta\rangle = |\psi\rangle|\theta\rangle = |\psi\rangle \otimes |\theta\rangle = \begin{bmatrix} \psi_1 \\ \vdots \\ \psi_n \end{bmatrix} \otimes \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_m \end{bmatrix} = \begin{bmatrix} \psi_1\theta_1 \\ \vdots \\ \psi_1\theta_m \\ \vdots \\ \psi_n\theta_1 \\ \vdots \\ \psi_n\theta_m \end{bmatrix} \quad (2.4)$$

Estados quânticos compostos por mais de um qubit, mas que não podem ser escritos na forma de um produto de Kronecker de dois ou mais estados (ou seja, não podem ser fatorados), são chamados de estados emaranhados. No paradigma quântico este conceito é importante pois toda ação que ocorre com um dos subsistemas tem efeito direto nos demais. Duas partículas quânticas podem estar a uma distância tão grande quanto se desejar, mas se estiverem emaranhadas toda ação que uma partícula sofrer a outra também sofrerá efeitos da mesma ação. Na Equação (2.5) temos um exemplo de estado emaranhado composto por dois qubits.

$$|\Psi\rangle = \frac{|0, 0\rangle + |1, 1\rangle}{\sqrt{2}} \quad (2.5)$$

O **quarto postulado** descreve o processo de medição. Este processo faz com que o sistema quântico deixe de estar isolado, resultando assim no colapso do estado corrente para algum dos vetores da base. A medição é um processo comum em algoritmos quânticos visto que os algoritmos buscam maximizar a amplitude de algum estado almejado fazendo com que o mesmo tenha alta probabilidade de se tornar o estado colapsado ao final de um processo de medição.

$$|\Psi\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \quad (2.6)$$

Na Equação (2.6), o estado $|\Psi\rangle$ após um processo de medição poderá colapsar para o estado $|0\rangle$ com probabilidade $p_1 = |\frac{1}{\sqrt{2}}|^2 = \frac{1}{2} = 50\%$, assim como a probabilidade de $|\Psi\rangle$ colapsar para o estado $|1\rangle$ é $p_2 = |\frac{1}{\sqrt{2}}|^2 = \frac{1}{2} = 50\%$.

2.3 Passeio Quântico no modelo de Szegedy

O modelo de Szegedy será abordado neste trabalho por motivos didáticos já que seu conceito é importante para o melhor entendimento de um modelo mais geral, conhecido como modelo Escalonado.

Classicamente existe o conceito de passeios aleatórios, ou *random walks*. Este conceito formula matematicamente o deslocamento de uma partícula, chamada de caminhante, através de sucessivos passos aleatórios em um grafo qualquer.

Pode-se descrever tais passos fazendo uso de operadores, por exemplo, como operador moeda e operador deslocamento. O primeiro irá funcionar de fato como o ato de “lançar uma moeda”, condicionando o movimento do caminhante, que é dado pelo operador deslocamento.

No paradigma quântico existe um conceito análogo que chamamos de passeios quânticos, ou *quantum walks* [7]. Assim como seu correspondente clássico, os passeios quânticos modelam o movimento de uma partícula através dos nós de um grafo. A principal diferença entre os passeios citados está no fato de o segundo fazer uso do paralelismo quântico, além do mais, o passeio quântico também está sob as regras definidas pelos postulados da mecânica quântica, sendo assim, os operadores que condicionam este passeio são unitários.

Podemos enunciar alguns modelos de caminhada em tempo discreto tais como o modelo com Moeda (*Coined model*), que é semelhante ao modelo clássico anteriormente descrito, o modelo de Szegedy [43], e o mais recente de todos, o modelo Escalonado (*Staggered model*) [35, 37, 38].

O modelo de passeio quântico de Szegedy é descrito por operadores de reflexão⁵ em um grafo bipartido. Denotamos por S e T as partições de vértices em um grafo bipartido, onde v_s e v_t são vértices genéricos destas partições, respectivamente. Sejam P e Q matrizes estocásticas cujos valores representam as probabilidades de movimentos de S para T e de T para S , respectivamente. Descrevemos suas componentes como

$$\sum_{v_t \in T} p_{st} = 1, \forall v_s \in S$$

e

$$\sum_{v_s \in S} q_{ts} = 1, \forall v_t \in T.$$

É associado a este grafo bipartido um espaço de Hilbert $\mathcal{H}^{n_S \cdot n_T}$, em que $n_S = |S|$ e $n_T = |T|$, cuja base computacional é $b = \{|v_s, v_t\rangle : v_s \in S, v_t \in T\}$. Definimos os operadores $A : \mathcal{H}^{n_S} \rightarrow \mathcal{H}^{n_S \cdot n_T}$ como

$$A = \sum_{v_s \in S} |\phi_s\rangle \langle v_s|,$$

⁵Operador de reflexão é um operador da forma $U = 2(\sum_i |\varphi_i\rangle \langle \varphi_i|) - I$ (ou $U = I - 2(\sum_i |\varphi_i\rangle \langle \varphi_i|)$), que faz uma reflexão em torno dos estados $|\varphi_i\rangle$, sendo I a matriz Identidade.

onde

$$|\phi_s\rangle = |v_s\rangle \otimes \left(\sum_{v_t \in T} \sqrt{p_{st}} |v_t\rangle \right)$$

e $B : \mathcal{H}^{n_T} \rightarrow \mathcal{H}^{n_S \cdot n_T}$ como

$$B = \sum_{v_t \in T} |\psi_t\rangle \langle v_t|,$$

onde

$$|\psi_t\rangle = \left(\sum_{v_s \in S} \sqrt{q_{ts}} |v_s\rangle \right) \otimes |v_t\rangle.$$

O estado quântico $|\phi_s\rangle$ representa a superposição de todas as arestas que são originadas em v_s , enquanto $|\psi_t\rangle$ representa a superposição de todas as arestas originadas em v_t .

Definimos então os operadores de reflexão R_A e R_B , onde R_A reflete um vetor genérico de $\mathcal{H}^{n_S \cdot n_T}$ em torno do subespaço \mathcal{H}_A , que é gerado pelos vetores $|\phi_s\rangle$, e analogamente ocorre com R_B . Estes operadores são definidos como

$$R_A = 2 \sum_{v_s \in S} |\phi_s\rangle \langle \phi_s| - I_{n_S \cdot n_T}$$

e

$$R_B = 2 \sum_{v_t \in T} |\psi_t\rangle \langle \psi_t| - I_{n_S \cdot n_T}$$

e juntos definem o operador de evolução do sistema, que é dado por $U_{P,Q} = R_B R_A$.

A interpretação que se tem em relação aos operadores R_A e R_B é que o primeiro faz uma reflexão de um vetor genérico do espaço $\mathcal{H}^{n_S \cdot n_T}$ em torno do subespaço gerado pelos vetores de $|\phi_s\rangle$, enquanto o segundo faz uma reflexão de um vetor genérico do espaço $\mathcal{H}^{n_S \cdot n_T}$ em torno do subespaço gerado pelos vetores de $|\psi_t\rangle$

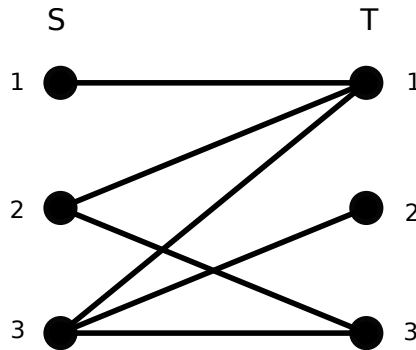


Figura 2.7: Grafo bipartido onde $S = \{1, 2, 3\}$ e $T = \{1, 2, 3\}$.

Exemplo 2.12 Tomando como exemplo o grafo da Figura 2.7, tem-se as matrizes

de probabilidade

$$P = \begin{pmatrix} 1 & 0 & 0 \\ 1/2 & 0 & 1/2 \\ 1/3 & 1/3 & 1/3 \end{pmatrix}$$

e

$$Q = \begin{pmatrix} 1/3 & 1/3 & 1/3 \\ 0 & 0 & 1 \\ 0 & 1/2 & 1/2 \end{pmatrix}.$$

No caso apresentado, temos os estados da base computacional no espaço de Hilbert, que são dados por

$$\{|1, 1\rangle, |1, 2\rangle, |1, 3\rangle, |2, 1\rangle, |2, 2\rangle, |2, 3\rangle, |3, 1\rangle, |3, 2\rangle, |3, 3\rangle\}.$$

Como $|\phi_s\rangle$ é a superposição de todas arestas que se originam a partir de v_s , temos a descrição dos estados que compõem $|\phi_s\rangle$ como

$$|\phi_1\rangle = |1, 1\rangle,$$

$$|\phi_2\rangle = \sqrt{\frac{1}{2}}|2, 1\rangle + \sqrt{\frac{1}{2}}|2, 3\rangle,$$

e

$$|\phi_3\rangle = \sqrt{\frac{1}{3}}|3, 1\rangle + \sqrt{\frac{1}{3}}|3, 2\rangle + \sqrt{\frac{1}{3}}|3, 3\rangle.$$

O mesmo vale para o caso de $|\psi_i\rangle$, que é formado pela superposição dos estados

$$|\psi_1\rangle = \sqrt{\frac{1}{3}}|1, 1\rangle + \sqrt{\frac{1}{3}}|2, 1\rangle + \sqrt{\frac{1}{3}}|3, 1\rangle,$$

$$|\psi_2\rangle = |3, 2\rangle,$$

e

$$|\psi_3\rangle = \sqrt{\frac{1}{2}}|2, 3\rangle + \sqrt{\frac{1}{2}}|3, 3\rangle.$$

A diferença entre a caminhada de Szegedy [43] e a caminhada quântica, introduzida por Aharonov *et al.* [7] está no fato de a segunda utilizar um operador Moeda, cuja ação é similar ao lançamento de uma moeda, como visto no passeio clássico, enquanto a primeira pode ser interpretada como uma caminhada através das arestas do grafo, onde definimos um estado $|a\rangle|b\rangle$ como sendo uma aresta (a, b) , que representa que o caminhante saiu de b e chegou em a , diferentemente da proposta de Aharonov *et al.*, onde os estados representam os vértices em que se encontra o

caminhante.

Através de um processo de duplicação podemos transformar todo grafo em um grafo bipartido, como visto na Figura 2.8. Tal processo consiste em duplicar cada vértice de modo que cada aresta (a_i, a_j) do grafo original é convertida em duas arestas (a_i, b_j) e (b_i, a_j) no grafo bipartido.

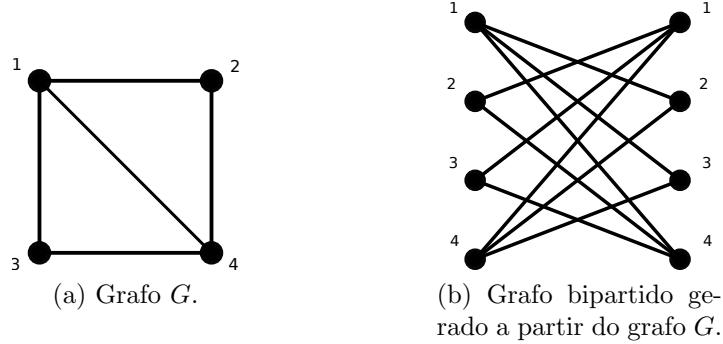


Figura 2.8: Exemplo de um grafo de 4 vértices e seu grafo bipartido gerado a partir do processo de duplicação.

2.4 Passeio Quântico no modelo Escalonado

Para definirmos um passeio quântico no modelo Escalonado sobre um grafo G com N vértices, precisamos definir duas ou mais tesselações sobre G . Como descrito na Seção 2.1, polígonos que compõem uma mesma tesselação no grafo serão disjuntos, enquanto que polígonos de tesselações distintas poderão possuir interseção não-nula entre si.

Cada polígono é associado a um vetor unitário no espaço de Hilbert \mathcal{H}^N . Tais vetores possuem componentes não-nulas que representam os vértices que compõem o polígono, enquanto que as componentes nulas representam os demais vértices do grafo.

Uma tesselação representa neste modelo um operador ortogonal⁶, reflexivo e unitário da forma

$$U_i = 2 \left(\sum_{k=0}^{m_i-1} |\varphi_k\rangle\langle\varphi_k| \right) - I, \quad (2.7)$$

onde, m_i é o número de polígonos pertencentes à tesselação C_i , que gera o operador U_i , e

$$|\varphi_k\rangle = \sum_{j \in \varphi_k} \psi_{k,j} |j\rangle, \quad (2.8)$$

corresponde a um polígono φ_k , com vértices indexados por j , da tesselação C_i , onde $\psi_{k,j}$ são as amplitudes não-nulas complexas do vetor unitário $|\varphi_k\rangle$.

