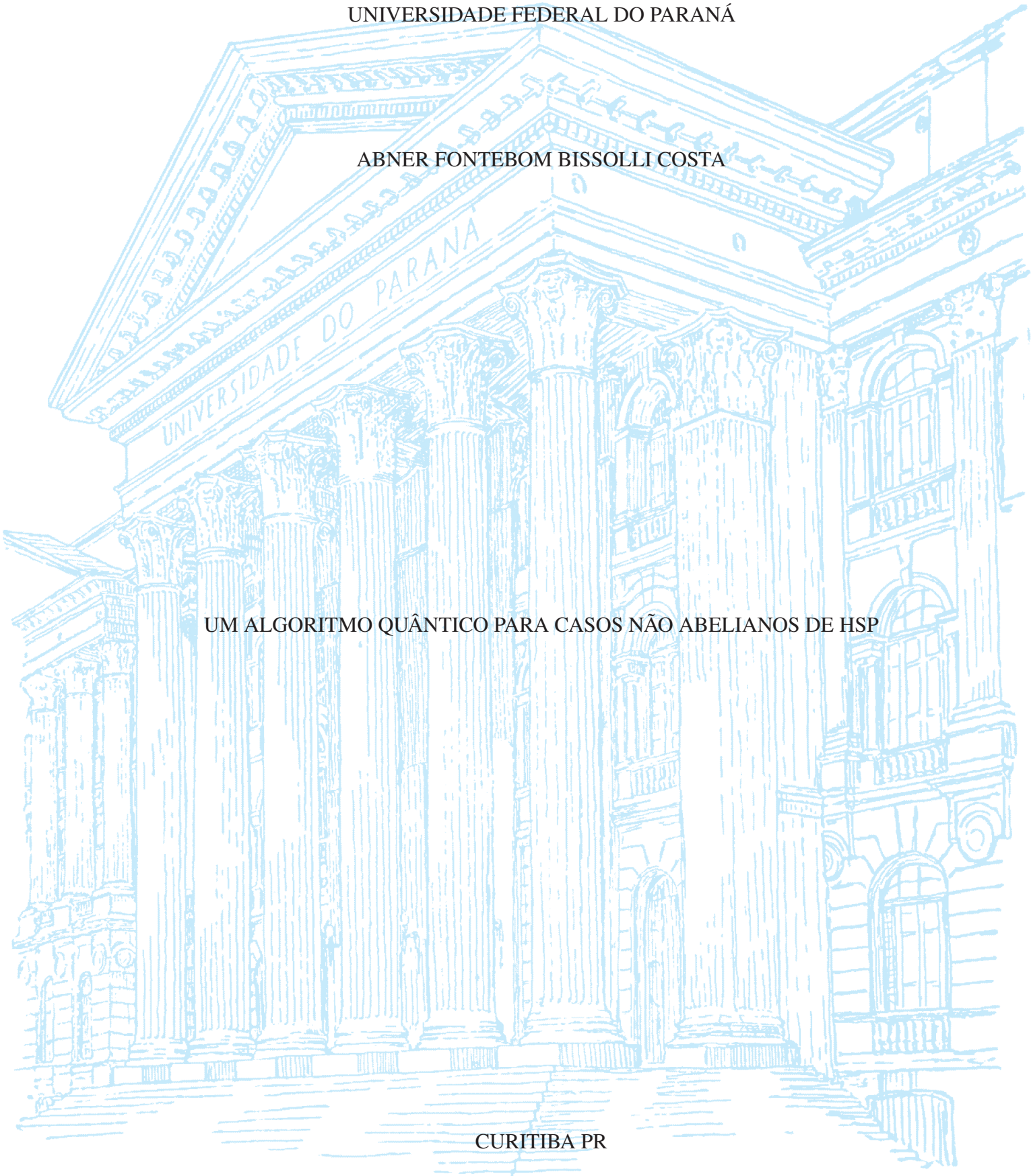
UNIVERSIDADE FEDERAL DO PARANÁ

ABNER FONTEBOM BISSOLLI COSTA



UM ALGORITMO QUÂNTICO PARA CASOS NÃO ABELIANOS DE HSP

CURITIBA PR

2024

ABNER FONTEBOM BISSOLLI COSTA

UM ALGORITMO QUÂNTICO PARA CASOS NÃO ABELIANOS DE HSP

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Informática no Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Computação*.

Orientador: Murilo V. G. da Silva.

Coorientador: Leandro M. Zatesko.

CURITIBA PR

2024

DADOS INTERNACIONAIS DE CATALOGAÇÃO NA PUBLICAÇÃO (CIP)
UNIVERSIDADE FEDERAL DO PARANÁ
SISTEMA DE BIBLIOTECAS – BIBLIOTECA DE CIÊNCIA E TECNOLOGIA

Costa, Abner Fontebom Bissolli
Um algoritmo quântico para casos não abelianos de HSP / Abner Fontebom Bissolli Costa. – Curitiba, 2024.
1 recurso on-line : PDF.

Dissertação (Mestrado) - Universidade Federal do Paraná, Setor de Ciências Exatas, Programa de Pós-Graduação em Informática.

Orientador: Murilo Vicente Gonçalves da Silva
Coorientador: Leandro Miranda Zatesko

1. Computação quântica. 2. Algoritmo quântico. I. Universidade Federal do Paraná. II. Programa de Pós-Graduação em Informática. III. Silva, Murilo Vicente Gonçalves da. IV. Zatesko, Leandro Miranda. V. Título.

Bibliotecário: Leticia Priscila Azevedo de Sousa CRB-9/2029

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação INFORMÁTICA da Universidade Federal do Paraná foram convocados para realizar a arguição da Dissertação de Mestrado de **ABNER FONTEBOM BISSOLLI COSTA** intitulada: **Um Algoritmo Quântico para casos não abelianos de HSP**, sob orientação do Prof. Dr. MURILO VICENTE GONÇALVES DA SILVA, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de mestre está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

CURITIBA, 30 de Novembro de 2023.

Assinatura Eletrônica
18/12/2023 15:36:38.0
MURILO VICENTE GONÇALVES DA SILVA
Presidente da Banca Examinadora

Assinatura Eletrônica
14/12/2023 10:17:45.0
RENATO JOSÉ DA SILVA CARMO
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica
20/12/2023 15:13:07.0
VINÍCIUS FERNANDES DOS SANTOS
Avaliador Externo (UNIVERSIDADE FEDERAL DE MINAS GERAIS)

Assinatura Eletrônica
13/12/2023 17:27:15.0
LEANDRO MIRANDA ZATESKO
Coorientador(a) (UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ)

A todos aqueles que este trabalho venha a contribuir e àqueles que tornaram possível sua existência.

AGRADECIMENTOS

Agradeço aos meus pais Romildo Costa e Bernardete Fontebom Fila Costa, que possibilitaram e apoiaram meus estudos, e ao Cali Sabino, que esteve presente e me apoiou em todos os momentos desta jornada, que me inspira em momentos difíceis e ilumina meu caminho quando estou perdido.

Sou grato aos meus amigos e familiares, que me proporcionaram uma rede de apoio fundamental, e contribuíram para que a pessoa que sou hoje pudesse existir.

Também agradeço ao meus professores, que sempre me instigaram a questionar e investigar, proporcionando uma vista sobre seus ombros gigantes. Em especial meus orientadores Murilo Vicente Gonçalves da Silva e Leandro Miranda Zatesko, que aturaram inúmeras discussões e teimosias de minha parte, e transmitiram da melhor forma possível seus conhecimentos para que eu possa realizar este trabalho.

De mesma maneira, agradeço aos meus colegas e amigos do grupo TEORIA, que me auxiliaram, instruíram e questionaram em etapas cruciais de meu mestrado.

Este trabalho foi realizado com apoio da Fundação Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).

RESUMO

O problema do subgrupo oculto, denominado HSP (de *Hidden Subgroup Problem)*, é um problema candidato à classe de problemas NP-intermediários. A versão de decisão de HSP, denominada dHSP (de *decision Hidden Subgroup Problem)*, é o problema de promessa de decidir se o subgrupo oculto é o subgrupo trivial ou não. Apresentamos uma análise de um algoritmo quântico de multiplicação de coclasses para dHSP, exibindo casos de grupos não abelianos que geram um estado próximo do estado maximamente misto, quando o subgrupo oculto não é o subgrupo trivial, e, por outro lado, geram um estado puro quando o subgrupo oculto é o subgrupo trivial. Adicionalmente, investigamos a relação de dHSP com classes de complexidade de conhecimento zero, apresentando casos particulares deste problema que estão em NIPZK_1, a classe de problemas que admitem provas não-interativas de conhecimento zero perfeito com completude perfeita.

Palavras-chave: Problema do Subgrupo Oculto. Não Abeliano. Conhecimento Zero Perfeito. Algoritmos Quânticos.

ABSTRACT

The Hidden Subgroup Problem (HSP) is a well-known candidate for NP-intermediate status. The decision version of HSP, called dHSP (short for Decision Hidden Subgroup Problem), is the promise problem of deciding whether the hidden subgroup is the trivial subgroup or not. We present an analysis of a coset multiplication quantum algorithm for dHSP, showing cases of non-abelian groups that the algorithm generate a state close to the maximally mixed state when the hidden subgroup is not the trivial subgroup, and, on the other hand, generate a pure state when the hidden subgroup is the trivial subgroup. Additionally, we investigate the relationship between dHSP and complexity classes of zero knowledge, presenting particular cases of this problem that are in NIPZK_1, the class of problems that admit non-interactive perfect zero-knowledge proofs with perfect completeness.

Keywords: Hidden Subgroup Problem. Non-Abelian. Perfect Zero-Knowledge. Quantum Algorithms.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ACRÔNIMOS

HSP	*Hidden Subgroup Problem*
dHSP	*Decision Hidden Subgroup Problem*
GI	*Graph Isomorphism Problem*
GA	*Graph Automorphism Problem*
#GA	*Counting Graph Automorphism Problem*
UN	*Uniform Problem*
US	*Uniform-or-Small Problem*
QFT	*Quantum Fourier Transform*
P	*Polynomial-Time Complexity*
PSPACE	*Polynomial-Space Complexity*
NP	*Non-Deterministic Polynomial-Time Complexity*
coNP	*Complement of Non-Deterministic Polynomial-Time Complexity*
SBP	*Small Bounded-Error Probability Complexity*
BPP	*Bounded-Error Probabilistic Polynomial-Time Complexity*
BQP	*Bounded-Error Quantum Polynomial-Time Complexity*
IP	*Interactive Proof Systems Complexity*
AM	*Arthur-Merlin Complexity*
coAM	*Complement of Arthur-Merlin Complexity*
SZK	*Statistical Zero Knowledge Complexity*
HVSZK	*Honest-Verifier Statistical Zero Knowledge Complexity*
PZK	*Perfect Zero Knowledge Complexity*
HVPZK	*Honest-Verifier Perfect Zero Knowledge Complexity*
NISZK	*Non-Interactive Statistical Zero Knowledge Complexity*
NIPZK	*Non-Interactive Perfect Zero Knowledge Complexity*
$NIPZK_1$	*Non-Interactive Perfect Zero Knowledge Complexity with perfect completeness*
CRS	*Common Reference String*

LISTA DE SÍMBOLOS

i	$\sqrt{-1}$
I_n	matriz identidade de dimensão $n \times n$
π	pi, também utilizado para representar uma permutação
$\sum$	símbolo matemático de uma sequência de somas
log	símbolo matemático de logaritmo na base 2
$\perp$	símbolo matemático de perpendicular
$\dagger$	símbolo matemático de conjugado transposto
$\otimes$	símbolo matemático de produto tensorial
$\in$	símbolo matemático de pertence
$\notin$	símbolo matemático de não pertence
$\subseteq$	símbolo matemático de está contido
$\nsubseteq$	símbolo matemático de não está contido
$\cong$	símbolo matemático de congruente
$\ncong$	símbolo matemático de não congruente
$\times$	símbolo matemático de produto cartesiano, também utilizado para representar as dimensões de uma matriz
$\exists$	símbolo matemático de existe
$\nexists$	símbolo matemático de não existe
$\forall$	símbolo matemático de para todo
$\cap$	símbolo matemático de intersecção de conjuntos
:	símbolo matemático de tal que
$\rightarrow$	símbolo matemático de mapeamento
$\iff$	símbolo matemático de se e somente se
$\circ$	símbolo matemático de composição
$\star$	símbolo estrela, utilizado para representar operação binária
$\lceil\rceil$	função matemática de teto
$\lfloor\rfloor$	função matemática de chão
$>$	símbolo matemático de maior que
$\geq$	símbolo matemático de maior ou igual que
$<$	símbolo matemático de menor que e de subgrupo próprio de
$\leq$	símbolo matemático de menor ou igual que, e de subgrupo de
$\not\leq$	símbolo matemático de não subgrupo de
$\trianglelefteq$	símbolo matemático de subgrupo normal de
$\mathbb{Z}$	conjunto dos números inteiros
$\mathbb{Z}_{>0}$	conjunto dos números inteiros positivos