⁶O operador ortogonal é todo aquele cuja matriz inversa é igual a matriz transposta.

Exemplo 2.13 Considere a tesselação da Figura 2.9. Nesta tesselação temos três polígonos, que possuem interseção vazia, respeitando a Definição 2.1.

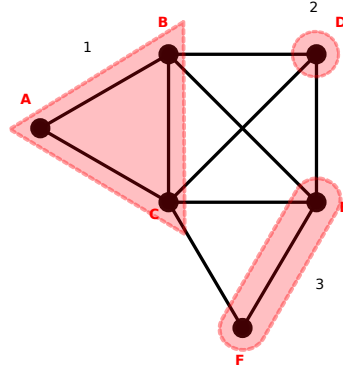


Figura 2.9: Uma tesselação no grafo G com polígonos disjuntos e que cobre todos os nós do grafo, como definido pela Definição 2.1.

Dessa maneira, podemos representar seus polígonos através dos vetores unitários

$$|\varphi_1\rangle = \frac{1}{\sqrt{3}}(|A\rangle + |B\rangle + |C\rangle) = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (2.9)$$

para o polígono 1,

$$|\varphi_2\rangle = |D\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad (2.10)$$

para o polígono 2, e

$$|\varphi_3\rangle = \frac{1}{\sqrt{2}}(|E\rangle + |F\rangle) = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \quad (2.11)$$

para o polígono 3.

Assim, definimos o operador unitário gerado por esta tesselação, na sua forma

matricial, como sendo

$$U_0 = 2 \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix} - I = \begin{bmatrix} -\frac{1}{3} & \frac{2}{3} & \frac{2}{3} & 0 & 0 & 0 \\ \frac{2}{3} & -\frac{1}{3} & \frac{2}{3} & 0 & 0 & 0 \\ \frac{2}{3} & \frac{2}{3} & -\frac{1}{3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \quad (2.12)$$

Tal operador, quando aplicado ao estado do caminhante, fará com que o mesmo possa passear pelas arestas cobertas por polígonos que compõem a tesselação associada a este operador. Neste exemplo, o caminhante estará apto a passear pelas arestas AB, AC, BC e EF .

Exemplo 2.14 Considere a tesselação da Figura 2.10. Nesta tesselação temos três polígonos disjuntos, respeitando a Definição 2.1.

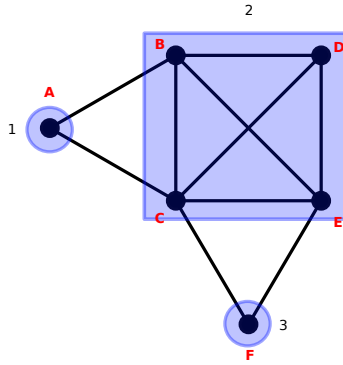


Figura 2.10: Uma tesselação no grafo G , que cobre todos os vértices do grafo e seus polígonos têm interseção vazia entre si.

Dessa maneira, representamos os polígonos desta tesselação através dos vetores unitários

$$|\varphi_1\rangle = |A\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (2.13)$$

para o polígono 1,

$$|\varphi_2\rangle = \frac{1}{2}(|B\rangle + |C\rangle + |D\rangle + |E\rangle) = \frac{1}{2} \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \quad (2.14)$$

para o polígono 2, e

$$|\varphi_3\rangle = |F\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (2.15)$$

para o polígono 3.

Assim definimos o operador unitário gerado por esta tesselação, em sua forma matricial, como sendo

$$U_1 = 2 \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} - I = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.16)$$

Tal operador, quando aplicado ao estado do caminhante, fará com que o mesmo possa passear pelas arestas cobertas por polígonos que compõem a tesselação associada a este operador. Neste exemplo, o caminhante estará apto a passear pelas arestas BC, BD, BE, CD, CE e DE .

Definimos então o operador de evolução em um grafo G \mathcal{T} -tesselável como $U = U_{\mathcal{T}-1} \dots U_1 U_0$, onde \mathcal{T} é a quantidade de tesselações do grafo G . Portugal *et al.* [35] e Portugal [37, 38] mostraram que todo passeio no modelo de Szegedy é também um passeio no modelo Escalonado, visto que o primeiro é um caso particular do segundo. Sendo assim, todo passeio de Szegedy corresponde a um passeio no modelo Escalonado, que ocorre em um grafo 2-tesselável com no máximo um único vértice na interseção de dois polígonos de tesselações distintas.

Capítulo 3

Resultados em Teoria de Grafos

3.1 Limite Superior para o Problema de Cobertura Mínima de Grafos por Tesselações

O problema de cobertura por tesselações em grafos ainda possui várias questões em aberto. Como visto no capítulo anterior, Portugal [38] provou que todo grafo 2-tesselável possui seu grafo clique como sendo um grafo bipartido, ou seja, um grafo 2-colorível. Porém ainda é uma questão em aberto caracterizar grafos cujo número de tesselação seja $T(G) = \mathcal{T}$, $\mathcal{T} > 2$, visto que não podemos afirmar que se um grafo possui número de tesselação $T(G) = \mathcal{T}$, ele possuirá um grafo clique cujo número cromático seja \mathcal{T} .

Neste contexto, estabelecemos um limite superior para o número de tesselação de grafos em geral, apresentado pelo Lema 3.1, fazendo com que o número de tesselação $T(G)$ de um grafo G esteja sempre limitado superiormente por

$$T(G) \leq \chi(K(G)),$$

em que $K(G)$ representa o grafo clique do grafo G e $\chi(K(G))$ representa o número cromático do grafo $K(G)$.

Lema 3.1 [3, 5] *Sejam G um grafo e $K(G)$ seu grafo clique. Temos que $T(G) \leq \chi(K(G))$, onde $T(G)$ indica o número de tesselação de G .*

Demonstração. Sabe-se que cada vértice do grafo clique $K(G)$ representa uma clique maximal no grafo G . Façamos então o seguinte:

- Para cada cor $c_i \in \{c_1, c_2, \dots, c_k\}$ que colore o grafo $K(G)$ faremos uma tesselação T_i para o grafo G ;
- Cada clique maximal em G relacionada com um vértice da cor c_i em $K(G)$ será englobada por um polígono na tesselação T_i , e;

- *Os demais vértices restantes em G que não foram tessellados, serão tessellados por polígonos quaisquer.*

Como cada clique maximal obrigatoriamente será englobada por algum polígono de alguma tesselação, temos que a união das tesselações cobrirá todas as arestas de G . Como cada tesselação gerada por este procedimento cobre todos os vértices sem que haja interseção entre polígonos desta mesma tesselação, temos que todas as propriedades que definem as tesselações são atendidas, logo este procedimento cobre todo o grafo G , logo não é necessário mais que $\chi(K(G))$ tesselações para cobrir G . Logo $T(G) \leq \chi(K(G))$. \square

Com o Lema 3.1 temos que todo grafo G possui o número de tesselação que não necessitará ser superior ao número cromático do grafo clique.

3.2 Classes de Grafos com Número de Tesselação Definido

Podemos enunciar duas famílias de grafos que possuem seus respectivos números de tesselação bem definidos, sendo uma delas exatamente igual ao limite superior proposto, revelando assim que o mesmo é justo. A primeira família é composta pelos grafos roda w -wheel, onde $w > 4$, como exemplificado pela Figura 3.1, em que o grafo roda 6-wheel é 3-tesselável enquanto seu grafo clique, representado por um K_6 , é 6-colorível. A segunda família, que atinge o limite superior, é composta por grafos do tipo Moinho-de-vento generalizado (*Generalized Windmill Graph*) onde cada uma das cliques maximais do mesmo compartilham com todas as demais cliques maximais um único vértice v , sendo v um vértice de articulação e universal, como mostrado pela Figura 3.2 em que temos 3 grafos K_3 onde todos eles compartilham um único vértice, sendo então um grafo cujo número de tesselação é 3, e possuindo o grafo clique K_3 , que é 3-colorível. Os resultados são apresentados a seguir.

Teorema 3.2 [3, 5] *Seja G um grafo roda w -wheel onde $w > 4$, e seja $K(G)$ seu grafo clique. Temos então que $T(G) = \lceil \frac{\chi(K(G))}{2} \rceil = \lceil \frac{w}{2} \rceil$, onde $K(G)$ é o grafo clique de G e $T(G)$ é o número de tesselação de G .*

Demonstração. Todo grafo w -wheel possui w grafos K_3 , compartilhando um vértice universal v e compartilhando uma aresta com cada um de seus vizinhos (esquerdo e direito).

Cada polígono que cobre alguma clique maximal só é capaz de cobrir duas arestas incidentes a v .

Tomando polígonos que cobrem cliques maximais de maneira alternada, temos que necessitamos de $\lceil \frac{w}{2} \rceil$ polígonos, cada qual em uma tesselação.

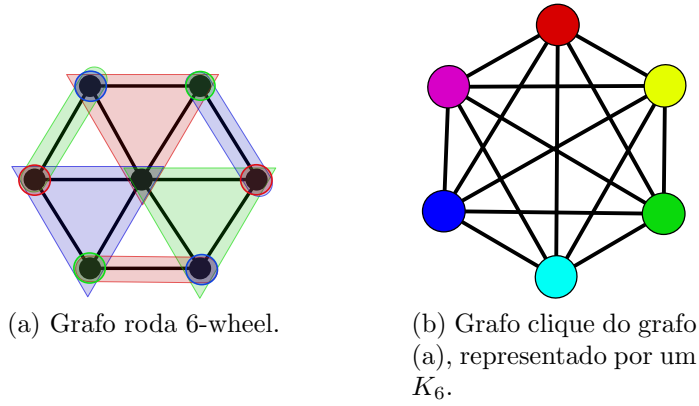


Figura 3.1: Grafo roda 6-wheel, que possui número de tesselação igual a 3, e seu grafo clique representado por um K_6 , que é 6-colorível.

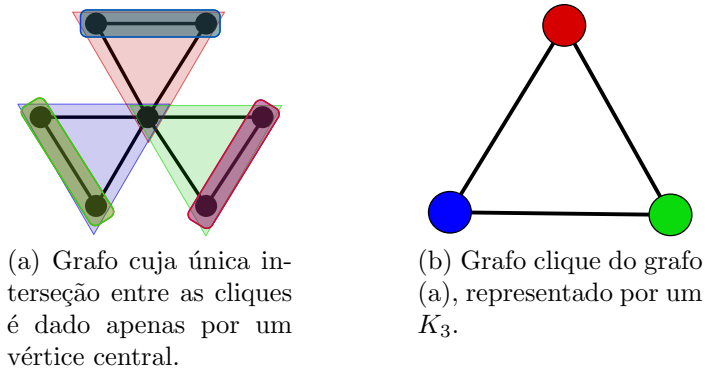


Figura 3.2: Grafo cujo número de tesselação é 3, e seu grafo clique representado por um K_3 , que é 3-colorível.

Para cobrir as demais arestas da borda que não foram cobertas desta maneira, colocamos polígonos como for conveniente. Logo $T(G) \leq \lceil \frac{\chi(K(G))}{2} \rceil$. Por outro lado a retirada de qualquer uma das tesselações utilizadas torna impossível que se cubra o grafo G com $\lceil \frac{w}{2} \rceil - 1$ tesselações, visto que todas as tesselações contém o vértice v . Logo $T(G) = \lceil \frac{w}{2} \rceil$.

□

Teorema 3.3 [3, 5] *Seja G um grafo Moinho-de-vento generalizado. Então $T(G) = \chi(K(G))$, onde $K(G)$ é o grafo clique de G e $T(G)$ é o número de tesselação de G .*

Demonstração. Como G é um grafo cujas cliques possuem interseção entre si dada apenas por um vértice v , que é uma articulação e um vértice universal no grafo, temos que $K(G)$ é um grafo completo. Suponha que G possua c cliques. Sendo assim $K(G)$ é um grafo K_c , que por sua vez é c -colorível. Usando a mesma estratégia apresentada na resolução do Lema 3.1 é possível encontrar uma cobertura de G com c tesselações. No entanto, a retirada de qualquer uma das tesselações T_i utilizadas na cobertura fará com que as $c - 1$ tesselações restantes não sejam capazes de cobrir todas as arestas do grafo. Isso ocorre devido ao fato de que as demais tesselações já

contém o vértice v , impossibilitando assim de que as arestas incidentes a v que ficaram descobertas pela retirada da tesselação T_i possam ser cobertas por algum polígono de alguma das tesselações remanescentes, visto que se isto ocorresse, polígonos de uma mesma tesselação não seriam disjuntos. Logo $T(G) = c = \chi(K(G))$. \square

Outro resultado, de certo modo já esperado, vem imediatamente do resultado apresentado por Portugal [38] ao caracterizar grafos 2-tesseláveis, em conjunto com o resultado demonstrado pelo Lema 3.1. Deste resultado podemos deduzir que se o número cromático do grafo clique for exatamente 3, imediatamente então temos que o grafo original é 3-tesselável por uma cobertura mínima, i.e., o número de tesselação para tal grafo será igual a 3.

Proposição 3.4 *Se $\chi(K(G)) = 3$, ou seja, $K(G)$ é 3-colorível, então $T(G) = 3$, onde $T(G)$ é o número de tesselação do grafo G e $K(G)$ é o grafo clique de G .*

Demonstração. Pelo Lema 3.1 vem que $T(G) \leq 3$. Pela Proposição 2.10 vem que $2 < T(G)$ visto que G não é 2-tesselável devido ao fato de seu grafo clique não ser 2-colorível.

Sendo assim temos que $2 < T(G) \leq 3$. Como $T(G)$ é inteiro, temos $3 \leq T(G) \leq 3$. Logo $T(G) = 3$. \square

É importante ressaltar que o resultado anterior é válido apenas quando o grafo clique de G é 3-colorível, porém o contrário não é verdade, i.e., nem sempre quando um grafo G for 3-tesselável, seu grafo clique será 3-colorível, como é o caso do grafo de *Hajós*, visto no capítulo anterior.

3.3 Algumas Conjecturas

Alguns enunciados a seguir serão formulados como conjecturas por que, apesar de termos algumas ideias de demonstração, estas não receberam ainda uma revisão mais ampla até o momento de escrita da dissertação. No caso da Conjectura 3.6¹, temos uma prova mais rigorosa, que será apresentada na Ref.[3], a ser submetida para o *IX Latin and American Algorithms, Graphs and Optimization Symposium (LAGOS)*. Conjecturamos² que a família de grafos livres de triângulos possuam um número de tesselação $T(G) = \chi'(G)$, onde $\chi'(G)$ representa o índice cromático da coloração de arestas do grafo G . A análise que nos aponta nesta direção está na ideia de que cada cor utilizada na coloração das aresta de G possa ser convertida em polígonos para tesselações associadas as respectivas cores, já que arestas de mesma cor são disjuntas, com isso podemos coloca-las em uma mesma tesselação.

¹Esta conjectura teve contribuição de C. M. H. de Figueiredo [21] e D. Posner [39].

²Esta conjectura teve contribuição de C. M. H. de Figueiredo [21] e D. Posner [39].

O caminho inverso é análogo. Convertemos cada polígono em uma cor para colorir a aresta coberta por tal polígono, associada as suas respectivas tesselações. Como polígonos de uma mesma cor são disjuntos, a coloração de arestas de mesma cor será disjunta também.

Conjectura 3.5 *Se G um grafo livre de triângulos, então $T(G) = \chi'(G)$, onde $\chi'(G)$ é o índice cromático para coloração de arestas de G .*

Justificativa. Seja $C = \{c_1, c_2, \dots, c_k\}$ o conjunto de cores utilizados na coloração própria das arestas do grafo G , que é livre de triângulos. Cada cor pode ser visto como um conjunto de arestas disjuntas. Sendo assim, convertemos cada cor c_i em uma tesselação T_i , onde cada aresta colorida com a cor c_i será tesselada por um polígono da tesselação T_i . Como as arestas são disjuntas, os polígonos gerados também serão disjuntos, e como o conjunto de cores C colore todas as arestas de G , temos que a união das tesselações geradas também irá cobrir todas as arestas de G . Além do mais, o fato de o grafo ser livre de triângulos faz com que tesselar tal grafo seja reduzido a cobrir arestas com tesselações, possuindo assim, exatamente as mesmas restrições da coloração de arestas. Concluimos cada tesselação tomando os vértices restantes por polígonos de tamanho unitário.

Por outro lado, como a coloração de arestas de G é própria, temos que não é possível que se possa diminuir a quantidade de cores utilizadas. Esse fato faz com que também não seja possível reduzir o número de tesselações utilizadas, visto que as interseções entre arestas que impedem a redução do índice cromático são as mesmas que impedem a redução no número de tesselações. Portanto, $T(G) = \chi'(G)$.

□

Essa conjectura faz com que tenhamos fortes indícios de que o problema de cobertura por tesselações seja NP-completo, visto que Koreas [24] mostrou que o problema de coloração de arestas em grafos livres de triângulos cujo grau máximo seja 3 é NP-completo. Deste modo, teríamos uma redução polinomial entre tal problema com o problema de tesselações em grafos [3].

Conjectura 3.6 [3] *Decidir se um grafo é 3-tesselável é NP-completo.*

Outra questão interessante pode ser analisar grafos que possuem uma geração fractal finita. Uma família de grafos bem conhecida que atende esta definição é a família de triângulos de Sierpinski [27], onde temos como exemplo o grafo de *Hajós*, apresentado anteriormente neste trabalho, cuja geração finita é $g = 1$. Conjecturamos que para qualquer grafo S_g da família de triângulos de Sierpinski para qualquer geração finita $g \geq 1$ possui $T(S_g) = 3$. O caso em que $g = 0$ é um caso degenerado já que o grafo S_0 é um K_3 . A conjectura é como segue.

Conjectura 3.7 *Todo grafo S_g presente na família de triângulos de Sierpinski através de uma geração fractal finita $g \geq 1$ possui número de tesselação igual a 3.*

Justificativa. Vamos demonstrar a ideia por indução em g , tomando como base $g = 1$.

Base ($i = 1$): *Esse caso nos dá o próprio Grafo de Hájos, que como mostrado anteriormente possui número de tesselação igual a 3.*

Hipótese: S_k possui número de tesselação $T(S_k) = 3$.

Passo: *Dado que S_k é um triângulo de Sierpinski, temos que o próximo triângulo da família, chamado de S_{k+1} , é construído pela junção de três grafos S_k (chamados de S_{k_1} , S_{k_2} e S_{k_3}). A Figura 3.3 ilustra esta construção.*

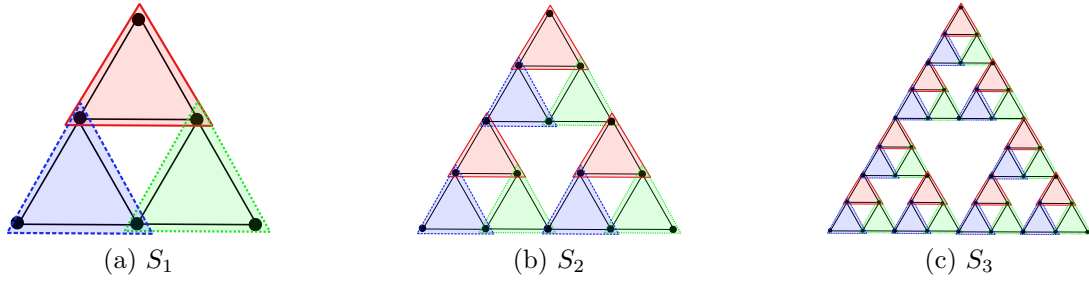


Figura 3.3: Exemplos de grafos da família de triângulos de Sierpinski criados a partir de uma geração fractal. Todos possuem número de tesselação igual a 3. Nestas figuras foram representados apenas os polígonos que cobrem cliques maximais do grafo por questões de legibilidade.

Note que nesta construção nunca é possível a existência de uma interseção entre polígonos de uma mesma tesselação visto que o vértice extremo esquerdo de S_{k_1} passa a ser o mesmo vértice do topo de S_{k_2} , ambos em tesselações distintas. O mesmo ocorre com o vértice do extremo direito de S_{k_1} que passa a ser o mesmo vértice do topo de S_{k_3} , também ambos em tesselações distintas. Ainda temos o vértice do extremo direito de S_{k_2} , que passa a ser o mesmo vértice do extremo esquerdo de S_{k_3} , e estes também estão em tesselações distintas.

Não é possível reduzir o número de tesselações visto que o grafo clique dos grafos da família de triângulos de Sierpinski não são 2-coloríveis.

□

Capítulo 4

Resultados em Computação Quântica

4.1 O problema de *Element k-Distinctness* e a abordagem de Ambainis

Dada uma lista de elementos (x_1, \dots, x_N) , não necessariamente ordenados, tal que $x_1, \dots, x_N \in \{1, \dots, M\}$, o problema de *Element Distinctness* consiste em descobrir se nesta lista existem dois índices $i, j \in [N]$ (sendo $[N] = \{1, 2, 3, \dots, N\}$, ou seja, o conjunto de índices da lista), com $i \neq j$, tal que $x_i = x_j$. De modo geral, podemos estar interessados em descobrir se nesta lista existem k elementos $x_{i_1} = \dots = x_{i_k}$ tais que $i_1 \neq \dots \neq i_k$, obtendo então um problema mais genérico conhecido como *Element k-Distinctness*. Buhrman et al. [16] construíram um algoritmo para resolver o problema de *Element Distinctness* em $O(N^{3/4})$, no entanto, Aaronson & Shi [1] mostraram que o *lower bound* para este problema é $O(N^{2/3})$. Sendo um dos pioneiros no desenvolvimento de algoritmos quânticos que utilizassem passeios quânticos e busca em grafos, Ambainis [8] apresentou um algoritmo para resolver o problema utilizando um grafo de Johnson¹ bipartido, cuja complexidade é $O(N^{k/k+1})$, sendo k o número de repetições de elementos na lista, portanto, ótimo para o caso em que $k = 2$, e necessitando de $O(r(\log N + \log M))$ qubits de memória, onde $r = \lfloor N^{k/k+1} \rfloor$.

Posteriormente surgiram novas abordagens quânticas para resolver o mesmo problema. Belovs [12] apresentou um algoritmo que é baseado no conceito de *Learning-Graphs*, cuja complexidade de *query* é $O(N^{1-2^{k-2}/(2^k-1)})$, sendo assim melhor assintoticamente do que a proposta apresentada por Ambainis para $k \geq 3$.

¹Os vértices de um grafo de Johnson $J(n, p)$ representam p partições de elementos do conjunto $\{1, 2, \dots, n\}$ onde dois vértices serão adjacentes se e somente se a interseção entre eles possuir tamanho $p - 1$.

Apesar do algoritmo apresentado por Belovs [12] possuir uma melhor complexidade assintótica, a abordagem de Ambainis [8] possui uma grande importância teórica no paradigma quântico, e o passeio quântico executado pelo mesmo possui relação com o modelo Escalonado de passeio quântico, introduzido no Capítulo 2.

O algoritmo de Ambainis consiste em fazer uma busca por um vértice marcado em um grafo de Johnson bipartido. Definimos duas partições, A e A' , em que a primeira possui $\binom{N}{r}$ vértices enquanto a segunda possui $\binom{N}{r+1}$ vértices. Seja $r = \lfloor N^{k/k+1} \rfloor$, e seja \mathcal{S} o conjunto de todos os possíveis subconjuntos de índices $S \subseteq [N]$, tal que $|S| = r$. Seja também um elemento $y \in [N], y \notin S$. Os vértices de A correspondem a estados quânticos da forma $|S\rangle|y\rangle|x\rangle$, onde $|S\rangle$ representa as informações dos índices armazenados no subconjunto $S \in \mathcal{S}$, $|y\rangle$ representa um elemento y , e $|x\rangle$ representa o subconjunto de tamanho r , tal que se $S = \{i_1, i_2, \dots, i_r\}$, então $|x\rangle = |x_{i_1}, x_{i_2}, \dots, x_{i_r}\rangle$. Tal partição necessita de um espaço de Hilbert \mathcal{H} de dimensão $\binom{N}{r} M^r (N - r)$ para armazenar todos os estados quânticos da mesma. A segunda partição A' possui vértices que correspondem estados quânticos da forma $|S'\rangle|y'\rangle|x'\rangle$, onde $|S'\rangle$ representa as informações dos índices armazenados em um subconjunto $S' \subseteq [N]$, cuja cardinalidade é $|S'| = r + 1$, $|y'\rangle$ representa um elemento y' tal que $y' \in S'$, e $|x'\rangle$ representa o subconjunto de tamanho $r + 1$ que contém os elementos cujos índices estão presentes em S' , tal que se $S' = \{i_1, i_2, \dots, i_{r+1}\}$, então $|x'\rangle = |x_{i_1}, x_{i_2}, \dots, x_{i_{r+1}}\rangle$. Para armazenar estes vetores é necessário o espaço de Hilbert \mathcal{H}' de dimensão $\binom{N}{r+1} M^{r+1} (r + 1)$. Um vértice será marcado na partição A se tal vértice possuir k elementos iguais em seu subconjunto x . Não haverá vértices marcados na partição A' . Um vértice v , presente na partição A , será adjacente a um vértice v' , presente na partição A' , se e somente se v' representar um subconjunto de índices que contenha o subconjunto de índices representado por v , ou seja, $S' \cap S = S$. Vale ressaltar que na representação do grafo utilizado pelo algoritmo de Ambainis cada um dos vértices de A é representado apenas pelo valor em $|S\rangle$, enquanto que cada um dos vértices de A' é representado apenas pelo valor de $|S'\rangle$.