$\mathbb{Z}_N$	conjunto dos números inteiros módulo N
$\mathbb{C}$	conjunto de números complexos
$\mathbb{C}_{\neq 0}$	conjunto de números complexos diferentes de 0
$\mathbb{S}_n$	grupo simétrico de permutações de n elementos
$\mathbb{A}_n$	grupo alternante de permutações de n elementos
$\mathbb{D}_n$	grupo diedral de simetrias de um polígono regular de n lados
$GL(V)$	grupo de matrizes invertíveis em um espaço vetorial V
$\mathbb{F}_q$	corpo finito de q elementos
e	elemento identidade de um grupo
σ	representação linear
χ	caracter de uma representação linear
$\mathcal{H}$	espaço de Hilbert
$\mathcal{G}$	grafo
$\mathcal{W}$	passeio aleatório
$\mathcal{U}$	espaço amostral
$\mathcal{O}$	oráculo
Pr	função de probabilidade
O	notação assintótica grande-O
poly	função limitada superiormente por uma função polinomial
exp	função limitada superiormente por uma função exponencial
Π	problema computacional
Π_Y	instâncias positivas de Π
Π_N	instâncias negativas de Π
μ	função negligenciável
Δ	função de distância estatística
L_1	função de distância L_1, distância de Manhattan
α	alfa, primeira letra do alfabeto grego
β	beta, segunda letra do alfabeto grego
Γ	gama, terceira letra do alfabeto grego em maiúsculo
ϵ	épsilon, quinta letra do alfabeto grego
κ	kappa, décima letra do alfabeto grego
φ	varphi, variante da vigésima primeira letra do alfabeto grego
ψ	psi, penúltima letra do alfabeto grego
ω	omega, última letra do alfabeto grego
$\|\psi\rangle$	vetor de números complexos
$\langle\psi\|$	vetor dual de $\|\psi\rangle$
$\langle\varphi\|\psi\rangle$	produto interno entre $\|\varphi\rangle$ e $\|\psi\rangle$
$\|\varphi\rangle\langle\psi\|$	produto externo entre $\|\varphi\rangle$ e $\|\psi\rangle$
ρ	matriz de densidade de um estado quântico

SUMÁRIO

1 INTRODUÇÃO

O Problema do Subgrupo Oculto (HSP, de *Hidden Subgroup Problem*) é um problema de teoria de grupos candidato a ser NP-intermediário. Um problema NP-intermediário, é um problema em NP que não é NP-completo, nem é decidido em tempo polinomial por uma máquina de Turing determinística. Em (Ladner, 1975) foi mostrado que se $\mathsf{P} \neq \mathsf{NP}$, então existem problemas NP-intermediários.

Grupos são estruturas algébricas que são compostas por um conjunto e uma operação binária, de modo que esta é associativa para os elementos do conjunto, e o conjunto contém tanto um elemento identidade quanto o inverso de todos os seus elementos. Teoria de grupos é um ramo da matemática estudado não somente por matemáticos, mas também físicos, cientistas da computação e químicos, e pode ser utilizada em física quântica, criptografia, teoria de jogos, simetria molecular e em diversas outras aplicações.

HSP é um problema que encapsula diversos problemas notáveis, como fatoração de inteiros, logaritmo discreto, isomorfismo de grafos e vetor mais curto em reticulados (Nielsen e Chuang, 2002; Regev, 2004), de modo que estes podem ser vistos como casos particulares de HSP. Este problema recebe um interesse particular, pois possui casos que podem ser solucionados por algoritmos quânticos em tempo polinomial, mas não tem solução clássica conhecida de tempo polinomial, como, por exemplo, fatoração de inteiros e logaritmo discreto (Shor, 1994). Muitas técnicas foram desenvolvidas desde Shor (1994) generalizando e aprimorando seus resultados (Kitaev, 1995; Hallgren et al., 2003; Grigni et al., 2004; Radhakrishnan et al., 2005).

Muitos dos casos solucionados por algoritmos quânticos são aqueles onde o grupo em questão é abeliano, isto é, um grupo onde sua a operação é comutativa para seus elementos, ou então quando o subgrupo oculto é próximo de ser um subgrupo normal. Entretanto, HSP possui diversos outros casos, não abelianos, para os quais não se conhece algoritmo polinomial clássico ou quântico. Um exemplo é quando o problema é sobre grupos de permutação, que parece ser um caso especialmente difícil de HSP, onde as técnicas desenvolvidas até então têm se mostrado insuficientes (Moore e Russell, 2005). Ainda, técnicas de amostragem de Fourier, utilizadas em Shor (1994); Kitaev (1995), parecem não trazer vantagem alguma comparadas com algoritmos clássicos de busca exaustiva (Kempe e Shalev, 2004).

A dificuldade de encontrarmos algoritmos polinomiais para o caso de grupos de permutação de HSP não é surpreendente, uma vez que uma solução polinomial resultaria em um algoritmo polinomial para o problema de isomorfismo de grafos (Nielsen e Chuang, 2002), para o qual o melhor algoritmo conhecido possui custo computacional quasipolinomial (Babai, 2016). Uma vez que muitas técnicas conhecidas não se aplicam para HSP, voltamos nosso estudo para a versão de decisão do problema do subgrupo oculto (dHSP, de *decision Hidden Subgroup Problem*). Dado que não sabemos se dHSP é tão difícil quanto HSP, decidir se o subgrupo oculto é o subgrupo trivial ou não pode ser computacionalmente mais fácil que encontrar o subgrupo oculto, no entanto, resolver dHSP também tem se mostrado uma tarefa difícil, mesmo para algoritmos quânticos (Kempe e Shalev, 2004).

Neste trabalho, apresentamos um análise sobre um algoritmo quântico de multiplicação de coclasses do subgrupo oculto. Estas análises visam buscar uma alternativa à abordagem de Shor e a amostragem de Fourier para o problema. Para realizar estar análises, consideramos um algoritmo quântico no qual obtemos sobreposições dos elementos de coclasses aleatórias do subgrupo oculto, em seguida combinamos estas diferentes coclasses, via operação do grupo, para obter uma sobreposição de elementos de G. Por meio de nossas análises, foi possível estabelecer

que este algoritmo quântico, para alguns casos particulares de grupos não abelianos, gera um estado quântico cuja distância de traço, em relação ao estado maximamente misto, é próxima de 1 quando o subgrupo oculto é o subgrupo trivial, e distante de 1 caso contrário. Além dos casos particulares apresentados, conjecturamos que quando o grupo em questão é o grupo alternante de permutações, resultados mais fortes podem ser enunciados, apresentando alguns indícios para a conjectura levantada.

Ainda, estudamos a relação de dHSP com classes de provas interativas de conhecimento zero, apresentando casos particulares de dHSP que admitem provas de conhecimento zero perfeito com completude perfeita, classe NIPZK_1 (i.e. NIPZK com completude perfeita). Para a classe NIPZK_1, também conseguimos mostrar que esta admite um problema completo, que é uma versão restrita do problema de promessa completo para a classe NIPZK.

A Figura 1.1 apresenta as classes de complexidade estudadas neste trabalho, juntamente com as relações de inclusão atualmente conhecidas entre elas.

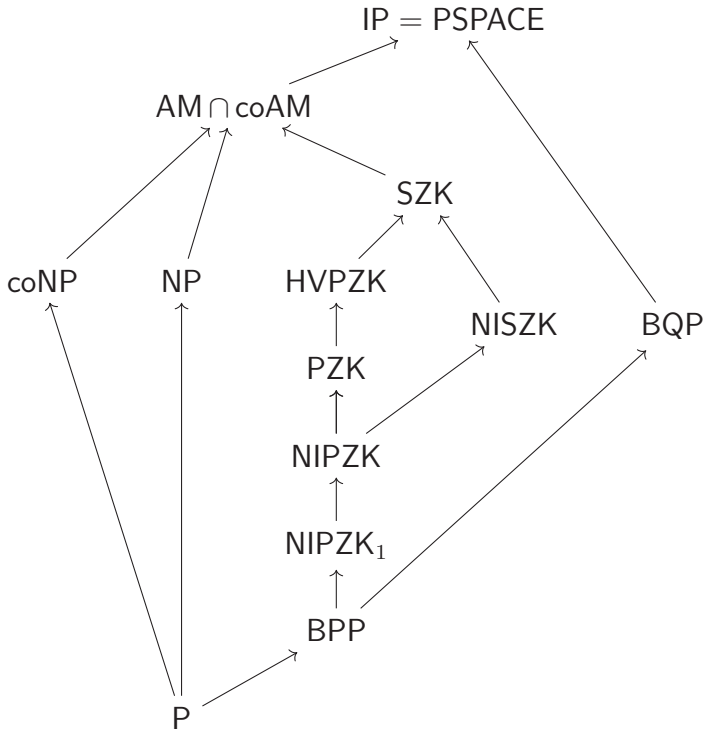


Figura 1.1: Relação conhecida entre as classes de complexidade discutidas neste trabalho. Onde a seta $\mathsf{A} \rightarrow \mathsf{B}$ representa que $\mathsf{A} \subseteq \mathsf{B}$.

Organizamos este trabalho de modo que, no Capítulo 2, apresentamos conceitos e resultados referentes às áreas de computação quântica, complexidade computacional, teoria de grupos, grupos de permutações e teoria de representação. No Capítulo 3, apresentamos a definição do problema do subgrupo oculto e de sua versão de decisão, assim como revisamos os principais algoritmos quânticos eficientes para HSP. Nossos resultados originais são apresentados no Capítulo 4 e nossas considerações finais no Capítulo 5.

2 FUNDAMENTOS

Dedicamos este capítulo a apresentar definições, conceitos, análises e resultados já estabelecidos na literatura que são fundamentais para a apresentação deste trabalho. Nenhum dos resultados apresentados neste capítulo é original do autor desta dissertação, e as demonstrações dos teoremas, lemas e proposições enunciados podem ser encontradas nas referências bibliográficas fornecidas. Para o entendimento do texto, assumimos que o leitor possui familiaridade com teoria de grafos e teoria da complexidade computacional, para os quais utilizamos os livros *Introduction to graph theory* (West et al., 2001) e *Computational complexity: a modern approach* (Arora e Barak, 2009). Neste trabalho assumimos que todos os grafos são simples (não direcionados, não ponderados e sem arestas múltiplas ou laços), a menos que explicitado o contrário. A menos que seja explicitamente indicado de outra forma, os logaritmos (log) são de base 2.

Na Seção 2.1, apresentamos conceitos de computação quântica, adaptados de Nielsen e Chuang (2002). A Seção 2.2 contempla as classes de complexidade utilizadas neste trabalho, estabelecendo conceitos de teoria de complexidade computacional utilizados, para qual a referência principal é Arora e Barak (2009). As Seções 2.3, 2.4 e 2.5, são dedicadas a apresentar conceitos de teoria de grupos, grupos de permutação e teoria de representação, respectivamente.

2.1 COMPUTAÇÃO QUÂNTICA

Nesta seção apresentamos alguns conceitos básicos de computação quântica que servem de alicerce para o entendimento do texto, estabelecendo representações e nomenclaturas, assim como algumas análises que são utilizadas ao longo do documento. As análises presentes nesta seção foram adaptadas do livro *Quantum Computation and Quantum Information* (Nielsen e Chuang, 2002). Para as representações de mecânica quântica utilizadas neste trabalho, referenciamos a Tabela 2.1, adaptada de Nielsen e Chuang (2002).

Tabela 2.1: Resumo de algumas representações padrão da mecânica quântica para conceitos da álgebra linear .

Notação	Descrição
z^*	Conjugado complexo do número complexo z.
	$(1+i)^* = 1-i$
$\|\psi\rangle$	Vetor de números complexos. Também conhecido como um *ket*.
$\langle\psi\|$	Vetor dual de $\|\psi\rangle$. Também conhecido como um *bra*.
$\langle\varphi\|\psi\rangle$	Produto interno entre os vetores $\|\varphi\rangle$ e $\|\psi\rangle$.
$\|\varphi\rangle\langle\psi\|$	Produto externo entre os vetores $\|\varphi\rangle$ e $\|\psi\rangle$.
$\|\varphi\rangle \otimes \|\psi\rangle$	Produto tensorial de $\|\varphi\rangle$ e $\|\psi\rangle$.
$\|\varphi\rangle\|\psi\rangle$	Notação abreviada para o produto tensorial de $\|\varphi\rangle$ e $\|\psi\rangle$.
I_n	Matriz identidade de dimensão $n \times n$.
A^*	Conjugado complexo da matriz A.
A^T	Transposto da matriz A.
$A^\dagger$	Transposto conjugado da matriz A, $A^\dagger = (A^T)^*$.
	$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^\dagger = \begin{bmatrix} a^* & c^* \\ b^* & d^* \end{bmatrix}$.
$\langle\varphi\|A\|\psi\rangle$	Produto interno entre $\|\varphi\rangle$ e $A\|\psi\rangle$.
	Equivalentemente, produto interno entre $A^\dagger\|\varphi\rangle$ e $\|\psi\rangle$.

Frequentemente nos referimos a *espaços de Hilbert*, que são espaços de vetores complexos com produto interno. Esses espaços desempenham um papel fundamental na compreensão da mecânica quântica e, por conseguinte, na computação quântica, uma vez que os utilizamos para representar estados quânticos.

Para estabelecer algumas premissas fundamentais de nosso estudo, utilizamos os seguintes postulados da mecânica quântica, os quais nos fornecem fundamentos essenciais para o estudo da computação quântica.