A Figura 4.1 exemplifica um grafo de Johnson bipartido para um conjunto $[N] = \{1, 2, 3, 4\}$ e um valor de $k = 2$. Para tal, temos que a lista de elementos é $\ell = \{1, 2, 1, 4\}$, fazendo com que o vértice que representa o estado $|13\rangle$ seja marcado.

As primeiras duas etapas do algoritmo de Ambainis (ver Algoritmo 1) geram uma superposição uniforme de todos os vértices que se encontram na partição A . Posteriormente na terceira etapa, que é repetida $O\left(\left(\frac{N}{r}\right)^{\frac{k}{2}}\right)$ vezes, primeiro o algoritmo aplica um operador condicional conhecido como *phase flip*, cujo efeito é mudar a fase da amplitude quântica de todos os vértices que possuem k elementos iguais em seus respectivos subconjuntos x , marcando assim, tais vértices. Em seguida ainda na terceira etapa, o algoritmo de fato executa os passos do passeio quântico $O(\sqrt{r})$ vezes. Finalmente, na quarta etapa, devemos efetuar a medição.

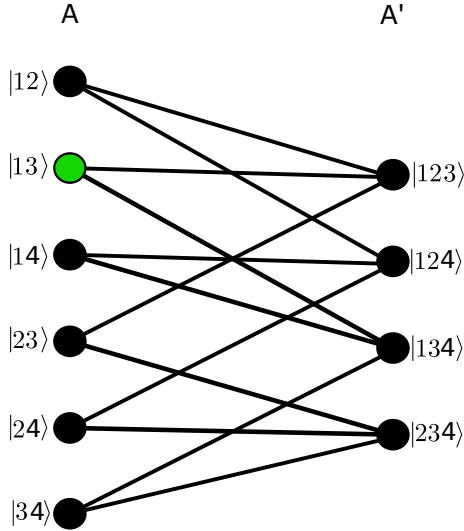


Figura 4.1: Grafo de Johnson bipartido para um conjunto $[N] = \{1, 2, 3, 4\}$ e um valor de $k = 2$. Para tal, temos que a lista de elementos é $\ell = \{1, 2, 1, 4\}$

As etapas do algoritmo de passeio quântico podem ser divididas em duas partes (ver Algoritmo 2). A primeira parte, composta por três etapas, tem por objetivo movimentar o caminhante da partição A para a partição A' . Isso é feito através de um mapeamento que aponta o caminhante para seus vizinhos presentes na partição A' . Após tal mapeamento as duas operações seguintes fazem com que o caminhante passe do espaço \mathcal{H} para o espaço \mathcal{H}' através do aumento dos conjuntos representados por $|S\rangle$ e $|x\rangle$, fazendo com que estes passem a ser, respectivamente $|S'\rangle$ e $|x'\rangle$, onde $|S'\rangle = |S \cup \{y\}\rangle$ e $|x'\rangle = |x \cup \{x_y\}\rangle$. A segunda parte, também composta por três etapas, tem por objetivo mover o caminhante de A' para A . Isto é feito através de um novo mapeamento que aponta o caminhante para seus vizinhos em A . Após isso, as duas operações seguintes se encarregam de reduzir o tamanho dos conjuntos representados por $|S'\rangle$ e $|x'\rangle$ fazendo com que estes passem a ser, respectivamente, $|S\rangle$ e $|x\rangle$, onde $|S\rangle = |S' \setminus \{y'\}\rangle = |S \cup \{y\} \setminus \{y'\}\rangle$ e $|x\rangle = |x' \setminus \{x_{y'}\}\rangle = |x \cup \{x_y\} \setminus \{x_{y'}\}\rangle$. Caso o leitor almeje o estudo de tal algoritmo, poderá fazer uso da ferramenta *QEDS: Quantum Element Distinctness Simulator* que executa e detalha passo-a-passo o algoritmo de Ambainis para o caso em que $k = 2$ [4].

Ambainis também propôs um algoritmo para resolver o problema de *Multiple Element k-Distinctness* (Múltiplos *Element k-Distinctness*). Nesta abordagem, o algoritmo irá retornar todos os elementos da lista que possuam pelo menos k repetições do mesmo elemento [8].

Como mostrado no Algoritmo 3, o procedimento consiste na aplicação de alguns passos clássicos, o que faz com que o Algoritmo 1 seja aplicado sucessivas vezes com o objetivo de obter todos os elementos que possuam pelo menos k repetições na lista de entrada.

Algoritmo 1: Algoritmo de Ambainis para o problema de *Element k -Distinctness*

1. Gere a superposição uniforme

$$\frac{1}{\sqrt{\binom{N}{r}(N-r)}} \sum_{S \in \mathcal{S}} \sum_{y \notin S} |S\rangle |y\rangle.$$

2. Consulte todos os x_i para $i \in S$. Isto faz com que o estado se transforme em

$$\frac{1}{\sqrt{\binom{N}{r}(N-r)}} \sum_{S \in \mathcal{S}} \sum_{y \notin S} |S\rangle |y\rangle \bigotimes_{i \in S} |x_i\rangle.$$

3. Repita $t_1 = O\left(\left(\frac{N}{r}\right)^{k/2}\right)$ vezes:

- (a) Aplique o operador condicional *phase flip* (Transforma $|S\rangle |y\rangle |x\rangle \rightarrow -|S\rangle |y\rangle |x\rangle$) para todo $S \in \mathcal{S}$, tal que $x_{i_1} = x_{i_2} = \dots = x_{i_k}$, $i_1, \dots, i_k \in S$, $i_1 \neq i_2 \neq \dots \neq i_k$
- (b) Realize $t_2 = O(\sqrt{r})$ passos do passeio quântico (Algoritmo 2).

4. Efetue a medição do estado final. Verifique se S contém uma k -colisão e responda “Existe uma k -colisão” ou “não existe k -colisão”, de acordo com o resultado.
-

O algoritmo de Ambainis para *Element k -Distinctness*, além de possuir grande importância teórica e ser alvo de estudo de pesquisadores da área, também contribuiu como inspiração o desenvolvimento do modelo de caminhada quântica de Szegedy [43]², sendo tal modelo uma generalização da caminhada executada pelo procedimento de Ambainis.

4.2 Algoritmo Quântico para o Problema de *Element k -Distinctness* no Modelo Escalonado

A definição do problema, bem como as variáveis, é exatamente a mesma vista na seção anterior. Relembrando, definimos $r = \lfloor N^{k/k+1} \rfloor$, sendo k o número de repetições buscadas na lista de entrada. Temos o conjunto \mathcal{S} de todos os possíveis subconjuntos $S \subseteq [N]$, tal que $|S| = r$, e um elemento y , tal que $y \in [N] \setminus S$, fazendo com que tenhamos $N - r$ valores de y associados a cada subconjunto S . Construiremos um grafo G 2-tesselável, onde cada vértice deste grafo representará

²A primeira versão do trabalho de Ambainis foi divulgada em 2002 no repositório do *arXiv*, sendo posteriormente publicado pela *SIAM Journal of Computing*, em 2007, enquanto que o trabalho de Szegedy foi publicado em 2004.

Algoritmo 2: Seis subpassos que descrevem um único passo de passeio quântico

1. Aplique a transformação mapeando $|S\rangle |y\rangle$ para

$$|S\rangle \left(\left(-1 + \frac{2}{N-r} \right) |y\rangle + \frac{2}{N-r} \sum_{y' \notin S, y' \neq y} |y'\rangle \right),$$

onde $S \in \mathcal{S}$ e $y \notin S$ são registradores do espaço \mathcal{H} . (Essa transformação é uma variação da “transformação de difusão” em [23].)

2. Mapeie o estado de \mathcal{H} para \mathcal{H}' , adicionando y ao S e alterando x para um vetor de $k+1$ posições, introduzindo 0 no local correspondente ao y .
3. Busque por x_y e insira-o no local de x correspondente ao y .
4. Aplique a transformação mapeando $|S\rangle |y\rangle$ para

$$|S\rangle \left(\left(-1 + \frac{2}{r+1} \right) |y\rangle + \frac{2}{r+1} \sum_{y' \in S, y' \neq y} |y'\rangle \right)$$

em que $S \in \mathcal{S}$ e $y' \in S$.

5. Reverta para zero o elemento de x correspondente ao novo y usando a entrada para consultar x_y (Lembre-se que a etapa 2 é uma operação reversível).
 6. Mapeie o estado de volta para \mathcal{H} removendo o componente 0 correspondente ao y de x e removendo y de S .
-

um par (S, y) .

Definição 4.1 *Definimos um grafo G com $\binom{N}{r}(N-r)$ vértices. Um vértice v corresponde a um par (S, y) , em que $S \in \mathcal{S}$ e $y \in [N] \setminus S$. Dois vértices v e v' serão adjacentes, para $v = (S, y)$ e $v' = (S', y')$, se e somente se ou (i) $S' = S$ e $y \neq y'$, ou (ii) $S' = S \cup \{y\} \setminus \{y'\}$.*

Um fato importante a ser mencionado é que o grafo construído pela Definição 4.1 é grafo linha do grafo utilizado por Ambainis em sua abordagem, como mostra a Proposição 4.3.

Definição 4.2 *Dado um grafo G , seu grafo linha $L(G)$ é tal que:*

- Cada vértice em $L(G)$ corresponde a uma aresta em G ;
- Um par de vértices em $L(G)$ são adjacentes se e somente se as arestas correspondentes possuem um vértice em comum em G .

Algoritmo 3: *Multiple Element k-Distinctness*

1. Seja $F_1 = [N]$ e $j = 1$.
2. Enquanto $|F_j| > \max(r, \sqrt{N})$ repita:
 - (a) Execute o Algoritmo 1 sobre $x_i, i \in F_j$, usando uma memória de tamanho $r_j = \frac{r|F_j|}{N}$. Faça a medição do estado final, obtendo um conjunto S , pare se encontrar uma k -colisão;
 - (b) Seja q_j um expoente par de um primo com $|F_j| \leq q_j \leq (1 + \frac{1}{2k^2})|F_j|$. Selecione uma permutação aleatória π_j sobre q_j , de uma $\frac{1}{N}$ -aproximação de uma família $2k \log N$ -wise independent de permutações;
 - (c) Seja

$$F_{j+1} = \left\{ \pi_1^{-1} \dots \pi_j^{-1}(i) \in \left[\left[\frac{2k}{2k+1} q_j \right] \right] \right\};$$

- (d) Seja $j = j + 1$.
 3. Se $|F_j| \leq r$, pesquise por todos os $x_i, i \in F_j$ classicamente, e pare se encontrar uma k -colisão;
 4. Se $|F_j| \leq \sqrt{N}$, execute a busca de Grover([23]) sobre o conjunto de no máximo $N^{k/2}k$ -tuplas $(i-1, \dots, i_k)$ de um par distinto $i_1, \dots, i_k \in F_j$, procurando por uma tupla (i_1, \dots, i_k) tal que $x_{i_1} = \dots = x_{i_k}$ e verifique se há uma k -colisão.
-

Proposição 4.3 *Seja G o grafo bipartido construído para o algoritmo de Ambainis. O grafo linha de G é exatamente o grafo H construído segundo a Definição 4.1.*

Demonstração. *Seja G o grafo bipartido construído para o algoritmo de Ambainis, cujas partições chamaremos aqui de A e B . Os vértices $v \in A$ correspondem a subconjuntos $S_a \subseteq [N]$, tal que $|S_a| = r$, e os vértices $u \in B$ correspondem a subconjuntos $S_b \subseteq [N]$ tal que $|S_b| = r + 1$ e $S_a \cap S_b = S_a$. Portanto, $S_b \setminus S_a$ possui um único elemento. Seja uma rotulação das arestas de G tal que cada aresta receba um rótulo (S_a, S_b) , indicando que a aresta é incidente ao vértice de A correspondente ao subconjunto S_a , e incidente ao vértice de B correspondente ao subconjunto S_b . Desse modo, duas arestas são incidentes ao mesmo vértice da partição A se possuírem em seus rótulos o mesmo valor em S_a , e serão incidentes ao mesmo vértice em B se possuírem em seus rótulos o mesmo valor de S_b .*

Vamos mostrar que (i) cada vértice em H corresponde a uma aresta em G , e que (ii) dois vértices em H são adjacentes se e somente se as arestas de G associadas a esses vértices incidem a um mesmo vértice em G .

(i) Pela Definição 4.1 temos que um vértice arbitrário de H é rotulado por

(S, y) , tal que $S \subseteq [N]$, $|S| = r$, e $y \in [N] \setminus S$. Seja $S_a = S$ e $S_b = S \cup \{y\}$. Então um vértice (S, y) de H é mapeado na aresta (S_a, S_b) de G . Analogamente, seja (S_a, S_b) uma aresta arbitrária de G . Seja $S = S_a$, e seja y o único elemento do conjunto $S_b \setminus S_a$. Então uma aresta (S_a, S_b) de G é mapeada no vértice (S, y) de H . Além disso, $|V(H)| = |E(G)| = \binom{N}{r}(N - r)$. Portanto, mostramos que existe uma bijeção entre o conjunto de vértices de H e o conjunto de arestas de G .

(ii) A demonstração deste item será dividida em duas partes.

(\rightarrow) Sejam v, v' dois vértices adjacentes em H , tal que v possui rótulo (S, y) e v' possui rótulo (S', y') . O vértice v é mapeado na aresta de G rotulada por (S_a, S_b) , em que $S_a = S$ e $S_b = S \cup \{y\}$. Analogamente, o vértice v' é mapeado na aresta de G rotulada por (S'_a, S'_b) , em que $S'_a = S'$ e $S'_b = S' \cup \{y'\}$. Pela Definição 4.1 sabemos que v, v' são adjacentes se uma das seguintes condições for satisfeita: ou (a) $S = S'$, ou (b) $S' = S \cup \{y\} \setminus \{y'\}$. Se v, v' forem adjacentes pelo caso (a), então v é mapeado na aresta $(S, S \cup \{y\})$ de G , e v' é mapeado na aresta $(S, S \cup \{y'\})$ de G , que são arestas incidentes ao mesmo vértice em G . Se v, v' forem adjacentes pelo caso (b), então v é mapeado na aresta $(S, S \cup \{y\})$ e v' é mapeado na aresta $(S \cup \{y\} \setminus \{y'\}, (S \cup \{y\} \setminus \{y'\}) \cup \{y'\}) = (S \cup \{y\} \setminus \{y'\}, S \cup \{y\})$, pois em $((S \cup \{y\} \setminus \{y'\}) \cup \{y'\})$, o conjunto dado por $\{S \cup \{y\}\}$ terá o elemento y' retirado (a partir da operação de subtração) e novamente inserido (a partir da operação de união), que são arestas incidentes ao mesmo vértice em G .

Portanto, demonstramos que se dois vértices em H são adjacentes, então as arestas de G associadas a esses vértices incidem a um mesmo vértice em G .

(\leftarrow) Sejam (S_a, S_b) e (S'_a, S'_b) duas arestas em G incidentes a um mesmo vértice. A aresta (S_a, S_b) é mapeada no vértice (S_a, y) de H , em que $y \in S_b \setminus S_a$. Analogamente, a aresta (S'_a, S'_b) é mapeada no vértice (S'_a, y') de H , em que $y' \in S'_b \setminus S'_a$. Para as arestas de G serem incidentes ao mesmo vértice, uma das seguintes condições precisa ser satisfeita: ou (a) $S_a = S'_a$, se o vértice em comum estiver na partição A , ou (b) $S_b = S'_b$, se o vértice em comum estiver na partição B . Se as arestas de G forem incidentes pelo caso (a), então a aresta (S_a, S_b) é mapeada no vértice (S_a, y) de H , em que $y \in S_b \setminus S_a$, e a aresta (S'_a, S'_b) é mapeada no vértice (S_a, y') de H , em que $y' \in S'_b \setminus S'_a$. Os vértices rotulados por (S_a, y) e (S_a, y') são adjacentes em H . Se as arestas de G forem incidentes pelo caso (b), então $S_b = S'_b$. Temos então que, como definido pela Definição 4.1, $S'_a = S_a \cup \{y\} \setminus \{y'\}$, que resulta em $S'_a = S_a \cup \{S_b \setminus S_a\} \setminus \{S_b \setminus S'_a\}$, que nos leva em $S'_a = S_b \setminus \{S_b \setminus S'_a\}$, que por sua vez resulta em $S'_a = S'_a$. Logo, os vértices rotulados por (S_a, y) e (S'_a, y') são adjacentes em H .

Portanto, mostramos que se duas arestas em G incidem a um mesmo vértice, então os vértices correspondentes em H são adjacentes. □

Exemplo 4.4 *Seja o conjunto de índices $[N] = \{1, 2, 3, 4\}$, onde temos $r = 2$ e $k = 2$. Seja então o grafo G bipartido utilizado por Ambainis em sua abordagem. Temos que seu grafo linha é dado por $L(G)$, que é exatamente o grafo construído segundo a Definição 4.1, como mostra a Figura 4.2.*

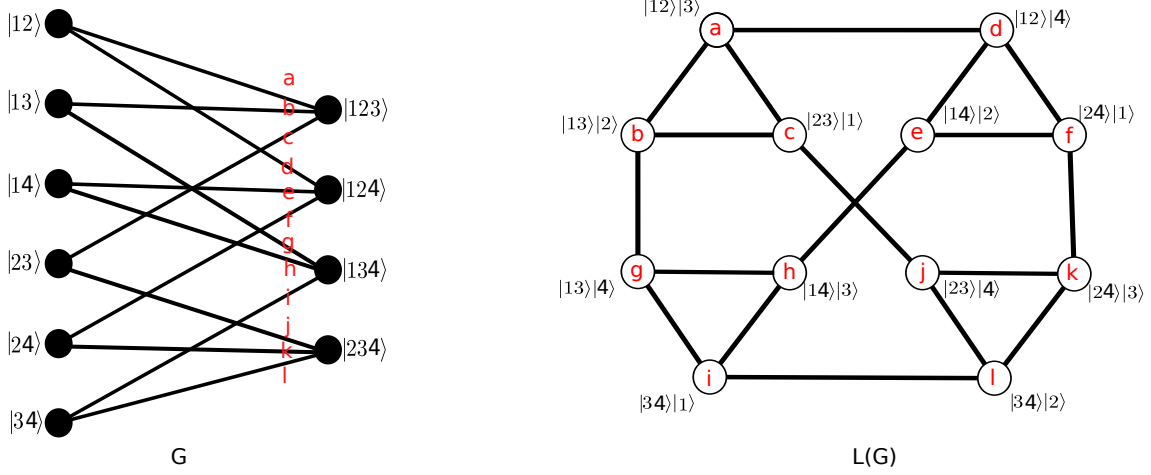


Figura 4.2: Grafo de Johnson bipartido G , utilizado por Ambainis em sua abordagem, e seu grafo linha $L(G)$, que é exatamente o grafo construído pela Definição 4.1, ambos para $[N] = \{1, 2, 3, 4\}$, onde temos $r = 2$ e $k = 2$.

Após construirmos este grafo, para executarmos um passeio quântico no modelo Escalonado, precisamos definir pelo menos duas tesselações para que assim, possamos gerar os operadores de evolução para a caminhada. Antes vamos provar que o grafo G construído pela Definição 4.1 é 2-tesselável.

Proposição 4.5 [38] *Um grafo é 2-tesselável se e somente se seu grafo clique é 2-colorível.*

Proposição 4.6 *O grafo G construído pela Definição 4.1 possui um grafo clique $K(G)$ bipartido.*

Demonstração. A Definição 4.1 descreve duas possibilidades de ocorrência de arestas (v, v') no grafo G , em que $v = (S, y)$ e $v' = (S', y')$, que são as seguintes: (i) $S' = S$ e $y \neq y'$, tal que $S \in \mathcal{S}$ e $y, y' \notin S$, ou (ii) $S' = S \cup \{y\} \setminus \{y'\}$, tal que $S \in \mathcal{S}$ e $y, y' \notin S, y' \in S$.

No caso (i) todos os vértices que possuem o mesmo valor S em seu par (S, y) irão formar uma clique, visto que tais vértices estão conectados entre si, por definição. O mesmo vale para o caso (ii), onde $S' = S \cup \{y\} \setminus \{y'\}$.

Suponha por contradição que duas cliques a , com p vértices, e b , com q vértices, geradas pelo caso (i) possuam o vértice $v = (S_r, y_r)$ em comum. Como v pertence a clique a , então $S_r = S_{a_i}$, $1 \leq i \leq p, a_i \neq r$ e como v pertence a clique b , então $S_r = S_{b_j}$, $1 \leq j \leq q, b_j \neq r$. Com isso temos que $S_{a_i} = S_{b_j}$, $\forall i, j, 1 \leq i \leq p$ e

$1 \leq j \leq q$, indicando que todos os demais vértices de a possuem adjacência com todos os demais vértices de b . Logo a e b na verdade são uma única clique, um absurdo. Logo as cliques geradas no caso (i) não possuem vértices em comum. A demonstração é análoga para as cliques geradas pelo caso (ii).

Sendo assim, ao gerarmos o grafo clique $K(G)$ teremos vértices que podem ser colocados em uma partição A (cliques geradas pelo caso (i)) e na partição A' (cliques geradas pelo caso (ii)). Sendo assim, temos que o grafo clique $K(G)$ é bipartido e portanto, 2-colorível. \square

Corolário 4.7 O grafo G gerado pela Definição 4.1 é 2-tesselável.

Demonstração. Segue diretamente da Proposição 4.5 e da Proposição 4.6. \square

Sendo assim, definimos duas tesselações para o grafo G , como segue.

Definição 4.8 Podemos definir duas tesselações no grafo G . A primeira tesselação, chamada de α , é definida por polígonos que cobrem cliques onde para todo o par de vértices (v, v') tal que $v = (S, y)$ e $v' = (S', y')$, temos $S = S'$ e $y \neq y'$, onde $y \notin S$ e $y' \notin S'$. A segunda tesselação, chamada de β , é definida por polígonos que cobrem cliques onde todo o par de vértices (v, v') , tal que $v = (S, y)$ e $v' = (S', y')$, temos $S' = S \cup \{y\} \setminus \{y'\}$, onde $y \notin S$ e $y' \in S$.

A partir destas duas tesselações, iremos extrair os dois operadores de evolução que juntos irão evoluir o estado quântico que representa o caminhante ao longo da caminhada. Como cada vértice representa um par (S, y) , representamos o caminhante como sendo o estado quântico $|S\rangle|y\rangle$ no espaço de Hilbert $\binom{N}{r}(N-r)$.

Definimos então os operadores de evolução U_0 e U_1 . O operador U_0 , que é gerado pela tesselação α (Definição 4.8.), cobre arestas onde o par de vértices incidentes são da forma $S = S'$, enquanto o operador U_1 , que é gerado pela tesselação β (Definição 4.8.), cobre arestas cujo par de vértices incidentes possuam valores S e S' da forma $S' = S \cup \{y\} \setminus \{y'\}$. Como existem $\binom{N}{r}$ valores possíveis para S , ambos os operadores de evolução irão percorrer todo o conjunto \mathcal{S} , cuja cardinalidade é $\binom{N}{r}$. Iremos mostrar que os operadores U_0 e U_1 do modelo Escalonado, para as tesselações aqui descritas, podem ser escritos como

$$U_0 = \frac{2}{N-r} \left[\sum_{S \in \mathcal{S}} \sum_{\substack{y \notin S \\ y \neq y'}} \left(|S, y\rangle\langle S, y| + \sum_{\substack{y' \notin S \\ y \neq y'}} |S, y'\rangle\langle S, y| \right) \right] - I, \quad (4.1)$$

e

$$U_1 = \frac{2}{r+1} \left[\sum_{S \in \mathcal{S}} \sum_{\substack{y \notin S \\ y \neq y'}} \left(|S, y\rangle\langle S, y| + \sum_{\substack{y' \in S \\ y \neq y'}} |S', y'\rangle\langle S, y| \right) \right] - I, \quad (4.2)$$

onde $|S'\rangle = |S \cup \{y\} \setminus \{y'\}\rangle$. Note que, no caso do operador U_0 , os valores de y , assim como os valores de y' , pertencem ao conjunto de índices $[N]$, no entanto não pertencem ao conjunto S , enquanto que no caso do operador U_1 , os valores de y são mantidos como definidos anteriormente, porém nesse caso, os valores de y' além de pertencerem ao conjunto $[N]$, devem ainda pertencer ao conjunto S .

Apesar do aspecto um pouco diferente, os operadores descritos pelas Equações (4.1) e (4.2) são como os operadores de tesselações descritos no Capítulo 2. Relembrando de maneira breve, vimos que uma tesselação define no modelo Escalonado um operador ortogonal, reflexivo e unitário da forma

$$U_i = 2 \left(\sum_{k=0}^{m_i-1} |\varphi_k\rangle\langle\varphi_k| \right) - I, \quad (4.3)$$

onde, m_i é o número de polígonos pertencentes a tesselação C_i , que gera o operador U_i , e

$$|\varphi_k\rangle = \sum_{j \in \varphi_k} \psi_{k,j} |j\rangle, \quad (4.4)$$

onde $\psi_{k,j}$ são as amplitudes não-nulas complexas do vetor unitário $|\varphi_k\rangle$, que representa o polígono φ_k em G com vértices indexados por j .