Postulado 1. Todo sistema físico isolado está associado a um espaço de Hilbert $\mathcal{H}$, conhecido por *espaço de estados* do sistema. O sistema é completamente descrito por um vetor de *estado (puro)*, um vetor unitário em $\mathcal{H}$.

Consideraremos apenas espaços de Hilbert de dimensão finita. Um sistema de n qubits, para $n \in \mathbb{Z}_{>0}$, é um sistema quântico em um espaço de Hilbert de dimensão 2^n. Na maioria dos casos, ao fixarmos uma base ortonormal $\{|i\rangle\}_{i=0}^{2^n-1}$ para o espaço de Hilbert $\mathcal{H}$ de dimensão 2^n, representamos um sistema de n qubits $|\psi\rangle \in \mathcal{H}$ por

$$|\psi\rangle = \sum_{i=0}^{2^n-1} \alpha_i \, |i\rangle \, ,$$

onde $\alpha_0, \cdots, \alpha_{2^n-1} \in \mathbb{C}$ e $\sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1$. A base fixada $\{|i\rangle\}_{i=0}^{2^n-1}$ é chamada de *base computacional* para n qubits. Neste texto iremos assumir que a base utilizada é a base computacional, a menos que seja explicitamente indicado o contrário. Às vezes, por abuso, consideramos os elementos da base computacional tanto como inteiros entre 0 e $2^n - 1$, quanto como elementos de $\{0,1\}^n$, associando cada inteiro a uma palavra binária de n bits que o representa.

Seja $\{|x\rangle\}_x$ estados de uma base de um espaço de Hilbert, uma *sobreposição* desses estados é um sistema quântico ao qual é associada uma combinação linear de $\{|x\rangle\}_x$.

Postulado 2. A evolução de um sistema quântico *fechado* é representado por uma *transformação unitária*. Isto é, um estado $|\psi\rangle$ de um sistema no tempo t_1 é relacionado a um estado $|\psi'\rangle$ do sistema no tempo t_2 por um operador unitário U que depende somente de t_1 e t_2,

$$|\psi'\rangle = U \, |\psi\rangle \, .$$

Postulado 3. Medições quânticas são descritas por uma coleção de operadores lineares $\{M_i\}$ chamados de *operadores de medição*, que atuam sobre o espaço de estados do sistema sendo medido e satisfazem a *equação de completude* $\sum_i M_i^\dagger M_i = I$, onde i é um índice referente aos possíveis resultados da medição.

Se o estado de um sistema quântico é $|\psi\rangle$ imediatamente antes da medição, então a probabilidade do resultado i acontecer é dada por

$$\Pr_i[|\psi\rangle] = \langle\psi| \, M_i^\dagger M_i \, |\psi\rangle$$

e o estado quântico do sistema imediatamente após a medição é

$$\frac{M_i |\psi\rangle}{\sqrt{\langle\psi| M_i^\dagger M_i |\psi\rangle}}$$

Ao realizar uma medição na base computacional a partir de um estado arbitrário $|\psi\rangle$, podemos calcular a probabilidade de obter o resultado i da seguinte forma:

$$\Pr_i[|\psi\rangle] = |\langle i|\psi\rangle|^2.$$

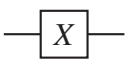
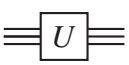
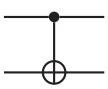
Essa abordagem simplifica o cálculo das medições na base computacional e será utilizada quando conveniente.

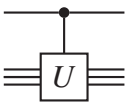
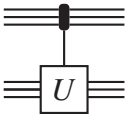
Postulado 4. O espaço de estados de um sistema físico composto é o produto tensorial dos espaços de estados dos sistemas físicos componentes. Se os sistemas forem numerados de 1 a n, onde o sistema i está no estado $|\psi_i\rangle$, então o estado conjunto do sistema total é $|\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_n\rangle$.

Um *circuito quântico* é uma rede acíclica de portas quânticas que age sobre qubits.

Na Tabela 2.2 apresentamos alguns dos operadores quânticos utilizados nesta dissertação, juntamente com sua representação matricial e seu respectivo diagrama de circuito quântico.

Tabela 2.2: Diagrama e representação matricial de operadores quânticos.

Descrição	Circuito	Notação e Matriz
Registrador de único qubit		
Registrador de múltiplos qubits		
Haddamard		$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Pauli-X (NOT)		$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Operador Unitário Arbitrário		$U = \begin{bmatrix} u_{00} & u_{01} & \cdots & u_{0n} \\ u_{11} & u_{11} & \cdots & u_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ u_{n1} & u_{n1} & \cdots & u_{nn} \end{bmatrix}$
NOT Controlado		$cNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

U Controlado		$cU = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & u_{00} & \cdots & u_{0n} \\ 0 & 0 & \vdots & \ddots & \vdots \\ 0 & 0 & u_{n1} & \cdots & u_{nn} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & U \end{bmatrix}$
U Controlado por n qubits		$cU = \begin{bmatrix} I_{2^n} & 0 \\ 0 & U \end{bmatrix}$

Note que o Postulado 1 refere-se a sistemas físicos isolados. Em casos onde o sistema não é isolado e não pode ser completamente descrito por um vetor unitário $|\psi\rangle$ em $\mathcal{H}$, dizemos que o mesmo está em um *estado misto*. Para exemplificar, considere o estado $|\psi_{AB}\rangle = \frac{1}{\sqrt{2}}|0_A 0_B\rangle + \frac{1}{\sqrt{2}}|1_A 1_B\rangle$, composto por dois qubits A e B. Observe que não conseguimos descrever um único qubit deste sistema por um vetor unitário em um espaço de Hilbert $\mathcal{H}$ de dimensão dois. Deste modo, dizemos que os sistemas quânticos compostos por um único qubit deste sistema são, de fato, estados mistos.

Para representar um estado misto podemos utilizar um conjunto $\{(p_i, |\psi_i\rangle)\}$, de estados puros $|\psi_i\rangle$, onde i é um índice, associados a suas respectivas probabilidades p_i. No caso do exemplo apresentado, podemos descrever os estados quânticos dos qubits através do conjunto $\{(1/2, |0\rangle), (1/2, |1\rangle)\}$.

Uma maneira de representar estados, puros ou mistos, é através de *operadores de densidade*, os quais são representados matematicamente por *matrizes de densidade*.

Definição 2.1 (Matriz de Densidade)**.** Uma matriz de densidade ρ é representação matemática de um sistema quântico associado a um espaço de Hilbert $\mathcal{H}$, onde ρ é uma matriz semidefinida positiva de traço 1. Se o sistema quântico é um estado puro $|\psi\rangle$, então $\rho = |\psi\rangle\langle\psi|$, por outro lado, se é um estado misto $\{(p_i, |\psi_i\rangle)\}$, estão $\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$.

Se uma matriz ρ representa um estado puro, então a *pureza* deste estado é 1, $\mathcal{P}(\rho) = \mathrm{tr}(\rho^2) = 1$, caso contrário $\mathcal{P}(\rho) < 1$. Deste modo, matrizes de densidade podem ser utilizadas para distinguir estados mistos de estados puros.

Por meio da Definição 2.1 podemos reformular os postulados da mecânica quântica apresentados para realizar análises utilizando matrizes de densidade.

Postulado 1. Todo sistema físico isolado está associado a um espaço de Hilbert $\mathcal{H}$, conhecido por *espaço de estados* do sistema. O sistema é perfeitamente descrito pela matriz de densidade ρ em $\mathcal{H}$. Se um sistema quântico está no estado ρ_i, com probabilidade p_i, então a matriz de densidade do sistema é $\sum_i p_i \rho_i$.

Postulado 2. A evolução de um sistema quântico *fechado* é descrito por uma *transformação unitária*. Isto é, um estado ρ de um sistema no tempo t_1 está relacionado a um estado ρ' do sistema no tempo t_2 por um operador unitário U que depende somente de t_1 e t_2,

$$\rho' = U\rho U^\dagger.$$

Postulado 3. Medições quânticas são descritas por uma coleção $\{M_i\}$ de *operadores de medição*, que atuam sobre o espaço de estados do sistema sendo medido e satisfazem a *equação de completude* $\sum_i M_i^\dagger M_i = I$, onde i é um índice referente aos possíveis resultados da medição.

Se o estado de um sistema quântico é ρ imediatamente antes da medição , então a probabilidade do resultado i acontecer é dada por

$$\Pr_i[\rho] = \mathrm{tr}(M_i^\dagger M_i \rho),$$

e o estado quântico do sistema é

$$\frac{M_i \rho M_i^\dagger}{\mathrm{tr}(M_i^\dagger M_i \rho)}.$$

Postulado 4. O espaço de estados de um sistema físico composto é o produto tensorial dos espaços de estados dos sistemas físicos componentes. Se os sistemas forem numerados de 1 a n, onde o sistema i está no estado ρ_i, então o estado conjunto do sistema total é $\rho_1 \otimes \rho_2 \otimes \cdots \otimes \rho_n$.

Quando estamos trabalhando com sistemas de múltiplos qubits, muitas vezes desejamos analisar o estado de apenas parte deste sistema. Seja ρ^{AB} o estado de um sistema composto de dois registradores A e B. Para propriamente analisar o estado de um destes registradores, é necessário obter a *matriz de densidade reduzida* do registrador desejado. O estado do registrador A no sistema no sistema ρ^{AB}, representado pela matriz de densidade reduzida ρ^A, pode ser obtido computando o *traço parcial* de ρ^{AB} referente ao registrador B.

$$\rho^A = \mathrm{tr}_B(\rho^{AB}).$$

onde $\mathrm{tr}_B(|a_1\rangle\langle a_2| \otimes |b_1\rangle\langle b_2|) = |a_1\rangle\langle a_2|\,\mathrm{tr}(|b_1\rangle\langle b_2|)$ e $|a_1\rangle$ e $|a_2\rangle$ são quaisquer dois vetores do estado do espaço de A, e $|b_1\rangle$ e $|b_2\rangle$ são quaisquer dois vetores do estado do espaço de B. Ainda, se $\rho^{AB} = \rho_1 \otimes \rho_2$, onde ρ_1 é o estado do sistema A e ρ_2 o estado do sistema B, então $\rho^A = \mathrm{tr}_B(\rho^{AB}) = \rho_1 \mathrm{tr}(\rho_2) = \rho^A$ e, similarmente, $\rho^B = \rho_2$. Tomemos como exemplo o estado de Bell $\left(\frac{|00\rangle+|11\rangle}{\sqrt{2}}\right)$, sua matriz de densidade é dada por

$$\begin{aligned}\rho &= \left(\frac{|00\rangle + |11\rangle}{\sqrt{2}}\right)\left(\frac{\langle 00| + \langle 11|}{\sqrt{2}}\right) \\ &= \frac{|00\rangle\langle 00| + |11\rangle\langle 00| + |11\rangle\langle 00| + |11\rangle\langle 11|}{2}.\end{aligned}$$

Realizando o traço parcial no segundo qubit obtemos

$$\begin{aligned}
\rho^1 &= \mathrm{tr}_2(\rho) \\
&= \frac{\mathrm{tr}_2(|00\rangle\langle 00|) + \mathrm{tr}_2(|11\rangle\langle 00|) + \mathrm{tr}_2(|11\rangle\langle 00|) + \mathrm{tr}_2(|11\rangle\langle 11|)}{2} \\
&= \frac{|0\rangle\langle 0|\,\mathrm{tr}(|0\rangle\langle 0|) + |1\rangle\langle 0|\,\mathrm{tr}(|1\rangle\langle 0|) + |0\rangle\langle 1|\,\mathrm{tr}(|0\rangle\langle 1|) + |1\rangle\langle 1|\,\mathrm{tr}(|1\rangle\langle 1|)}{2} \\
&= \frac{|0\rangle\langle 0|\langle 0|0\rangle + |1\rangle\langle 0|\langle 0|1\rangle + |0\rangle\langle 1|\langle 1|0\rangle + |1\rangle\langle 1|\langle 1|1\rangle}{2} \\
&= \frac{|0\rangle\langle 0|\,|1\rangle\langle 1|}{2} \\
&= \frac{I}{2}.
\end{aligned}$$

Note que o estado $\frac{I}{2}$ é um estado misto, uma vez que $\mathrm{tr}((\frac{I}{2})^2) \leq 1$, de fato, $\frac{I}{2}$ é o estado maximamente misto em um sistema de dois qubits, pois $\mathcal{P}(\rho) = \frac{1}{2}$. Para um sistema de n qubits o estado maximamente misto é $\frac{I}{2^n}$ e sua pureza é $\mathcal{P}(\rho) = \frac{1}{2^n}$.

Matrizes de densidade ainda podem ser usadas para determinarmos a distância entre dois estados quântico, uma métrica comumente utilizada é a distância do traço.

Definição 2.2 (Distância de Traço)**.** A distância do traço entre dois estados ρ e ρ' é dada por

$$D(\rho, \rho') = \frac{1}{2}\mathrm{tr}\left(\sqrt{(\rho - \rho')(\rho - \rho')^\dagger}\right).$$

A distância de traço é uma generalização distância de Kolmogorov e da distância L_1, também possuindo a propriedade muito utilizada é da desigualdade triangular.

Proposição 2.3 (Propriedade de Desigualdade Triangular da Distancia de Traço)**.** Sejam ρ, ρ', ρ'' matrizes de densidade, então

$$D(\rho, \rho') \leq D(\rho, \rho'') + D(\rho'', \rho')$$

2.2 CLASSES DE COMPLEXIDADE

Nesta seção, apresentamos definições e resultados relacionados a teoria de complexidade computacional, para os quais foi utilizado o livro *Computational complexity: a modern approach* (Arora e Barak, 2009) como referência principal. Assumimos que o leitor tem familiaridade com assuntos como máquina de Turing determinística e máquina de Turing probabilística.

Iniciamos apresentando duas definições de máquinas de Turing especiais, *máquina de Turing com conselho* e *máquina de Turing com oráculo*, que possuem propriedades adicionais a uma máquina de Turing convencional.

Definição 2.4 (Máquina de Turing com conselho)**.** Uma máquina de Turing M com *conselho* $[a] = \{a_n\}_{n\in\mathbb{Z}_{>0}}$, onde $a_n \in \{0,1\}^*$, denotada $M_{[a]}$, é uma máquina de Turing que na entrada x, tem como uma segunda entrada o conselho $a_{|x|}$.

Uma vez que o tamanho da palavra de conselho não é limitado, o tempo de execução de uma máquina de Turing com conselho $M_{[a]}$ é dado em função to tamanho da entrada x.

Definição 2.5 (Máquina de Turing com oráculo)**.** Uma *máquina de Turing com oráculo* é uma máquina de Turing M que possui uma fita de leitura e escrita especial, chamada de *fita do oráculo*, e três estados especiais $q_{\texttt{consulta}}$, $q_{\texttt{sim}}$ e $q_{\texttt{não}}$. Para executar M, é preciso especificar um linguagem $O \subseteq \{0,1\}^*$, além da entrada x, que é usada como o *oráculo* para M. Sempre que M entra no estado $q_{\texttt{consulta}}$, a máquina move para o estado $q_{\texttt{sim}}$ se $q \in O$ e $q_{\texttt{não}}$ se $q \notin O$, onde q representa o conteúdo da fita do oráculo. Note que cada consulta a O conta como um único passo computacional. Se M é uma máquina de oráculo, $O \subseteq \{0,1\}^*$ uma linguagem, e $x \in \{0,1\}^*$, representamos a saída de M com oráculo O sobre a entrada x por $M^O(x)$.

Enxergando o oráculo O, que é uma linguagem, como um problema computacional, permitindo analisar o poder computacional de máquinas de Turing que conseguem resolver, em um único passo computacional, o problema O. Em alguns casos, o problema computacional O é um problema que contempla a dificuldade de toda uma *classe de complexidade*, tais problemas são chamados de *problemas completos* para a classe de complexidade em questão. Estes problemas são discutidos em detalhes nesta seção. Desta maneira, é importante notar que as análises que utilizam uma máquina de Turing com oráculo podem levar a resultados muito distintos se comparadas com análises que utilizam máquinas de Turing convencionais, como em relações entre classes de complexidade que não refletem a relação dessas classes sem o uso de oráculos.

Usualmente, em teoria de complexidade computacional, definimos problemas de decisão como linguagens sobre $\{0,1\}^*$. No entanto, nesta dissertação, vamos trabalhar com problemas de promessa, onde um problema de promessa Π é um par de conjuntos disjuntos (Π_Y, Π_N) tal que $\Pi_Y, \Pi_N \subseteq \{0,1\}^*$, onde Π_Y corresponde a instâncias $\texttt{sim}$ e Π_N a instâncias $\texttt{não}$. Problemas de promessa são problemas mais genéricos que problemas de decisão, sendo convenientes no estudo de classes de complexidade que envolvem conceitos como conhecimento zero e computação quântica. Dado $x \in \Pi_Y \cup \Pi_N$, o problema de decidir se $x \in \Pi_Y$ ou $x \in \Pi_N$ é dito um problema de promessa, onde strings em $\Pi_Y \cup \Pi_N$ são chamadas de instâncias de Π. Dado um algoritmo que decida um problema de promessa Π, não há nenhum requerimento quanto ao seu comportamento caso seja fornecida como entrada uma string que não pertença a $\Pi_Y \cup \Pi_N$. O problema complementar de problema de promessa $\Pi = (\Pi_Y, \Pi_N)$ é o problema de promessa $\overline{\Pi} = (\overline{\Pi}_Y, \overline{\Pi}_N)$, onde $\overline{\Pi_Y} = \Pi_N$ e $\overline{\Pi_N} = \Pi_Y$.

Dizemos que um problema de promessa T é polinomialmente Karp redutível para um problema de promessa Π, denotado por $T \leq_p \Pi$, se existe uma função $f : \{0,1\}^* \to \{0,1\}^*$ computável em tempo polinomial por uma máquina de Turing, de modo que se $x \in T_Y$, então $f(x) \in \Pi_Y$, e se $x \in T_N$, então $f(x) \in \Pi_N$. Se $T \leq_p \Pi$ e $\Pi \leq_p T$, então dizemos que os problemas T e Π são polinomialmente equivalentes, denotado $T \equiv_p \Pi$.

Uma *classe de complexidade* é um conjunto de funções que podem ser computadas dentro de determinados limites de recursos, como tempo e espaço. Um problema Π pertence a uma classe de complexidade C se Π pode ser decidido dentro dos limites de recursos de C.

A classe de complexidade computacional P (*Polynomial-Time*) é a classe de problemas de promessa $\Pi = (\Pi_Y, \Pi_N)$ que são decididos por um maquina de Turing determinística M de tempo polinomial. A classe P usualmente é definida como uma classe de linguagem, mas abusamos da definição de problemas de promessa, uma vez que nesta dissertação voltamos nossas discussões para problemas de promessa. Definir uma classe de complexidade como sendo uma

classe de problemas promessa ao invés de uma classe de linguagem, pode possuir diferenças nos resultados obtidos. Enquanto a classe P, definida como uma classe de linguagem é fechada sob complemento, não é garantido que a classe P, definida como uma classe de problemas de promessa, também seja. Este abuso de notação também é empregado ao definirmos a classe NP a seguir. Neste trabalho, discutimos apenas classes de complexidade que contêm P, portanto reduções Karp polinomiais são adequadas em nosso contexto. No entanto, salientamos que para outras classes de complexidade, pode ser necessário utilizar diferentes tipos de reduções.

A classe de complexidade computacional NP (*Non-Deterministic Polynomial-Time*) é a classe de problemas de promessa $\Pi = (\Pi_Y, \Pi_N)$ para os quais existe um polinômio $p : \mathbb{Z}_{>0} \to \mathbb{Z}_{>0}$ e uma máquina de Turing determinística M de tempo polinomial, chamada de *verificador*, tal que se $x \in \Pi_Y$, então M `aceita` (x, u) para alguma string $u \in \{0,1\}^{p(|x|)}$ e se $x \in \Pi_N$ então M `rejeita` (x, u) para toda string $u \in \{0,1\}^{p(|x|)}$. Se $x \in \Pi_Y$ e M `aceita` uma string $u \in \{0,1\}^{p(|x|)}$, então chamamos u de *certificado* para a string x em relação ao problema de promessa Π e à máquina de Turing M.

A classe de complexidade computacional coNP é definida como o conjunto de problemas cujos complementos estão contidos na classe de complexidade NP, isto é $\mathsf{coNP} = \{\overline{\Pi} : \Pi \in \mathsf{NP}\}$. De maneira geral, dada uma classe de complexidade C, a classe de complexidade coC é definida como o conjunto de problemas cujos complementos estão contidos na classe de complexidade C, isto é $\mathsf{coC} = \{\overline{\Pi} : \Pi \in \mathsf{C}\}$.

Dizemos que um problema de promessa Π é NP-difícil (difícil para NP) se todo problema $T \in \mathsf{NP}$ é polinomialmente Karp redutível para Π. Se Π é NP-difícil e $\Pi \in \mathsf{NP}$, então dizemos que Π é NP-completo (completo para NP).

Um problema $\Pi \in \mathsf{NP}$ é considerado NP-intermediário se ele não pertence a P e não é um problema NP-completo. Em outras palavras, um problema NP-intermediário é mais difícil do que os problemas em P, mas não tão difícil quanto os problemas NP-completos.

A existência de problemas NP-intermediários é uma questão fundamental em teoria de complexidade computacional, pois sua existência ou não tem implicações importantes para a relação entre as classes de complexidade P e NP Trivialmente, se existem problemas NP-intermediários, então $\mathsf{P} \neq \mathsf{NP}$. Por outro lado, o Teorema de Ladner é um resultado que estabelece que se $\mathsf{P} \neq \mathsf{NP}$, então existem problemas NP-intermediários.

Teorema 2.6 (Ladner, 1975)**.** Se $\mathsf{P} \neq \mathsf{NP}$, então existem problemas NP-intermediários.

Uma vez que ainda não se sabe se as classes P e NP são iguais ou diferentes, não conhecemos nenhum problema NP-intermediário. Apesar disso, muitos problemas são candidatos a serem NP-intermediários, como o problema de fatoração de inteiros, isomorfismo de grafos e logaritmo discreto.

Com exceção da classe P, para todas as classes discutidas nesta dissertação podemos definir noções de completudes (problemas difíceis e completos) de maneira similar à maneira como definimos para a classe NP. Seja C uma classe de problemas de promessa, dizemos que o problema de promessa Π é C-difícil (difícil para C) se todo problema $T \in \mathsf{C}$ é polinomialmente Karp redutível para Π. Se Π é C-difícil e $\Pi \in \mathsf{C}$, então dizemos que Π é C-completo (completo para C).

Observe que esta definição de completude pode não ser adequada quando estamos trabalhando com classes de complexidade que estão contidas em P, um exemplo é que, se definido deste modo, qualquer problema em P seria P-completo. Para o estudo de classes de complexidade que estão contidas em P é necessário utilizar outro tipo de redução, mais fracas do que os limites dos recursos da classe em questão. Uma vez que as classes de complexidade discutidas nesta dissertação contém P, a definição apresentada é adequada.

A classe de complexidade computacional BPP (*Bounded-Error Probabilistic Polynomial-Time*) é a classe dos problemas $\Pi = (\Pi_Y, \Pi_N)$ para os quais existe um máquina de Turing probabilística M de tempo polinomial tal que se $x \in \Pi_Y$, então a probabilidade de M `aceitar` x é maior ou igual a $\frac{2}{3}$ e se $x \in \Pi_N$, então a probabilidade de M `aceitar` x é menor ou igual a $\frac{1}{3}$.

A classe de complexidade computacional PSPACE (*Polynomial-Space*) é a classe de problemas de promessa $\Pi = (\Pi_Y, \Pi_N)$ que são decididos por um maquina de Turing determinística M de espaço polinomial.

A Figura 2.1 apresenta as relações conhecidas entre as classes complexidade apresentadas até então.

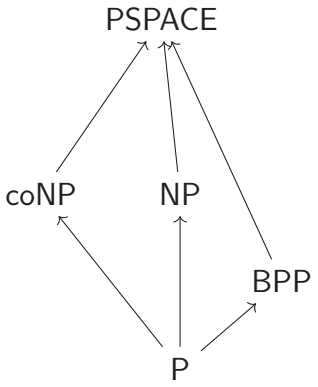


Figura 2.1: Relação conhecida entre as classes de complexidade P, BPP, NP, coNP e PSPACE. A seta $\mathsf{A} \rightarrow \mathsf{B}$ representa que $\mathsf{A} \subseteq \mathsf{B}$.

Apresentamos agora a classe de complexidade quântica BQP. Quando dizemos que um problema de promessa Π é decidido eficientemente por um computador quântico, ou admite algoritmo quântico eficiente, significa que $\Pi \in$ BQP.

Sendo $S \in \{0,1\}^*$, uma família $\mathcal{Q} = \{Q_x : x \in S\}$ de circuitos quânticos é *gerada uniformemente em tempo polinomial* se existe uma máquina de Turing determinística polinomial que, para toda entrada $x \in S$, devolve uma descrição de Q_x. Dado um polinômio p, se cada circuito da família $\mathcal{Q}$ possui só um qubit de saída, dizemos que $\mathcal{Q}$ aceita x se, ao medirmos o estado resultante de $Q_x |0\rangle^{\otimes p(|x|)}$, obtemos 1.

A classe de complexidade BQP (*Bounded-Error Quantum Polynomial-Time*) é a classe dos problemas $\Pi = (\Pi_Y, \Pi_N)$ que admitem uma família uniforme de circuitos quânticos $\mathcal{Q} = \{Q_x : x \in \{0,1\}^*\}$, onde cada circuito Q_x possui só um qubit de saída, de modo que se $x \in \Pi_Y$, então a probabilidade de Q aceitar x é de pelo menos $\frac{2}{3}$ e se $x \in \Pi_N$ então a probabilidade de Q aceitar x é de no máximo $\frac{1}{3}$.

O Lema 2.7 é um resultado importante para relacionar BQP com outras classes de complexidade clássicas.