Vamos então mostrar que pela Equação 4.3, tomando amplitudes uniformes para os polígonos na Equação 4.4, os operadores descritos pelas Equações (4.1) e (4.2) são operadores do tipo descrito de maneira genérica pela Equação (4.3).

Proposição 4.9 *O operador descrito pela Equação (4.1) é um operador do mesmo tipo do operador descrito pela Equação (4.3), sendo deste modo um operador reflexivo, ortogonal e unitário.*

Demonstração. Substituindo a Equação (4.4) na Equação (4.3) temos

$$U_0 = 2 \left(\sum_{k=0}^{m_0-1} \sum_{j' \in \varphi_k} \sum_{j \in \varphi_k} \psi_{k,j'} \psi_{k,j}^* |j'\rangle\langle j| \right) - I. \quad (4.5)$$

Temos que o operador U_0 atua sobre polígonos da tesselação α , portanto atua sobre polígonos que cobrem cliques onde para todo par de vértices (v, v') , tal que $v = (S, y)$ e $v' = (S', y')$, temos $S = S'$ e $y \neq y'$, com $y, y' \notin S$. Sendo assim, possuímos $(N - r)$ vértices em cada polígono, pois para cada possível valor de S existem $(N - r)$ possíveis valores para y . Tomando os polígonos com amplitudes uniformes, temos $\psi_{k,j} = \frac{1}{\sqrt{N-r}}$. Logo, temos nosso operador como sendo

$$U_0 = \frac{2}{N - r} \left(\sum_{k=0}^{m_0-1} \sum_{j' \in \varphi_k} \sum_{j \in \varphi_k} |j'\rangle\langle j| \right) - I. \quad (4.6)$$

Cada vértice j corresponde a um par $|S, y\rangle$. Como temos $S \subseteq \mathcal{S}$, e nesse caso $S = S'$, podemos reescrever nosso operador como

$$U_0 = \frac{2}{N-r} \left(\sum_{S \in \mathcal{S}} \sum_{y \notin S} \sum_{y' \notin S} |S, y'\rangle \langle S, y| \right) - I. \quad (4.7)$$

Podemos reescrever os somatórios em y e y' de tal maneira que explicitemos os casos em que $y = y'$ e $y \neq y'$. Sendo assim, temos

$$U_0 = \frac{2}{N-r} \left[\sum_{S \in \mathcal{S}} \sum_{y \notin S} \left(|S, y\rangle \langle S, y| + \sum_{\substack{y' \notin S \\ y \neq y'}} |S, y'\rangle \langle S, y| \right) \right] - I, \quad (4.8)$$

que é exatamente a equação descrita pela Equação (4.1). \square

Proposição 4.10 O operador descrito pela Equação (4.2) é um operador do mesmo tipo do operador descrito pela Equação (4.3).

Demonstração. Substituindo a Equação (4.4) na Equação (4.3) temos

$$U_1 = 2 \left(\sum_{k=0}^{m_1-1} \sum_{j' \in \varphi_k} \sum_{j \in \varphi_k} \psi_{k,j'} \psi_{k,j}^* |j'\rangle \langle j| \right) - I. \quad (4.9)$$

Temos que o operador U_1 atua sobre polígonos da tesselação β , portanto atua sobre polígonos que cobrem cliques onde todo par de vértices (v, v') tal que $v = (S, y)$ e $v' = (S', y')$, temos $S' = S \cup \{y\} \setminus \{y'\}$ ($S = S' \cup \{y'\} \setminus \{y\}$). Sendo assim, possuímos $(r+1)$ vértices em cada polígono. Tomando os polígonos com amplitudes uniformes, temos $\psi_{k,j} = \frac{1}{\sqrt{r+1}}$. Logo, temos nosso operador como sendo

$$U_1 = \frac{2}{r+1} \left(\sum_{k=0}^{m_1-1} \sum_{j' \in \varphi_k} \sum_{j \in \varphi_k} |j'\rangle \langle j| \right) - I. \quad (4.10)$$

Cada vértice j corresponde a um par $|S, y\rangle$. Como temos $S \subseteq \mathcal{S}$, podemos reescrever nosso operador como

$$U_1 = \frac{2}{r+1} \left(\sum_{S \in \mathcal{S}} \sum_{y \notin S} \sum_{y' \in S} |S', y'\rangle \langle S, y| \right) - I. \quad (4.11)$$

onde, neste caso, $S' = S \cup \{y\} \setminus \{y'\}$.

Podemos reescrever os somatórios em y e y' de tal maneira que explicitemos os casos em que $y = y'$ e $y \neq y'$. Sendo assim, temos

$$U_1 = \frac{2}{r+1} \left[\sum_{S \in \mathcal{S}} \sum_{y \notin S} \left(|S, y\rangle \langle S, y| + \sum_{\substack{y' \in S \\ y \neq y'}} |S', y'\rangle \langle S, y| \right) \right] - I, \quad (4.12)$$

onde $|S'\rangle = |S \cup \{y\} \setminus \{y'\}\rangle$.

A equação descrita acima é exatamente a equação descrita pela Equação (4.2).

□

Note que o operador U_0 permite que o caminhante se mova por arestas cujos vértices incidentes possuem o mesmo valor no registrador $|S\rangle$, enquanto o operador U_1 faz com que o caminhante se movimente por arestas cujos vértices incidentes possuam valores no registrador $|S\rangle = |S' \cup \{y'\} \setminus \{y\}\rangle$.

O Algoritmo 4 apresenta o algoritmo para resolver o problema de *Element Distinctness* no modelo Escalonado.

Algoritmo 4: Algoritmo no modelo Escalonado para *Element k -Distinctness*

1. Gere a superposição uniforme

$$\frac{1}{\sqrt{\binom{N}{r}(N-r)}} \sum_{|S|=r, y \notin S} |S, y\rangle.$$

2. Consulte todos os x_i para $i \in S$. Isto faz com que o estado se transforme em

$$\frac{1}{\sqrt{\binom{N}{r}(N-r)}} \sum_{S \in \mathcal{S}} \sum_{y \notin S} |S\rangle |y\rangle \bigotimes_{i \in S} |x_i\rangle.$$

3. Repita $t_1 = O((N/r)^{k/2})$ vezes:

- (a) Aplique o operador condicional *phase flip* (Transforma $|S\rangle |y\rangle |x\rangle \rightarrow -|S\rangle |y\rangle |x\rangle$) para todo S de tal modo que $x_{i_1} = x_{i_2} = \dots = x_{i_k}$ para k distintos $i_1, \dots, i_k \in S$.

- (b) Execute $t_2 = O(\sqrt{r})$:

- i. Aplique o operador U_0 ;
- ii. Altere o registrador $|x\rangle$ para um vetor de tamanho $r + 1$, colocando 0 na posição correspondente a y . Procure por x_y e insira-o em x no local correspondente a y ;
- iii. Aplique o operador U_1 ;
- iv. Apague de x o elemento correspondente ao novo y usando-o como entrada para procurar x_y . Remova o valor 0 do registrador $|x\rangle$, transformando-o novamente em um vetor de tamanho r .

4. Efetue a medição do estado final. Verifique se S contém uma k -colisões e responda “Existe uma k -colisão” ou “não existe k -colisão”, de acordo com o resultado.
-

O algoritmo para encontrar a resposta para o problema de *Multiple Element k -Distinctness* é semelhante ao apresentado no Algoritmo 3, no entanto, no passo

em que é chamado o algoritmo quântico de Ambainis, é chamado na verdade o Algoritmo 4.

Exemplo 4.11 *Vamos demonstrar a execução do operadores do algoritmo apresentado. Para simplificar, ao invés de analisarmos o estado inicial do caminhante como uma superposição, tomaremos o caminhante em um único vértice a fim de que fique visível como a amplitude do estado quântico é modificada e espalhada ao longo das aplicações dos operadores U_0 e U_1 .*

Seja o grafo construído segundo a Definição 4.1 e tesselado segundo a Definição 4.8 para $N = 4$ tal que $[N] = \{1, 2, 3, 4\}$ e $k = 2$. Seja então o estado inicial do caminhante $|\Psi_0\rangle = |12\rangle|3\rangle$, como mostra a Figura 4.3.

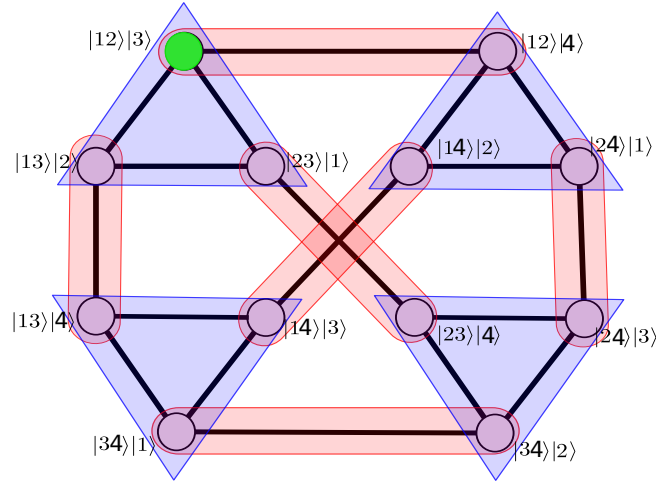


Figura 4.3: Grafo construído segundo a Definição 4.1 e tesselado segundo a Definição 4.8 para $N = 4$ tal que $[N] = \{1, 2, 3, 4\}$ e $k = 2$.

Ao aplicarmos o operador U_0 , descrito na Equação (4.1) o caminhante passará a estar em uma superposição entre os vértices $|12\rangle|3\rangle$ e $|12\rangle|4\rangle$, resultando no estado $|\Psi_1\rangle = \alpha|12\rangle|3\rangle + \beta|12\rangle|4\rangle$, onde $\alpha, \beta \in \mathbb{C}$ representam amplitudes quânticas, como mostrado na Figura 4.4.

Após isso, aplica-se o operador U_1 , descrito pela Equação (4.2), levando-nos ao estado $|\Psi_2\rangle = \alpha_1|12\rangle|3\rangle + \alpha_2|13\rangle|2\rangle + \alpha_3|23\rangle|1\rangle + \beta_1|12\rangle|4\rangle + \beta_2|14\rangle|2\rangle + \beta_3|24\rangle|1\rangle$, como mostrado pela Figura 4.5.

Dessa forma a amplitude quântica do estado inicial do caminhante irá se espalhar pelos vértices do grafo. Como o estado inicial do caminhante originalmente é uma superposição de todos os vértices, temos que ao marcarmos um vértice utilizando a operação de phase flip para realizarmos a busca por ele, a medida que os passos quânticos forem executados, ocorrerá uma convergência das amplitudes no vértice que foi marcado devido a soma e subtração das amplitudes dos estados que compõem a superposição.

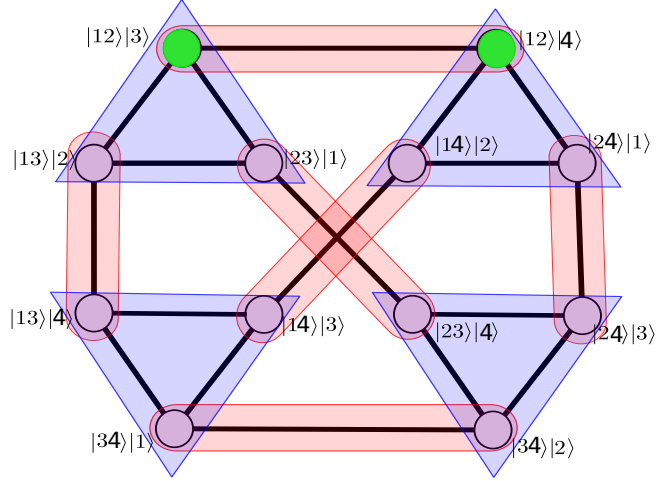


Figura 4.4: Grafo construído segundo a Definição 4.1 e tesselado segundo a Definição 4.8 após a aplicação do operador U_0 sobre o estado quântico $|\Psi_0\rangle$.