Lema 2.7 (Circuitos booleanos como uma subclasse de circuitos quânticos)**.** Se $f : \{0,1\}^n \rightarrow \{0,1\}^m$ é computável por um circuito booleano de tamanho S, então existe um circuito quântico Q de tamanho $2S+m+n$ tal que $Q |x\rangle |0^m\rangle \left|0^{m+S}\right\rangle = |x\rangle |f(x)\rangle \left|0^{S+m}\right\rangle$, para todo $x \in \{0,1\}^n$. □

Observe que se S é polinomial em n, então o circuito quântico também é de tamanho polinomial em n.

Corolário 2.8. BPP $\subseteq$ BQP □

A Figura 2.2 acrescenta a classe de complexidade BQP à relação de classes apresentada na Figura 2.1.

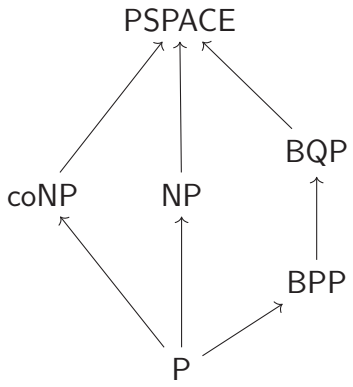


Figura 2.2: Relação conhecida entre as classes de complexidade apresentadas na Figura 2.1 e a classe BQP. A seta $\mathsf{A} \rightarrow \mathsf{B}$ representa que $\mathsf{A} \subseteq \mathsf{B}$.

Agora, apresentamos definições e resultados referentes a provas interativas e conhecimento zero, para os quais foram utilizados como referencia os textos (Vadhan, 1999; Goldreich et al., 1998; Sdroievski et al., 2019a), além de (Arora e Barak, 2009).

Definição 2.9 (Protocolo interativo)**.** Um protocolo interativo (P, V) é um par de funções P e V denominadas de *provador* e *verificador*, respectivamente. A interação entre P e V sobre uma string comum x é o processo aleatório a seguir, representado por $(P, V)(x)$.

1. P e V têm acesso a uma sequência infinita de bits aleatórios e uniformes r_P e r_V, respectivamente, da qual podem ler tantos bits quanto necessário.
2. Para i de 1 até o número de rodadas.
   (a) Se i é ímpar $m_i = P(x, m_1, \cdots, m_{i-1}; r_P)$.
   (b) Se i é par $m_i = V(x, m_1, \cdots, m_{i-1}; r_V)$.
   (c) Se $m_i \in \{\texttt{aceite}, \texttt{rejeite}, \texttt{pare}\}$, então termine.

Dizemos que um protocolo é polinomialmente limitado se, na entrada comum x, um número $\text{poly}(|x|)$ de mensagens são trocadas, sendo que cada mensagem têm tamanho $\text{poly}(|x|)$.

Definição 2.10 (Prova interativa)**.** Dizemos que um protocolo (P, V) é uma *prova interativa* para um problema de promessa $\Pi = (\Pi_Y, \Pi_N)$ se ele atende às três condições a seguir:

1. (*Eficiência*) (P, V) é polinomialmente limitado e V é computável em tempo polinomial;
2. (*Completude*) Se $x \in \Pi_Y$, então V aceita com probabilidade pelo menos $\frac{2}{3}$ na interação $(P, V)(x)$;
3. (*Corretude*) Se $x \in \Pi_N$, então para qualquer provador $P*$, V rejeita com probabilidade pelo menos $\frac{2}{3}$ na interação $(P*, V)(x)$.

A classe de complexidade IP (*Interactive Proof Systems*) é classe de problemas de promessa $\Pi = (\Pi_Y, \Pi_N)$, para os quais existe uma prova interativa (P, V).

Um resultado notável sobre a classe IP é o Teorema 2.11 a seguir, que estabelece uma relação entre provas interativas e máquinas de Turing de espaço polinomial.

Teorema 2.11 (Shamir, 1992; Lund et al., 1992)**.** IP = PSPACE. □

A classe de complexidade AM (*Arthur-Merlin*) é a classe de problemas de promessa $\Pi = (\Pi_Y, \Pi_N)$ para os quais existe uma prova interativa (P, V), na qual apenas duas mensagens são trocadas, a primeira sendo enviada de V (*Arthur*) para P (*Merlin*) e a segunda de P para V. Este modelo simula o processo onde *Arthur*, limitado, faz uma pergunta a *Merlin*, ilimitado, que responde com uma mensagem, então baseado na mensagem de *Merlin*, *Arthur* toma sua decisão ao problema em questão.

Uma vez que apenas duas trocas de mensagens são feitas, este modelo torna-se significativamente mais limitado que o anterior, no entanto ainda é possível observar que NP ⊆ AM.

Um caso particular de provas interativas que é fundamental nesta dissertação é o caso em que as provas são de *conhecimento zero*. Intuitivamente, uma prova interativa de conhecimento zero é uma prova interativa na qual o verificador não obtém nenhuma informação adicional, além do fato de que a instância do problema de promessa $\Pi = (\Pi_Y, \Pi_N)$ pertence a Π_Y.

Primeiramente, para podermos discutir provas de conhecimento zero, iremos apresentar alguns conceitos de probabilidade e estatística que são necessários para discutirmos tais provas, sendo também utilizados em outras partes deste trabalho.

Um *espaço amostral* $\mathcal{U}$ de um experimento aleatório, é o conjunto de todos os possíveis resultados deste um experimento. Uma *distribuição de probabilidade* X em um espaço amostral $\mathcal{U}$ é uma função $X : \mathcal{U} \to [0, 1]$ tal que $\sum_{s\in\mathcal{U}} X(s) = 1$. Uma distribuição X é dita *uniforme* em um espaço amostral finito $\mathcal{U}$ se $X(s) = 1/|\mathcal{U}|$ para todo $s \in \mathcal{U}$. Dependendo do contexto, podemos representar a distribuição uniforme de um espaço amostral $\mathcal{U}$ por $U_{\mathcal{U}}$ ou simplesmente U, quando o espaço amostral esta claro dado o contexto.

Uma *variável aleatória* R é um mapeamento do espaço amostral $\mathcal{U}$ para um conjunto T. A variável aleatória R junto da distribuição uniforme em $\mathcal{U}$ induz uma distribuição X em T. É comum, neste trabalho, nos referirmos a uma distribuição induzida por R por apenas R. Para um conjunto $S \subseteq T$ e uma variável aleatória R, definimos $\Pr[R \in S] = \sum_{r\in S} \Pr[R = r]$. O *suporte* de uma distribuição de probabilidade (ou variável aleatória) X, denotado $sup(X)$, é o conjunto de valores para os quais X possui uma probabilidade não nula, isto é, $sup(X) = \{x \in X : \Pr[X = x] \neq 0\}$.

Para compararmos duas distribuições de variáveis aleatórias (ou distribuições de probabilidade) X e Y definidas sobre o mesmo espaço amostral $\mathcal{U}$, utilizamos a *distância estatística*, denotada $\Delta(X, Y)$ e dada por

$$\Delta(X, Y) = max_{S\in\mathcal{U}}\{|\Pr[X \in S] - \Pr[Y \in S]|\}.$$

Uma outra medida comumente utilizada para distância de duas distribuições de probabilidade X e Y é a distância L_1, dada por

$$L_1(X, Y) = \sum_{s\in\mathcal{U}} |X(s) - Y(s)|.$$

Agora, para melhor definir a propriedade de conhecimento zero, é necessário entender como um verificador poderia aprender algo a mais. Seja (P, V) um protocolo interativo. A *visão* de V em relação à (P, V) na entrada comum x é a variável aleatória $\langle P, V(x)\rangle = (m_1, ..., m_t; r)$, que consiste em todas as mensagens $m_1, \cdots, m_t$ trocadas entre P e V junto da substring r de r_V que contém os bits aleatórios que V leu durante a interação.

Observe que $\langle P, V\rangle(x)$ contém toda informação que V teve acesso na interação $(P, V)(x)$.

Dizemos que uma máquina de Turing S é *útil* se ela falha, indicado pela saída `falha`, com probabilidade no máximo $\frac{1}{2}$ e denotamos por $\tilde{S}(x)$ a saída de S com a entrada x condicionada a S não falhar.

A princípio, consideramos apenas verificadores que seguem o protocolo estabelecido, desta maneira conseguimos definir as classes de complexidade HVSZK e HVPZK, definidas a seguir.

Definição 2.12 (Prova de Conhecimento Zero Estatístico para Verificadores Honestos)**.** Uma prova interativa para um problema de promessa $\Pi = (\Pi_Y, \Pi_N)$ é uma *prova de conhecimento zero estatístico para verificadores honestos*, se existe uma máquina de Turing probabilística útil S e uma função *negligenciável* μ tais que para todo $x \in \Pi_Y$ e $k > 0$

$$\Delta\big(\tilde{S}(x, 1^k), \langle P, V\rangle(x, 1^k)\big) \leq \mu(k).$$

onde uma função $\mu : \mathbb{Z}_{>0} \to [0, 1]$ é dita *negligenciável* se para todo polinômio p, temos $\mu(k) < 1/p(k)$ para k suficientemente grande. Chamamos a função negligenciável μ de *desvio do simulador* e o parâmetro 1^k de *parâmetro de segurança*.

Dizemos que uma prova de conhecimento zero estatístico é uma prova de conhecimento zero perfeito se $\mu = 0$. Desta maneira o desvio do simulador é 0 e as distribuições da simulação e da visão do verificador são idênticas.

A classe de complexidade HVSZK (*Honest-Verifier Statistical Zero Knowledge*) é classe de problemas de promessa $\Pi = (\Pi_Y, \Pi_N)$ que admitem provas interativas de conhecimento zero estatístico para verificadores honestos.

A classe de complexidade HVPZK (*Honest-Verifier Perfect Zero Knowledge*) é classe de problemas de promessa $\Pi = (\Pi_Y, \Pi_N)$ que admitem provas interativas de conhecimento zero perfeito estatístico para verificadores honestos.

Definimos agora classes de provas de conhecimento zero com verificadores desonestos.

Definição 2.13 (Prova de Conhecimento Zero Estatístico para Verificadores Desonestos)**.** Uma prova interativa para um problema de promessa $\Pi = (\Pi_Y, \Pi_N)$ é uma *prova de conhecimento zero estatístico*, se para todo algoritmo probabilístico polinomial V^* existe um algoritmo probabilístico polinomial útil S e uma função negligenciável μ tal que para todo $x \in \Pi_Y$, $k > 0$ e $a \in \{0, 1\}^*$

$$\Delta\big(\tilde{S_{[a]}}(x, 1^k), \langle P, V^*_{[a]}\rangle(x, 1^k)\big) \leq \mu(k).$$

onde $S_{[a]}$ e $V^*_{[a]}$, respectivamente, denotam as máquinas de Turing S e V^* com conselho $a \in \{0, 1\}^*$.

Dizemos que uma prova interativa de conhecimento zero estatístico é uma prova interativa de conhecimento zero perfeito se $\mu = 0$.

Desta maneira, garantimos que para qualquer verificador (honesto ou desonesto) o desvio do simulador será negligenciável, resultando que o verificador receba apenas a informação de que a instância é positiva, acompanhada de um erro estatístico. Se este erro estatístico é nulo, então o protocolo é de conhecimento zero perfeito e a única informação que o verificador recebe é que a instância é positiva.

Apesar de provas de conhecimento zero com verificadores desonestos parecerem muito distintas de provas de conhecimento zero com verificadores honestos, temos o seguinte teorema para conhecimento zero estatístico.

Teorema 2.14 (Vadhan1999)**.** HVSZK = SZK □

No entanto, apesar deste teorema, ainda não sabemos se HVPZK é igual a PZK, conhecendo apenas a relação trivial HVPZK $\subseteq$ PZK.

Além de modelos de provas interativas de conhecimento zero, existem provas *não interativas* de conhecimento zero, definidas a seguir

Definição 2.15 (Protocolos Não Interativos)**.** Um protocolo *não interativo* é um protocolo (P, V) no qual P e V compartilham uma *string de referência* (CRS, de *Common Reference String*) $r \in \{0,1\}^m$ escolhida uniformemente ao acaso e a única mensagem entre P e V é enviada pelo provador.

Similarmente a provas interativas, uma *prova não-interativa* é um protocolo não-interativo que atende às condições de eficiência, completude e corretude.

A classe de complexidade NISZK (*Non-Interactive Statistical Zero Knowledge*) é a classe de problemas de promessa $\Pi = (\Pi_Y, \Pi_N)$ que admitem provas não-interativas de conhecimento zero estatístico, similar à definição de SZK, porém o protocolo é não-interativo. De maneira similar, a classe de complexidade NIPZK (*Non-Interactive Perfect Zero Knowledge*) é a classe de problemas de promessa $\Pi = (\Pi_Y, \Pi_N)$ que admitem provas não-interativas de conhecimento zero perfeito.

Na definição original de provas interativas, para a condição de completude é necessário que o verificador aceite uma instância $x \in \Pi_Y$ de um problema de promessa $\Pi = (\Pi_Y, \Pi_N)$ com probabilidade de pelo menos $\frac{2}{3}$. Alterando esta definição, se exigimos que o verificador sempre aceite uma instância $x \in \Pi_Y$ (aceite com probabilidade 1), dizemos que esta é uma prova com completude perfeita.