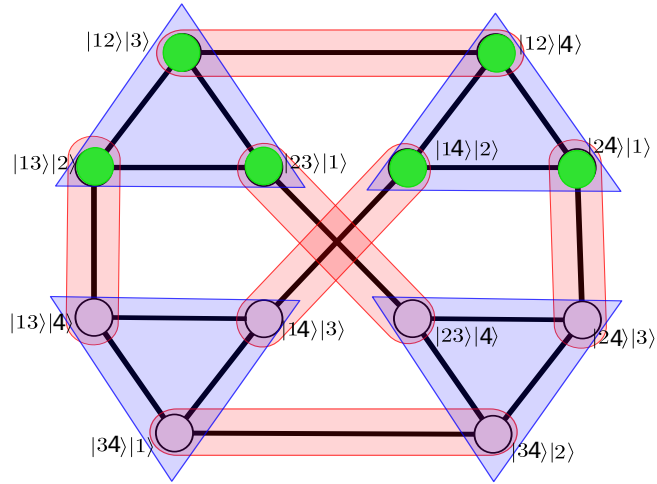


Figura 4.5: Grafo construído segundo as Definição 4.1 e Definição 4.8 após a aplicação do operador U_1 no estado quântico $|\Psi_1\rangle$

4.2.1 Corretude e Complexidade do Algoritmo de *Element k-Distinctness*

Vamos mostrar que nosso algoritmo é equivalente ao algoritmo de Ambainis através das proposições a seguir. Com isso, mostraremos a corretude e a complexidade do Algoritmo 4.

Proposição 4.12 *O estado inicial do algoritmo de Ambainis (Algoritmo 1) e o estado inicial do Algoritmo 4 apresentado são equivalentes.*

Demonstração. As duas primeiras etapas de ambos os algoritmos são exatamente as mesmas, com isso para uma mesma entrada, ambos os algoritmos terão uma mesma superposição inicial. \square

Proposição 4.13 *A aplicação das etapas de passeio quântico do algoritmo apresentado é equivalente a aplicação das etapas de passeio quântico apresentados no Algoritmo 2.*

Demonstração. Como descrito anteriormente, as três primeiras etapas do Algoritmo 2 fazem o caminhante se mover da partição A para a partição A' . Enquanto isso, as três últimas etapas do mesmo algoritmo movimentam o caminhante da partição A' para a partição A . (i) Vamos mostrar que o mapeamento executado na etapa 1 do Algoritmo 2 é equivalente à aplicação do operador U_0 , e (ii) vamos mostrar que as 5 etapas seguintes do Algoritmo 2 são equivalentes à aplicação das três últimas etapas do algoritmo proposto.

Vamos mostrar primeiro a equivalência em (i) aplicando primeiro o mapeamento da etapa 1 do Algoritmo 2 sobre um estado genérico $|\Psi_0\rangle = |S_i, y_i, x_i\rangle$ e posteriormente comparar com a aplicação do operador U_0 sobre o mesmo estado $|\Psi_0\rangle$. O mapeamento da etapa 1 do Algoritmo 2 faz

$$|S\rangle|y\rangle \rightarrow |S\rangle \left(\left(-1 + \frac{2}{N-r} \right) |y\rangle + \frac{2}{N-r} \sum_{\substack{y' \notin S \\ y' \neq y}} |y'\rangle \right), \quad (4.13)$$

para todo $S \in \mathcal{S}$ e $y \notin S$. Dado um estado genérico $|\Psi_0\rangle = |S_i, y_i, x_i\rangle \in \mathcal{H}$, a aplicação do mapeamento em $|\Psi_0\rangle$ leva ao estado

$$\left(\left(-1 + \frac{2}{N-r} \right) |S_i, y_i, x_i\rangle + \frac{2}{N-r} \sum_{\substack{y'_i \notin S_i \\ y'_i \neq y_i}} |S_i, y'_i, x_i\rangle \right). \quad (4.14)$$

Aplicando o operador U_0 sobre o mesmo estado genérico $|\Psi_0\rangle$ temos a operação

$$\left(\frac{2}{N-r} \left[\sum_{S \in \mathcal{S}} \sum_{y \notin S} \left(|S, y\rangle \langle S, y| + \sum_{\substack{y' \in S \\ y' \neq y}} |S, y'\rangle \langle S, y| \right) \right] - I \right) |S_i, y_i, x_i\rangle \quad (4.15)$$

que nos leva ao estado

$$\left(\left(-1 + \frac{2}{N-r} \right) |S_i, y_i, x_i\rangle + \frac{2}{N-r} \sum_{\substack{y'_i \notin S_i \\ y'_i \neq y_i}} |S_i, y'_i, x_i\rangle \right), \quad (4.16)$$

que é exatamente o estado descrito pela Equação(4.14). Portanto, o mapeamento presente na etapa 1 do Algoritmo 2 possui efeito equivalente à aplicação do operador U_0 .

Vamos agora mostrar a equivalência descrita em (ii), aplicando sobre um estado $|\Psi_1\rangle = |S_i, y_i, x_i\rangle$ as 5 etapas restantes presentes no Algoritmo 2, e comparar com o

resultado da aplicação das três últimas etapas do algoritmo proposto sobre o mesmo estado $|\Psi_1\rangle$.

No Algoritmo 2, a segunda etapa adiciona o valor presente no registrador $|y\rangle$ no registrador $|S\rangle$, ou seja, o valor de y_i será adicionado ao conjunto representado por S_i , levando-nos ao estado

$$|S_i \cup \{y_i\}, y_i, x_i\rangle.$$

A terceira etapa modifica o estado $|x\rangle$ a partir da inserção do valor x_y . Isto nos leva ao estado

$$|S_i \cup \{y_i\}, y_i, x_i \cup \{x_{y_i}\}\rangle.$$

Na etapa 4 aplicamos o mapeamento

$$|S\rangle|y\rangle \rightarrow |S\rangle \left(\left(-1 + \frac{2}{r+1} \right) |y\rangle + \frac{2}{r+1} \sum_{\substack{y' \in S \\ y' \neq y}} |y'\rangle \right), \quad (4.17)$$

que sobre o estado descrito anteriormente, nos leva ao estado

$$\left(-1 + \frac{2}{r+1} \right) |S_i \cup \{y_i\}, y_i, x_i \cup \{x_{y_i}\}\rangle + \frac{2}{r+1} \sum_{\substack{y'_i \in S_i \\ y'_i \neq y_i}} |S \cup \{y_i\}, y'_i, x_i \cup \{x_{y_i}\}\rangle. \quad (4.18)$$

A quinta etapa irá apagar do registrador $|x\rangle$ o valor de x_y , sendo y o valor no registrador $|y\rangle$. Isto leva-nos ao estado

$$\left(-1 + \frac{2}{r+1} \right) |S_i \cup \{y_i\}, y_i, x_i\rangle + \frac{2}{r+1} \sum_{\substack{y'_i \in S_i \\ y'_i \neq y_i}} |S \cup \{y_i\}, y'_i, x_i \cup \{x_{y_i}\} \setminus \{x_{y'_i}\}\rangle. \quad (4.19)$$

A sexta etapa irá remover os respectivos valores de y que foram inseridos nos respectivos registradores $|S\rangle$, presente em cada uma das componentes que compõem o estado corrente. Com isso obtemos o estado

$$\left(-1 + \frac{2}{r+1} \right) |S_i \cup \{y_i\} \setminus \{y_i\}, y_i, x_i\rangle + \frac{2}{r+1} \sum_{\substack{y'_i \in S_i \\ y'_i \neq y_i}} |S_i \cup \{y_i\} \setminus \{y'_i\}, y'_i, x_i \cup \{x_{y_i}\} \setminus \{x_{y'_i}\}\rangle, \quad (4.20)$$

que resulta em

$$\left(-1 + \frac{2}{r+1} \right) |S_i, y_i, x_i\rangle + \frac{2}{r+1} \sum_{\substack{y'_i \in S_i \\ y'_i \neq y_i}} |S'_i, y'_i, x'_i\rangle, \quad (4.21)$$

onde $S'_i = S_i \cup \{y_i\} \setminus \{y'_i\}$ e $x'_i = x_i \cup \{x_{y_i}\} \setminus \{x_{y'_i}\}$.

Enquanto isso, sobre um mesmo estado genérico $|\Psi_1\rangle$, ao aplicarmos a etapa 2 do algoritmo proposto temos o estado

$$|S_i, y_i, x_i \cup \{x_{y_i}\}\rangle. \quad (4.22)$$

Após isso, a aplicação do operador U_1 sobre o estado descrito anteriormente faz com que tenhamos a operação

$$\left(\frac{2}{r+1} \left[\sum_{S \in \mathcal{S}} \sum_{y \notin S} \left(|S, y\rangle \langle S, y| + \sum_{\substack{y' \in S \\ y' \neq y}} |S', y'\rangle \langle S, y| \right) \right] - I \right) |S_i, y_i, x_i \cup \{x_{y_i}\}\rangle, \quad (4.23)$$

onde $S' = S \cup \{y\} \setminus \{y'\}$, que faz com que tenhamos o estado

$$\left(-1 + \frac{2}{r+1} \right) |S_i, y_i, x_i\rangle + \frac{2}{r+1} \sum_{\substack{y'_i \in S_i \\ y'_i \neq y_i}} |S'_i, y'_i, x_i \cup \{x_{y_i}\}\rangle, \quad (4.24)$$

onde $S'_i = S_i \cup \{y_i\} \setminus \{y'_i\}$. Finalmente, a aplicação da etapa 4 do passeio quântico do algoritmo proposto faz com que tenhamos o estado

$$\left(-1 + \frac{2}{r+1} \right) |S_i, y_i, x_i\rangle + \frac{2}{r+1} \sum_{\substack{y'_i \in S_i \\ y'_i \neq y_i}} |S'_i, y'_i, x'_i\rangle, \quad (4.25)$$

onde $x'_i = x_i \cup \{x_{y_i}\} \setminus \{x_{y'_i}\}$, que é exatamente o estado descrito pela Equação(4.21). Logo a aplicação do operador U_1 e a aplicação das 5 últimas etapas do Algoritmo 2 são também equivalentes.

Com isso, mostramos que aplicar as etapas de passeio quântico do algoritmo proposto é equivalente à aplicar as etapas descritas pelo Algoritmo 2. □

Proposição 4.14 [37] *Toda instância de caminhada quântica no modelo Escalonado com duas tesselações que possua apenas um único vértice em comum em cada interseção de polígonos de tesselações distintas pode ser convertida em uma caminhada quântica no modelo de Szegedy.*

Proposição 4.15 *Um grafo G construído pela Definição 4.1 e tesselado segundo a Definição 4.8 sempre tem no máximo um único vértice na interseção de dois polígonos de tesselações distintas.*

Demonstração. Suponha por contradição que G tenha pelo menos dois vértices v e v' , em uma interseção de dois polígonos, chamados de α_i na tesselação α e β_j na

tesselação β .

Então, v e v' possuem a mesma vizinhança dentro do polígono α_i e do polígono β_j . Seja $v = (S, y)$ e $v' = (S', y')$. Pela Definição 4.1 e Definição 4.8, se v e v' estão no mesmo polígono α_i , então temos que (i) $S = S'$. Pela Definição 4.8 temos também que se v e v' estão no mesmo polígono β_i , então temos que (ii) $S' = S \cup \{y\} \setminus \{y'\}$ ou $S = S' \cup \{y'\} \setminus \{y\}$, levando-nos a uma contradição com (i). \square

Corolário 4.16 *O passeio quântico executado sobre o grafo G construído pela Definição 4.1 e tesselado segundo a Definição 4.8 pode ser convertido em um passeio quântico no modelo de Szegedy.*

Demonstração. Pelo Corolário 4.7 e pela Proposição 4.6 temos que o grafo G construído pela Definição 4.1 e tesselado segundo a Definição 4.8 pode ser convertido em um grafo bipartido. Pela Proposição 4.15 temos que o grafo G possui apenas um único vértice na interseção de dois polígonos de tesselações distintas. Logo pela Proposição 4.14 temos que o passeio quântico executado sobre o grafo G pode ser convertido em um passeio quântico no modelo de Szegedy. \square

Teorema 4.17 *Algoritmo 4 precisa de $O(N^{k/(k+1)})$ passos de passeio quântico para encontrar uma k -colisão, caso exista.*

Demonstração. Pelo Corolário 4.16 podemos converter o passeio quântico no modelo Escalonado sobre o grafo G em um passeio quântico no modelo de Szegedy sobre um grafo bipartido. Pela Proposição 4.12 temos que o algoritmo de Ambainis e o Algoritmo 4 possuem estados iniciais equivalentes, e pela Proposição 4.13 temos que as operações unitárias de evolução de ambos os algoritmos são equivalentes. Com isso, os estados finais de ambos os algoritmos serão também equivalentes.

Logo, se o algoritmo de Ambainis necessita de $O(N^{k/(k+1)})$ passos, o Algoritmo 4 também necessita de $O(N^{k/(k+1)})$ passos. \square

Pelas proposições acima, vemos que o algoritmo apresentado neste trabalho possui a mesma complexidade assintótica da abordagem de Ambainis, no entanto a proposta trás um resultado teórico importante para a computação quântica visto que utilizamos um modelo de passeio quântico mais geral.

Capítulo 5

Conclusão

Neste trabalho abordamos assuntos em dois paradigmas distintos: clássico e quântico. No paradigma clássico, abordamos o problema de cobertura por tesselações em grafos, enquanto no paradigma quântico propomos uma reformulação do algoritmo quântico para o problema de *Element k -Distinctness*, utilizando um novo modelo de passeios quânticos, conhecido como modelo escalonado [35, 37, 38].