A classe de classe de complexidade NIPZK$_1$ (*Non-Interactive Perfect Zero Knowledge with perfect completeness*) é a classe de problemas de promessa $\Pi = (\Pi_Y, \Pi_N)$ que admitem provas não-interativas de conhecimento zero perfeito com completude perfeita.

A Figura 2.3 apresenta a relação entre as classes de complexidade discutidas nesta seção.

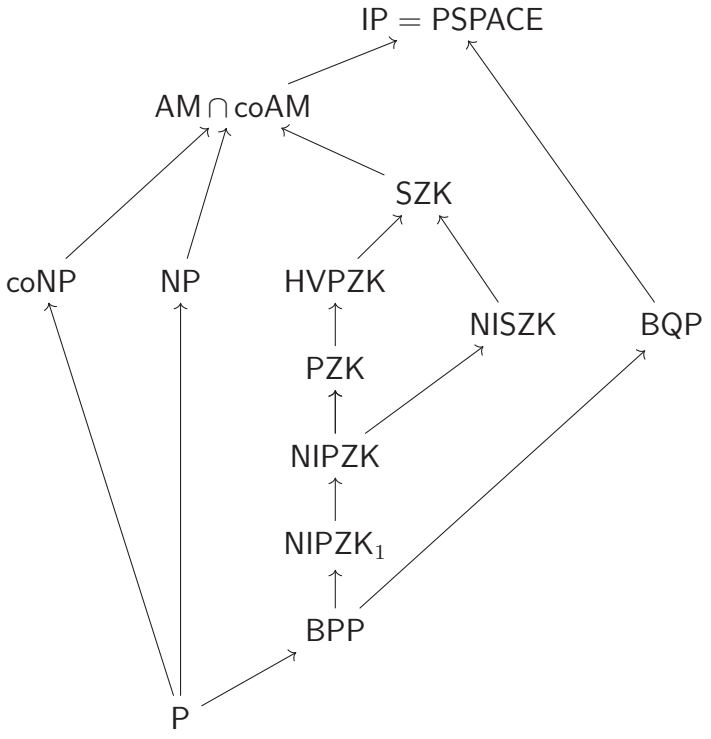


Figura 2.3: Relação conhecida entre as classes de complexidade discutidas na Seção 2.2. A seta A $\rightarrow$ B representa que A $\subseteq$ B.

2.3 TEORIA DE GRUPOS

Nesta seção apresentamos conceitos e análises relacionados à teoria de grupos. O conteúdo apresentado nesta seção foi retirado e adaptado do livro *A First Course in Abstract Algebra* (Rotman, 2005), que contempla teoria de grupos de maneira mais abrangente, uma vez que nesta proposta apenas utilizamos alguns conceitos básicos desse assunto.

Definição 2.16. Um *grupo* é um par $(G, *)$, onde G é um conjunto e $*$ é uma operação binária, tal que

1. A operação $*$ é associativa para os elementos do grupo.
2. Existe um único elemento $e \in G$, chamado de *identidade* ou *elemento neutro*, tal que $a * e = a$ para todo $a \in G$.
3. Para todo $a \in G$ existe um elemento $a^{-1} \in G$ tal que $a * a^{-1} = e$, onde a^{-1} é denominado de *inverso* de a.

Muitas vezes representamos um grupo $(G, *)$ por apenas G, quando a operação $*$ está implícita ou não é relevante para o contexto em questão.

Como exemplos de grupos, temos o grupo $(\mathbb{Z}_N, +)$, de adição módulo N e o grupo $(\mathbb{Z}_N, \times)$ de multiplicação módulo N.

A *ordem* de um grupo $(G, *)$ é a quantidade de elementos do conjunto G e é denotada por $|G|$. Um grupo G é um *grupo finito* se e somente se sua ordem é finita, caso contrário dizemos G é um *grupo infinito*. Dizemos que dois grupos $(G_1, *)$ e $(G_2, \star)$ são *isomorfos*, denotado $G_1 \cong G_2$, se existe uma bijeção $f : G_1 \to G_2$ tal que $f(a * b) = f(a) \star f(b)$ para todo $a, b \in G_1$. Logo, por definição, temos a se dois grupos $(G_1, *)$ e $(G_2, \star)$ são isomorfos, então $|G_1| = |G_2|$.

Neste trabalho consideramos que todos os grupos são finitos, a menos que seja explicitamente colocado o contrário. Para quaisquer dois subconjuntos $A, B \subseteq G$, denotamos por AB o conjunto $AB = \{a * b : a \in A, b \in B\}$.

Em muitos casos torna-se impraticável realizar cálculos em grupos cuja ordem é muito grande. Para contornar este problema é comum utilizarmos um subconjunto do grupo, denominado de *conjunto gerador*, para representá-lo. Um conjunto gerador é um subconjunto S de G para o qual todo elemento de G pode ser representado por uma sequência de elementos $a_1, a_2, \cdots, a_n$ de S, não necessariamente distintos. Denotamos por $\langle S \rangle$ o grupo *gerado* pelo conjunto S. Note que um grupo pode possuir diferentes conjuntos geradores e de diferentes tamanhos, no entanto a Proposição 2.17 apresenta um fato importante sobre conjuntos geradores.

Proposição 2.17. Todo grupo finito G possui um conjunto gerador de até $\log(|G|)$ elementos. □

Neste trabalho, quando estamos tratando de problemas computacionais que envolvem a teoria de grupos, consideramos problemas para um grupo G específico, onde seus elementos são unicamente representados por strings de tamanho $p(\log |G|)$ para um polinômio p. Ainda, consideramos que operações do grupo de inverso, produto e teste de identidade são computáveis em tempo polinomial em $\log |G|$. De forma mais geral, analisamos o custo computacional de algoritmos para problemas sobre um grupo G sempre em função de $\log |G|$.

Apesar de não utilizado neste trabalho, um outro modelo comumente encontrado na literatura é o *modelo de caixa-preta para grupos* introduzido em Babai e Szemerédi (1984). Neste modelo, são considerados problemas para uma família $\mathcal{B} = \{B_n\}_{n \geq 1}$ de grupos finitos

B_n, onde o grupo $G \leq B_n$ de interesse é dado por um conjunto gerador T de $p(n)$ elementos e cada elemento de B_n é unicamente representado por uma string de tamanho $q(n)$, para dois polinômios p e q. Ainda, também é considerado que operações do grupo são computáveis em tempo polinomial em n e o custo computacional de algoritmos para problemas deste modelo é dado em função de n.

Um subconjunto H de um grupo G é um *subgrupo* de G, denotado $H \leq G$, se é fechado para a operação de G. Se H é um subgrupo próprio de G, denotamos por $H < G$, e se H não é um subgrupo de G, denotamos por $H \nleq G$. Se $H \leq G$ dizemos que o conjunto $aH = \{ah : h \in H\}$ é uma *coclasse* de H em G, onde $a \in G$.

Os conceitos de subgrupo e coclasse nos proporcionam uma variedade relações interessantes. O Lema 2.18 apresenta algumas propriedades triviais de coclasses.

Lema 2.18. Sejam G um grupo, $H \leq G$, e $a, b \in G$. Então

1. $aH = bH \iff a^{-1}b \in H$, em particular $aH = H \iff a \in H$.
2. Se $aH \cap bH \neq \emptyset$, então $aH = bH$.
3. $|aH| = |H|$, para todo $a \in G$. □

Um dos principais teoremas envolvendo subgrupos é o Teorema de Lagrange.

Teorema 2.19 (Teorema de Lagrange)**.** Seja H um subgrupo de um grupo finito G, então $|H|$ é um divisor de $|G|$. □

Utilizando o Teorema de Lagrange e o Lema 2.18, podemos ver que as coclasses de um subgrupo $H \leq G$ formam uma equipartição em G, como ilustrado a Figura 2.4

G

H	aH
bH	cH
$\cdots$	$\cdots$
$\cdots$	gH

Figura 2.4: Equipartição de um grupo G por coclasses de um subgrupo $H \leq G$, onde $a, b, \cdots, g$ são elementos de G.

O número de coclasses distintas de um subgrupo $H \leq G$ é o *índice* de H em G, denotado $[G : H]$. Uma vez que para todo subgrupo $H \leq G$ de um grupo finito G, $|H|$ é um divisor de $|G|$, então $[G : H] = \frac{|G|}{|H|}$.

Proposição 2.20. A interseção de quaisquer subgrupos H_1 e H_2 de um grupo G é também um subgrupo de G. □

Uma consequência imediata desta proposição é que a intersecção de um conjunto $\{H_1, H_2, \cdots, H_m\}$ de subgrupos de G também será um subgrupo de G.

Proposição 2.21 (Fórmula do Produto)**.** Sejam H_1 e H_2 dois subgrupos de um grupo G, então

$$|H_1H_2| = \frac{|H_1| \cdot |H_2|}{|H_1 \bigcap H_2|}.$$ □

Dois elementos a e b são *conjugados* se existe um elemento $g \in G$ tal que $gag^{-1} = b$, neste caso, dizemos que b é o *conjugado* de a por g, denotado a^g. Ainda, dizemos que dois subgrupos H_1, H_2 de G são conjugados se existe um elemento $g \in G$ tal que $H_1 = H_2{}^g$, onde $H_2{}^g = \{h^g : h \in H_2\}$.

Um subgrupo H de um grupo G é um *subgrupo normal*, e escrevemos $H \trianglelefteq G$, quando $h^g \in H$ para todo $g \in G$ e para todo $h \in H$. Se os únicos subgrupos normais de um grupo G são G e o subgrupo trivial $\{e\}$, então G é dito um *grupo simples*.

Definição 2.22. (Fecho Normal) Sejam G um grupo e $S \subseteq G$. O *fecho normal* de S em G é o menor subgrupo normal de G que contém S.

Equivalentemente (Holt et al., 2005), podemos definir $ncl_G(S)$ como o grupo gerado pelo conjunto $S^G = \{s^g : s \in S, g \in G\}$, isto é $ncl_G(S) = \langle S^G \rangle$.

A *classe de conjugação* de um elemento $a \in G$, denotada, $Cl_G(a)$, é dada por $Cl_G(a) = \{a^g : g \in G\}$. Duas classes de conjugação $Cl_G(a)$ e $Cl_G(b)$ são iguais se a e b são conjugados, senão $Cl_G(a) \cap Cl_G(b) = \emptyset$. Imediatamente, temos que o conjunto de todas as classes de conjugação de um grupo G formam uma partição deste grupo. A classe de conjugação de um subgrupo H, denotada $Cl_G(H)$, é dada por $Cl_G(H) = \{H^g : g \in G\}$.

Definição 2.23 (Normalizador)**.** Seja G um grupo. O *normalizador* de um subconjunto $S \subseteq G$ é dado por $N_G(S) = \{g \in G : S^g = S\}$.

Se S é um subgrupo de G, então $S \trianglelefteq N_G(S)$, sendo $N_G(S)$ o maior subgrupo de G para o qual S é normal.

Proposição 2.24. Sejam G um grupo e $H \leq G$. Então $|Cl_G(H)| = [G : N_G(H)]$. □

Proposição 2.25. Sejam G um grupo, $H \leq G$ e $a, b \in G$. Então $H^a = H^b$ se e somente se $aN_G(H) = bN_G(H)$.

Demonstração. Se $H^a = H^b$, então $H^{ab^{-1}} = H$, o que implica que $ab^{-1} \in N_G(H)$. Portanto, $aN_G(H) = bN_G(H)$. Por outro lado, se $aN_G(H) = bN_G(H)$, então $ab^{-1} \in N_G(H)$, o que implica que $H^{ab^{-1}} = H$. Consequentemente, temos $H^a = H^b$. □

Definição 2.26 (Grafo de Cayley)**.** Sejam G um grupo e $R \subseteq G$ tais que $G = \langle R \rangle$. O *grafo de Cayley*[1] $\Gamma(G, R)$, é um grafo direcionado $\mathcal{G} = (V, E)$ que pode possuir laços, onde o conjunto de vértices é dado por $V = \{v_g : g \in G\}$, de modo que existe uma bijeção $\varphi : G \to V$ tal que $\forall g \in G, \varphi(g) = v_g$, e o conjunto de arestas é dado por $E = \{(v_g, v_{gr}) : g \in G, r \in R\}$.

[1]Grafo de Cayley também pode ser definido sem a necessidade de R ser um conjunto gerador de G, nestes casos $\Gamma(G, R)$ é um grafo desconexo e cada componente conexo é associado a coclasse de $\langle R \rangle$.

Dizemos que um conjunto R é um *conjunto simétrico* se para todo $r \in R$, o inverso de r, r^{-1}, também pertence ao conjunto R. Sejam G um grupo e $R \subseteq G$ tais que R é simétrico e $g = \langle R \rangle$, então o grafo de Cayley $\Gamma(G, R)$ é um grafo $|R|$-regular não direcionado.

Em teoria de grupos, muitas vezes queremos examinar famílias de grupos que possuem características específicas. Para alguns mais notórios, estas famílias recebem um nome específico, como é caso de *grupos cíclicos* e *grupos abelianos*.

Definição 2.27 (Grupo cíclico)**.** Um grupo $(G, +)$ é um *grupo cíclico* se contém um elemento a tal que $\langle a \rangle = G$.

Um grupo cíclico já mencionado é o grupo $(\mathbb{Z}_N, +)$ de adição módulo N.

Definição 2.28 (Grupo abeliano)**.** Um grupo $(G, +)$ é um grupo *grupo abeliano* se possui a propriedade de comutatividade, isto é, se $a_1 + a_2 = a_2 + a_1$ para todo $a_1, a_2 \in G$.

Quando estamos trabalhando com grupos abelianos, comumente iremos representar a operação do grupo por +, pois diferentemente da operação de multiplicação $*$, a operação + é comutativa para matrizes.

Uma outra família de grupos muito estudada é a família de Grupos de Permutação, para a qual dedicamos a Seção 2.4 a seguir.

2.4 GRUPOS DE PERMUTAÇÃO

Uma *permutação sobre um conjunto* Ω é uma bijeção $\pi : \Omega \to \Omega$. O conceito de permutação também pode ser aplicado a grafos, onde uma permutação $\pi : V \to V$ sobre um grafo $\mathcal{G} = (V, E)$, denotada $\pi(\mathcal{G})$, resulta em um grafo $\mathcal{G}' = (V, E')$, onde $E' = \{\{\pi(u), \pi(v)\} : \{u, v\} \in E\}$. Neste caso, dizemos que $\mathcal{G}$ é *isomorfo* a $\mathcal{G}'$, denotado $\mathcal{G} \cong \mathcal{G}'$, e π é um *isomorfismo* de $\mathcal{G}$ para $\mathcal{G}'$.

Uma permutação π é um automorfismo de um grafo $\mathcal{G}$ se e somente se $\pi(\mathcal{G}) = \mathcal{G}$, isto é, se π é um isomorfismo de $\mathcal{G}$ para ele mesmo. O conjunto de todos os automorfismos de $\mathcal{G}$ forma um grupo com a operação de composição, denotado $\mathrm{Aut}(\mathcal{G})$ ou apenas Aut se o grafo em questão está implícito.

Definição 2.29 (Grupo Simétrico)**.** O conjunto de todas as permutações de um conjunto Ω forma um grupo com a operação de composição $\circ$, chamado de *grupo simétrico* em Ω e denotado por $\mathbb{S}_\Omega$. Quando $\Omega = \{1, 2, ..., n\}$, usualmente denotamos $\mathbb{S}_\Omega$ por apenas $\mathbb{S}_n$ e chamamos de grupo simétrico em n.

Dizemos que uma permutação $\pi \in \mathbb{S}_n$ *fixa* um elemento $i \in \{1, 2, ..., n\}$ se $\pi(i) = i$, senão dizemos que π *move* i. Duas permutações são disjuntas se todo i movido por uma delas é fixado pela outra.

O conjunto de elementos de um conjunto Ω que são movidos por uma permutação $\pi \in \mathbb{S}_\Omega$ é o *suporte* de π, denotado $sup(\pi)$.

Existem várias maneiras de representar um elemento de $\mathbb{S}_n$. Uma das maneiras é por meio da notação em duas linhas de Cauchy, como é o caso do exemplo a seguir

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 1 & 2 & 5 & 4 \end{pmatrix}$$

Nesse exemplo representamos o elemento $\pi \in \mathbb{S}_n$, tal que $\pi(1) = 3$, $\pi(2) = 1$, $\pi(3) = 2$, $\pi(4) = 5$ e $\pi(5) = 4$.

Sejam $\{i_1, i_2, \cdots, i_r\}$ inteiros distintos que pertencem ao conjunto $\{1, 2, \cdots, n\}$. Chamamos uma permutação π de *r-ciclo* se $\pi(i_1) = i_2, \pi(i_2) = i_3, \cdots, \pi(i_r) = i_r, \pi(i_r) = i_1$ e $\pi(j) = j$ para todo $j \in \{1, 2, \cdots, n\}$ tal que $j \notin \{i_1, i_2, \cdots, i_r\}$. Também dizemos que π é um ciclo de tamanho r.

Proposição 2.30. Toda permutação $\pi \in \mathbb{S}_n$ é ou um ciclo ou a composição de ciclos disjuntos. □

Apresentamos um exemplo de permutação $\pi \in \mathbb{S}_n$ junto da sua representação como uma composição de ciclos disjuntos.

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 4 & 1 & 5 & 2 & 8 & 6 & 3 & 7 \end{pmatrix}$$

que pode ser representada pela composição de ciclos $\pi = (1\ 4\ 2)(3\ 5\ 8\ 7)(6)$.

Uma *transposição* é uma permutação π que é um ciclo de tamanho dois. Ou seja, uma permutação $\pi = (j, k)$ que permuta apenas dois elementos de um conjunto. Dizemos que uma permutação π é uma *permutação par*, se π pode ser decomposta por um produto que possui uma quantidade par de transposições, de outro modo, dizemos que π é uma *permutação ímpar*. A permutação π apresentada no exemplo, pode ser decomposta em transposições de modo que $\pi = (1\ 4)(4\ 2)(2\ 1)(3\ 5)(5\ 8)(8\ 7)(7\ 3)$. Ainda, podemos definir permutações pares como permutações que tem um número par de ciclos de tamanho impar, uma vez que cada ciclo de tamanho par é decomposto em um número ímpar de transposições, enquanto um ciclo de tamanho impar é decomposto em um número par de transposições.

Subgrupos do grupo simétrico são chamados de *grupos de permutação*. Em especial temos o *grupo alternante*, também conhecido como um dos grupos de permutação *gigantes*, juntamente com o grupo simétrico.

Definição 2.31. O conjunto com todas as permutações de $\mathbb{S}_n$ forma um grupo denominado de *grupo alternante* de permutações de n elementos, denotado por $\mathbb{A}_n$, e possui exatamente $n!/2$ elementos.

Finalizamos esta seção com o Lema 2.32 sobre grupos subgrupos normais em $\mathbb{S}_n$.

Lema 2.32. Para $n > 4$ o único subgrupo normal de $\mathbb{S}_n$ que não é o grupo trivial nem o próprio $\mathbb{S}_n$ é o grupo alternante $\mathbb{A}_n$. □

2.5 TEORIA DE REPRESENTAÇÃO

Nesta seção apresentaremos conceitos e análises referentes a teoria de representação, um ramo da matemática que estuda estruturas algébricas abstratas, muitas vezes representando elementos na forma de matrizes. Originalmente, teoria de representação foi desenvolvida para representação de grupos e análises referentes a teoria de grupos, nossas análises acerca deste tema estão direcionadas a esse contexto.

Nesta seção, apresentaremos conceitos e análises relacionadas à teoria de representação, um ramo da matemática que estuda estruturas algébricas abstratas, frequentemente representando elementos na forma de matrizes. Originalmente, teoria de representação foi desenvolvida para representar grupos e realizar análises referentes a teoria dos grupos. Nossas análises sobre este tema estão focadas nesse contexto específico.

Para uma visão mais completa de teoria de representação recomendamos os livros *Linear representations of finite groups* (Serre, 1977), *Representation theory: a first course*

(Fulton e Harris, 2013) e *Representation theory of finite groups* (Steinberg, 2009), utilizados como referência para a elaboração desta seção.

Assumimos que os espaços vetoriais considerados são de dimensão finita e sobre o corpo $\mathbb{C}$ de números complexos. Dito isso, denotamos por $GL(V)$ o grupo matrizes invertíveis em um espaço vetorial V. Note que todo elemento $M \in GL(V)$ é uma transformação linear tal que $M : V \to V$.

Uma vez que estas noções foram estabelecidas, podemos definir o conceito de *representação linear* (*representação*), cuja principal função será representar grupos finitos, com suas respectivas operações, por meio de matrizes.

Definição 2.33 (Representação linear)**.** Uma *representação* de um grupo G em um espaço vetorial V é um homomorfismo $\sigma : G \to GL(V)$. Uma vez que toda representação σ está associada a um espaço vetorial V, dizemos que a dimensão de σ, denotada d_σ, é a dimensão de V. Ao fixar uma base para o espaço vetorial V, a representação $\sigma(g)$ de um elemento $g \in G$ pode ser descrita por uma matriz $d_\sigma \times d_\sigma$ de valores complexos, denotada $\big[\sigma(g)\big]$, para a qual o elemento da linha j e da coluna k é representado por $\big[\sigma(g)\big]_{j,k}$.

A partir da definição apresentada, as seguintes proposições podem ser formuladas, as quais servirão para provar determinados resultados desejados.

Proposição 2.34. Seja $\sigma : G \to GL(V)$ uma representação linear de um grupo G, então $\sigma(g^{-1}) = \sigma(g)^{-1}$. Adicionalmente, quando fixamos uma base para V, $\sigma(g)^{-1} = \big[\sigma(g)\big]^\dagger$, para todo $g \in G$. □

Proposição 2.35. Sejam σ uma representação linear de um grupo G e a, b elementos deste grupo, então

$$\big[\sigma(ab)\big]_{j,k} = \sum_{l=1}^{d_\sigma} \big[\sigma(a)\big]_{j,l}\big[\sigma(b)\big]_{l,k}. \qquad \square$$

Sejam σ_1 e σ_2 duas representações de um grupo G nos espaços vetoriais V_1 e V_2 respectivamente. Dizemos que σ_1 e σ_2 são *isomorfas*, denotado $\sigma_1 \cong \sigma_2$, se e somente se existe uma transformação linear $T : V_1 \to V_2$ tal que $T \circ \sigma_1(g) = \sigma_2(g) \circ T$ para todo $g \in G$. Se σ_1 e σ_2 não são isomorfas denotamos $\sigma_1 \ncong \sigma_2$. Portanto temos que se σ_1 e σ_2 são isomorfas, o seguinte diagrama comuta, isto é, todos os seus caminhos com o mesmo inicio e fim levam ao mesmo resultado por composição.

$$\begin{array}{ccc} V_1 & \xrightarrow{\sigma_1(g)} & V_1 \\ \Big\downarrow{\scriptstyle T} & & \Big\downarrow{\scriptstyle T} \\ V_2 & \xrightarrow{\sigma_2(g)} & V_2 \end{array}$$

Observe que quando fixamos bases para os espaços vetoriais V_1 e V_2, isto significa que existe uma matriz invertível T tal que $T\big[\sigma_1(g)\big] = \big[\sigma_2(g)\big]T$ para todo $g \in G$. Note também que se a dimensão dos espaços vetoriais V_1 e V_2 é diferente, então $\sigma_1 \ncong \sigma_2$.

Dizemos que um subespaço $W \leq V$ é *G-invariante* se, para todo $g \in G$ e $w \in W$, temos que $\sigma(g)(w) \in W$, onde G é um grupo e $\sigma : G \to GL(V)$ uma representação de G em um espaço vetorial V. Dado um espaço vetorial V dizemos que os subespaços $\{0\}$ e V são subespaços triviais, onde $\{0\}$ é o espaço vetorial nulo. Outros subespaços são considerados não triviais.

Uma representação $\sigma : G \to GL(V)$ é dita *redutível* se existem subespaços não triviais *G-invariantes* em V.

Definição 2.36 (Representações irredutíveis)**.** Uma representação σ de um grupo G em um espaço vetorial V é dita *irredutível* se e os únicos subespaços *G-invariante*s de V são subespaços triviais.

As representações irredutíveis são as representações mais simples e fundamentais de um grupo. Elas não podem ser decompostas em representações menores, tornando-as blocos de construção básicos para entender a estrutura de grupos e suas representações. No Apêndice B, apresentamos alguns conceitos e discussões que podem auxiliar no entendimento de representações irredutíveis.

Uma vez compreendido o conceito de representação irredutível, conseguimos agora estabelecer uma definição adequada para o conjunto de representações irredutíveis, distintas sob isomorfismo.

Definição 2.37 (Conjunto de representações irredutíveis)**.** O *conjunto de representações irredutíveis* de um grupo G, denotado $\hat{G}$, é o conjunto que contém todas as representações irredutíveis de G, onde cada representação irredutível σ de está associada a espaço vetorial V_σ específico. Além disso para quaisquer σ_1 e σ_2 em $\hat{G}$, temos $\sigma_1 \ncong \sigma_2$. Se G é um grupo finito, então o conjunto $\hat{G}$ é também finito.

A Proposição 2.38 estabelece uma relação fundamental entre as dimensões das representações irredutíveis em $\hat{G}$ e a ordem do grupo G.

Proposição 2.38. Seja $\hat{G}$ o conjunto de representações irredutíveis de um grupo G, então $\sum_{\sigma\in\hat{G}} d_\sigma^2 = |G|$. □

A *representação trivial* de um grupo G é um homomorfismo $\sigma : G \to \mathbb{C}_{\neq 0}$ dada por $\sigma(g) = 1$ para todo $g \in G$. Além de representações triviais e não triviais, também fazemos uso de *representações regulares*. Sejam G um grupo e V um espaço vetorial de dimensão $|G|$, para o qual o conjunto de vetores ortonormais $\{v_g : g \in G\}$ compõe sua base. Uma representação regular $\mathrm{reg}_G : G \to GL(V)$ é definida por $\mathrm{reg}_G(x) : V \to V$, de modo que $\mathrm{reg}_G(x)(v_g) = v_{xg}$, para qualquer $x \in G$. Em nossas análises utilizamos duas representações regulares, a representação regular à esquerda, denotada por L, e a representação regular à direita, denotada por R. Uma vez que utilizamos tais representações em um contexto de computação quântica, definimos a ação de L por $L(x)\,|y\rangle = |xy\rangle$ e R por $R(x)\,|y\rangle = |yx^{-1}\rangle$, onde x e y são elementos de um grupo.