O problema de cobertura por tesselações em grafos consiste em cobrir todas as arestas de um grafo utilizando tesselações, em que cada tesselação é um conjunto de cliques disjuntas cuja união cobre todos os vértices do grafo. Portugal [38] mostrou que os grafos 2-tesseláveis são aqueles cujo grafo clique é bipartido, isto é, 2-colorível. No entanto, não podemos afirmar que se um grafo possuir um número de tesselação $T(G) > 2$, o número cromático de seu grafo clique será igual a $T(G)$. Neste contexto propusemos um limite superior justo para o problema, sendo assim temos que o número de tesselação de qualquer grafo é tal que $T(G) \leq \chi(K(G))$. Além disso apresentamos duas famílias de grafos que possuem o número de tesselação bem definido em relação ao número cromático de seus respectivos grafos clique. A primeira família é a dos grafos *Moinho-de-vento*, que possui um número de tesselação $T(G) = \chi(K(G))$. A segunda família de grafos apresentada foi a dos grafos roda w -wheel, para $w \geq 5$, cujo número de tesselação é $T(G) = \lceil \frac{\chi(K(G))}{2} \rceil$. Também foi mostrado que se o número cromático do grafo clique for exatamente 3 então o grafo original é 3-tesselável.

Para o paradigma quântico, propusemos uma reformulação do algoritmo para o problema de *Element k -Distinctness*, utilizando o modelo Escalonado de passeio quântico. Este problema foi abordado por Ambainis com a proposta de executar um passeio quântico em um grafo bipartido de Johnson, reduzindo o problema de *Element k -Distinctness* a encontrar um vértice marcado no grafo em $O(N^{k/k+1})$ passos. Em nossa abordagem utilizamos um grafo 2-tesselável para executar a busca pelo vértice marcado. Em nossa proposta, conseguimos uma complexidade de tempo equivalente a complexidade obtida pela abordagem de Ambainis. Pelo fato de o

grafo utilizado em nosso algoritmo ser 2-tesselável, além de que o mesmo possui apenas um único vértice na interseção entre polígonos de tesselações distintas, faz com que seja possível convertê-lo em um grafo bipartido, que é exatamente o grafo utilizado pelo algoritmo de Ambainis. Além do mais, os estados iniciais, bem como os operadores quânticos de evolução utilizados por ambas as propostas torna possível a equivalência entre os dois algoritmos.

Ainda existem diversas questões em aberto a serem abordadas a partir deste trabalho. Alguns problemas em que a comunidade científica possui interesse podem, talvez, possuir alguma relação com o problema de tesselações em grafos, como por exemplo, os conhecidos problemas de cobertura de conjuntos (*set cover*) e o problema de cobertura por cliques (*clique cover*). Devido a isso, em trabalhos futuros desejamos abordar problemas que possam ter relação com o problema de cobertura por tesselações em grafos, e também buscar novos parâmetros, além do número cromático do grafo clique, que possam ter relação com o número de tesselação de um grafo. Um trabalho que se encontra em andamento é demonstrar que grafos livres de triângulos possuem um número de tesselação igual ao índice cromático do próprio grafo, como conjecturado neste trabalho, além de mostrar que todo grafo da família de triângulos de Sierpinski, gerado através de uma geração finita $g \geq 1$, é 3-tesselável. Outra conjectura em que estamos trabalhando é que o problema de tesselações em grafos é NP-completo. Sendo assim, desejamos propor um algoritmo aproximativo eficiente para o problema de tesselações. Outra questão importante é responder se existe um limite inferior não trivial para o número de tesselação de grafos em geral em função de algum parâmetro conhecido de grafos. Quanto a parte deste trabalho referente ao paradigma quântico, a abordagem apresentada abre margem para novas abordagens aplicando o modelo Escalonado em outros algoritmos quânticos conhecidos baseados em passeios quânticos, como por exemplo o algoritmo para encontrar triângulos em grafos.

Referências Bibliográficas

- [1] Aaronson, S., Shi, Y., 2004, “Quantum lower bounds for the collision and the element distinctness problems”, *Journal of the ACM (JACM)*, v. 51, n. 4, pp. 595–605.
- [2] Abreu, A. S., Kowada, L. A. B., Marquezino, F. L., 2016, “Staggered quantum algorithm for element distinctness”, *Submetido para Journal of Communication and Information Systems*.
- [3] Abreu, A. S., Cunha, L. F. I., Fernandes, T. D., et al., 2016, “On clique graphs and the tessellation problem”, Trabalho submetido ao IX Latin and American Algorithms, Graphs and Optimization Symposium (LAGOS), .
- [4] Abreu, A. S., Ferreira, M. M., Kowada, L. A. B., et al., 2016, “QEDS: A classical simulator for quantum element distinctness”, *Revista de Informática Teórica e Aplicada*, v. 23, n. 2, pp. 51–66.
- [5] Abreu, A. S., Cunha, L. F. I., Kowada, L. A. B., et al., 2017, “Bounds on the number of tessellations in graphs”, Trabalho submetido ao VII Latin American Workshop on Cliques in Graphs (LAWCG).
- [6] Aharonov, D., Ambainis, A., Kempe, J., et al., 2001, “Quantum walks on graphs”. In: *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pp. 50–59. ACM.
- [7] Aharonov, Y., Davidovich, L., Zagury, N., 1993, “Quantum random walks”, *Physical Review A*, v. 48, n. 2, pp. 1687.
- [8] Ambainis, A., 2007, “Quantum Walk Algorithm for Element Distinctness”, *SIAM Journal on Computing*, v. 37, n. 1, pp. 210—239.
- [9] Ambainis, A., 2004, “Quantum query algorithms and lower bounds”. In: *Classical and New Paradigms of Computation and their Complexity Hierarchies*, Springer, pp. 15–32.
- [10] Ambainis, A., 2003, “Quantum walks and their algorithmic applications”, *International Journal of Quantum Information*, v. 1, n. 04, pp. 507–518.

- [11] Ambainis, A., Bach, E., Nayak, A., et al., 2001, “One-dimensional quantum walks”. In: *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pp. 37–49. ACM.
- [12] Belovs, A., 2012, “Learning-Graph-Based Quantum Algorithm for k-Distinctness”. In: *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on*, pp. 207–216.
- [13] Belovs, A., Childs, A. M., Jeffery, S., et al., 2013, “Automata, Languages, and Programming: 40th International Colloquium, ICALP 2013, Riga, Latvia, July 8-12, 2013, Proceedings, Part I”. cap. Time-Efficient Quantum Walks for 3-Distinctness, pp. 105–122, Berlin, Heidelberg, Springer.
- [14] Berg, H. C., 1993, *Random walks in biology*. Princeton University Press.
- [15] Buhrman, H., De Wolf, R., 2002, “Complexity measures and decision tree complexity: a survey”, *Theoretical Computer Science*, v. 288, n. 1, pp. 21–43.
- [16] Buhrman, H., Dürr, C., Heiligman, M., et al., 2005, “Quantum algorithms for element distinctness”, *SIAM Journal on Computing*, v. 34, n. 6, pp. 1324–1330.
- [17] Childs, A. M., Gosset, D., Webb, Z., 2013, “Universal computation by multi-particle quantum walk”, *Science*, v. 339, n. 6121, pp. 791–794.
- [18] de Souza, D. S., 2016, *QUANDOOP: Um simulador para caminhadas quânticas com apoio de computação paralela e distribuída*. Tese de Mestrado, Universidade Federal do Rio de Janeiro.
- [19] Falk, M., 2013, “Quantum search on the spatial grid”, *arXiv preprint arXiv:1303.4127*.
- [20] Feynman, R. P., 1982, “Simulating physics with computers”, *International journal of theoretical physics*, v. 21, n. 6, pp. 467–488.
- [21] Figueiredo, C. M. H., 2017. “Comunicação privada”. Fevereiro.
- [22] Grady, L., 2006, “Random walks for image segmentation”, *IEEE transactions on pattern analysis and machine intelligence*, v. 28, n. 11, pp. 1768–1783.
- [23] Grover, L. K., 1996, “A fast quantum mechanical algorithm for database search”. In: *Proceedings of the twenty-eighth annual ACM symposium on Theory of Computing*, pp. 212–219. ACM.

- [24] Koreas, D. P., 1997, “The NP-completeness of chromatic index in triangle free graphs with maximum vertex of degree 3”, *Applied mathematics and computation*, v. 83, n. 1, pp. 13–17.
- [25] Kutin, S., 2005, “Quantum Lower Bound for the Collision Problem with Small Range.” *Theory of Computing*, v. 1, n. 1, pp. 29–36.
- [26] Landau, D., Wang, F., 2002, “Determining the density of states for classical statistical models by a flat-histogram random walk”, *Computer physics communications*, v. 147, n. 1-2, pp. 674–677.
- [27] Lara, P. C. S., Portugal, R., Boettcher, S., 2013, “Quantum walks on Sierpinski gaskets”, *International Journal of Quantum Information*, v. 11, n. 08, pp. 1350069.
- [28] Leao, A. B., 2015, *Um novo simulador de alta performance de caminhadas quânticas*. Tese de Mestrado, Laboratório Nacional de Computação Científica.
- [29] Lima, E. L., 1996, *Algebra Linear, 2a. edição*. IMPA, Rio de Janeiro.
- [30] Magniez, F., Santha, M., Szegedy, M., 2007, “Quantum algorithms for the triangle problem”, *SIAM Journal on Computing*, v. 37, n. 2, pp. 413–424.
- [31] Marquezino, F. L., Portugal, R., 2008, “The QWalk simulator of quantum walks”, *Computer Physics Communications*, v. 179, n. 5, pp. 359–369.
- [32] Nielsen, M. A., Chuang, I. L., 2010, *Quantum computation and quantum information*. Cambridge university press.
- [33] Nosofsky, R. M., Palmeri, T. J., 1997, “An exemplar-based random walk model of speeded classification.” *Psychological review*, v. 104, n. 2, pp. 266.
- [34] Patel, A., Raghunathan, K., Rungta, P., 2005, “Quantum random walks do not need a coin toss”, *Physical Review A*, v. 71, n. 3, pp. 032347.
- [35] Portugal, R., Santos, R., Fernandes, T., et al., 2016, “The staggered quantum walk model”, *Quantum Information Processing*, v. 15, n. 1, pp. 85–101.
- [36] Portugal, R., de Oliveira, M., Moqadam, J., 2017, “Staggered quantum walks with Hamiltonians”, *Physical Review A*, v. 95, n. 1, pp. 012328.
- [37] Portugal, R., 2015, “Establishing the equivalence between Szegedy’s and coined quantum walks using the staggered model”, *Quantum Information Processing*, pp. 1–23.

- [38] Portugal, R., 2016, “Staggered quantum walks on graphs”, *Physical Review A*, v. 93 (Jun), pp. 062335.
- [39] Posner, D., 2017. “Comunicação privada”. Fevereiro.
- [40] Salamon, P., Sibani, P., Frost, R., 2002, *Facts, conjectures, and improvements for simulated annealing*. SIAM.
- [41] Santos, R. A. M., 2010, *Cadeias de Markov Quânticas*. Tese de Mestrado, Laboratório Nacional de Computação Científica, Petrópolis, Brasil.
- [42] Shor, P. W., 1994, “Algorithms for quantum computation: Discrete logarithms and factoring”. In: *Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on*, pp. 124–134. IEEE.
- [43] Szegedy, M., 2004, “Quantum speed-up of Markov chain based algorithms”. In: *Foundations of Computer Science, 2004. Proceedings. 45th Annual IEEE Symposium on*, pp. 32–41. IEEE.
- [44] Szwarcfiter, J. L., 2003, “A survey on clique graphs”. In: *Recent advances in algorithms and combinatorics*, Springer, pp. 109–136.

Índice Remissivo

- 2-tesselável, 6
- algoritmo
 - de Ambainis, 25
 - no modelo escalonado, 32
 - quântico, 2
- cobertura, 4
 - mínima, 4
- coloração, 6
- computação
 - quântica, 6
- Element Distinctness, 24
- emaranhamento, 9
- esfera de Bloch, 7
- grafo
 - clique, 5, 6
 - de Hajós, 6, 21
 - de Johnson, 24, 25
 - moinho-de-vento, 18
 - roda, 18
- limites, 17
- modelo
 - com moeda, 10
 - de Szegedy, 9, 10
 - escalonado, 10, 13
- NP-completude, 22
- operador de reflexão, 10, 11, 30
- passoio
 - aleatório, 1, 10
 - quântico, 2, 10
- polígono, 4, 13, 22
- postulados, 7
- produto
 - de Kronecker, 8
 - tensorial, 8
- qubit, 7
- simulação, 2
- tesselação, 4, 13