Na Definição 2.39, apresentamos o conceito de *caracter*, que desempenha um papel fundamental na teoria de representação, especificamente na teoria de caracteres. Um caracter χ_σ está intrinsecamente ligado a uma representação $\sigma : G \to GL(V)$, e sua característica distintiva é que ele permanece inalterado, independentemente da base escolhida para o espaço vetorial V.

Definição 2.39 (Caracter)**.** O *caracter* de uma representação $\sigma : G \to GL(V)$ é a função $\chi_\sigma : G \to \mathbb{C}_{\neq 0}$, dada por $\chi_\sigma(g) = \mathrm{tr}(\sigma(g))$ para todo $g \in G$, onde tr é a função do traço. Se σ é uma representação irredutível, então chamamos χ_σ de *caracter irredutível*.

O lema a seguir apresenta algumas propriedades de caracteres.

Lema 2.40. Seja $\sigma : G \to GL(V)$ uma representação de um grupo G.

1. $\chi_\sigma(e) = d_\sigma$.
2. $\chi_\sigma(a^{-1}) = \chi_\sigma(a)^*$, para todo $a \in G$.
3. $\chi_\sigma(bab^{-1}) = \chi_\sigma(a)$, para todo $a, b \in G$. □

Em especial, caracteres são úteis quando trabalhamos com grupos abelianos, sendo equivalente dizer que um grupo G é abeliano e todas as suas representações irredutíveis possuem dimensão 1, sendo assim, de fato, caracteres irredutíveis. Logo, para grupos abelianos, é possível observar que $G \cong \hat{G}$. Uma propriedade particular de caracteres irredutíveis de grupos abelianos é que $\chi(a+b) = \chi(a)\chi(b)$, onde $\chi : G \to \mathbb{C}_{\neq 0}$ é um caracter irredutível de um grupo abeliano $(G, +)$ e $a, b \in G$.

Encerramos esta seção com a Proposição 2.41 que é um caso particular da ortogonalidade de caractere e da ortogonalidade de representações irredutíveis (ver Apêndice B), fundamentais em teoria de representação e que podem ser derivadas do Lema de Schur, também apresentado no Apêndice B.

Proposição 2.41. Sejam $(G, +)$ um grupo abeliano, $\chi : G \to \mathbb{C}_{\neq 0}$ uma caracter irredutível de G, e $H \leq G$. Então

$$\frac{1}{|H|} \sum_{h \in H} \chi(h) = \begin{cases} 1, & \text{Se } \chi(h) = 1, \text{ para todo } h \in H; \\ 0 & \text{Caso contrário} \end{cases}$$

Demonstração. Se $\chi(h) = 1$ para todo $h \in H$ então $\sum_{h \in H} \chi(h) = |H|$. Senão, existe $a \in H$ tal que $\chi(a) \neq 1$, logo

$$\chi(a) \sum_{h \in H} \chi(h) = \sum_{h \in H} \chi(a)\chi(h) = \sum_{h \in H} \chi(a+h) = \sum_{h' \in H} \chi(h').$$

Como $\chi(a) \neq 1$, então $\sum_{h \in H} \chi(h) = 0$. □

3 O PROBLEMA DO SUBGRUPO OCULTO

Neste capítulo apresentamos o problema do subgrupo oculto e discutimos alguns algoritmos quânticos para o mesmo, assim como outras análises sobre a complexidade deste problema. O Problema do Subgrupo Oculto (HSP, de *Hidden Subgroup Problem*) é um problema de teoria de grupos candidato a ser NP-intermediário. Problemas como fatoração, logaritmo discreto, isomorfismo de grafos (Nielsen e Chuang, 2002), e o problema do vetor mais curto em reticulados (Regev, 2004) podem ser vistos como casos particulares de HSP.

Dizemos que uma função f *oculta* um subgrupo $H \leq G$ se para todo $g_1, g_2 \in G$, $f(g_1) = f(g_2)$ se e somente se $g_1 H = g_2 H$. Ou seja, uma função f oculta um subgrupo H em G se ela é tem o mesmo valor para todos para os elementos de G que pertencem à mesma coclasse e diferentes valores para elementos de diferentes coclasses.

Definição 3.1 (Problema do Subgrupo Oculto)**.** Dado um grupo G e uma função $f : G \to S$, onde S é um conjunto finito e f oculta um subgrupo H em G, o problema consiste em encontrar um conjunto gerador para H.

Em um contexto clássico, a função f é dada por um circuito caixa-preta $C_f : \{0,1\}^n \to \{0,1\}^m$ que recebe como entrada strings que representam elementos de G e tem como saída strings de m bits que representam a saída da função $f(x)$. Para o contexto quântico, recebemos um circuito quântico caixa-preta U_f de modo que

$$U_f \left|x\right\rangle \left|y\right\rangle = \left|x\right\rangle \left|y \oplus f(x)\right\rangle ,$$

quando x, y são estados da base computacional, de modo que $x \in \{0,1\}^n$ é a codificação de um elemento $g \in G$ e $y \in \{0,1\}^m$ (usualmente $y = 0^m$). Aqui, $\oplus$ denota a operação XOR bit a bit.

Uma vez que o problema de isomorfismo de grafos é polinomialmente equivalente ao problema de contagem de automorfismo de grafos (#GA, de *Graph Automorphism*), podemos discutir apenas algoritmos para #GA. Em um contexto que estamos discutindo o problema do subgrupo oculto, é conveniente discutirmos #GA ao invés do problema de isomorfismo de grafos, pois o primeiro é mais simples de analisar como caso de HSP.

Definição 3.2 (Problema de Contagem de Automorfismos de Grafos - #GA)**.** Dado um grafo $\mathcal{G}$ determinar o número de automorfismos de $\mathcal{G}$.

Podemos observar que #GA é um caso particular de HSP fazendo $G = \mathbb{S}_n$, $H = \mathrm{Aut}(\mathcal{G})$, $f(\pi) = \pi(\mathcal{G})$ e a operação do grupo a operação de composição $\circ$.

HSP se destaca na computação quântica por possuir casos particulares que admitem algoritmos quânticos polinomiais, mas que não se conhece algoritmos clássicos polinomiais, como os problemas de fatoração e logaritmo discreto (Shor, 1997).

Para o caso de #GA, as técnicas utilizadas em Shor (1997) se mostraram insuficientes para resolver o problema em tempo polinomial (Moore et al., 2005; Hallgren et al., 2010).

Apesar disso, Ettinger, Høyer e Knill (Ettinger et al., 2004) apresentam resultados promissores utilizando novas técnicas, mostrando que com um número polinomial de avaliações de f e um pós processamento ineficiente sem invocar f, conseguimos resolver HSP, diferenciando do caso clássico, onde ainda se considera necessário um número superpolinomial[1] de avaliações de f para resolvê-lo.

3.1 TRANSFORMAÇÕES DE FOURIER

A transformação de Fourier é uma ferramenta matemática que converte dados que estão originalmente representados no *domínio do sinal* em dados representados no *domínio da frequência*. Existem diferentes tipos de transformações de Fourier, tais como a transformação discreta, contínua, booleana, quântica, entre outras. Em particular apresentamos a transformação quântica de Fourier, uma transformação que recebe como entrada um estado quântico e produz como saída um outro estado quântico. Apresentamos na Definição 3.3 a representação matemática da transformação quântica de Fourier que será utilizada neste texto.

Definição 3.3 (Transformação Quântica de Fourier)**.** A transformação quântica de Fourier (QFT, de *Quantum Fourier Transform*) é uma transformação unitária que transforma um estado quântico $|\psi_{in}\rangle \in \mathcal{H}_N$, em um outro estado quântico $|\psi_{out}\rangle \in \mathcal{H}_N$, onde $N > 0$, podendo ser representada pelo seguinte operador:

$$F_N = \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} \sum_{x=0}^{N-1} \omega_N^{yx} \, |y\rangle \langle x|$$

onde $\omega_N = e^{2\pi i/N}$.

Observe que a transformação quântica de Fourier, de acordo com a Definição 3.3, pode ser vista como uma operação sobre os elementos do grupo cíclico $\mathbb{Z}_N$. Inicialmente, foi desenvolvido um circuito quântico polinomial que computa a transformação quântica de Fourier para o grupo $\mathbb{Z}_{2^n}$ (Coppersmith, 1994; Cleve, 1994), isto é, quando N é uma potência de 2. Posteriormente, o circuito foi adaptado para o grupo $\mathbb{Z}_N$ (Kitaev, 1995; Shor, 1997), permitindo assim computar F_N em tempo polinomial para qualquer inteiro positivo N, recebendo diversos aprimoramentos desde então (Hales e Hallgren, 2000; Cleve e Watrous, 2000; Hales, 2002). Para construirmos uma transformação quântica de Fourier que atue sobre elementos de um grupo G que não é cíclico, é necessário reformular a definição de acordo com o grupo G desejado (Serre, 1977).

Quando G é um grupo abeliano finito, podemos decompô-lo em subgrupos cíclicos (Rotman, 2005), isto é, $G \cong \mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2} \times \cdots \times \mathbb{Z}_{N_k}$, de modo que cada elemento $g \in G$ pode ser representado por uma tupla $(g_1, g_2, \cdots, g_k)$, onde $g_i \in \mathbb{Z}_{N_i}$. Dado que aplicar a transformação quântica de Fourier sobre o produto tensorial $|g_1\rangle \, |g_2\rangle \cdots |g_k\rangle$, onde $(g_1, g_2, \ldots, g_k) \in \mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2} \times \cdots \times \mathbb{Z}_{N_k}$, é simplesmente aplicar o produto tensorial das transformações quânticas de Fourier sobre cada $\mathbb{Z}_{N_j}$, basta decompor o grupo abeliano G em subgrupos cíclicos, algo que pode ser realizado em tempo polinomial por um algoritmo quântico (Mosca, 1999; Cheung e Mosca, 2001), e utilizar o

[1] Dizemos que uma função de complexidade $f(n)$ é subexponencial se $\log f(n) = o(n)$. Dizemos que $f(n)$ é superpolinomial se para todo inteiro k, $f(n) = \omega(n^k)$.

circuito polinomial para a Definição 3.3. A Definição 3.4 apresenta a representação matemática da transformação quântica de Fourier para grupos abelianos finitos.

Definição 3.4 (Transformação de Fourier para Grupos Abelianos)**.** A transformação quântica de Fourier (QFT) para um grupo abeliano $G \cong \mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2} \times \cdots \times \mathbb{Z}_{N_k}$ é uma transformação unitária que transforma um estado quântico $|\psi_{in}\rangle \in \mathcal{H}_G$, em um outro estado quântico $|\psi_{out}\rangle \in \mathcal{H}_G$, podendo ser representada pelo seguinte operador:

$$\begin{aligned} F_G &= F_{N_1} \otimes F_{N_2} \otimes \cdots \otimes F_{N_k} \\ &= \frac{1}{\sqrt{|G|}} \sum_{y \in G} \sum_{x \in G} \chi_y(x) \, |y\rangle \, \langle x| \end{aligned}$$

onde $\chi_y(x) = \omega_{N_1}^{y_1 x_1} \omega_{N_2}^{y_2 x_2} \cdots \omega_{N_k}^{y_k x_k}$, $|x\rangle = |x_1\rangle \, |x_2\rangle \cdots |x_k\rangle$ e $|y\rangle = |y_1\rangle \, |y_2\rangle \cdots |y_k\rangle$

Diferentemente da transformação quântica de Fourier para grupos abelianos finitos, ainda não conhecemos um circuito quântico polinomial para a transformações de Fourier para grupos em geral (incluindo grupos não-abelianos), como o grupo $\mathrm{GL}_n(\mathbb{F}_q)$ de matrizes invertíveis $n \times n$ sobre um corpo finito de q elementos, para o qual o melhor algoritmo conhecido tem custo subexponencial, mas ainda superpolinomial (Moore et al., 2006). No entanto, para o caso particular do grupo simétrico $\mathbb{S}_n$ é possível construir um circuito polinomial (Beals, 1997), o que particularmente é de nosso interesse, uma vez que o problema de automorfismo de grafos é reduzido ao problema do subgrupo oculto para grupos simétricos. Também são conhecidos circuitos para a transformação quântica de Fourier para outros grupos não-abelianos, como grupos de Clifford (Moore et al., 2006), metabelianos (Moore et al., 2006), metacíclicos (Hoyer, 1997), entre outros (Moore et al., 2006; Püschel et al., 1999). A Definição 3.5 apresenta a representação matemática da transformação quântica de Fourier para qualquer grupo finito, definição adaptada de (Childs e Van Dam, 2010) e (Serre, 1977).

Definição 3.5 (Transformação de Fourier para Grupos Arbitrários)**.** A transformação quântica de Fourier (QFT) para um grupo G é uma transformação unitária que transforma um estado quântico $|\psi_{in}\rangle \in \mathcal{H}_G$, em um outro estado quântico $|\psi_{out}\rangle \in \mathcal{H}_G$, podendo ser representada pelo seguinte operador:

$$F_G = \sum_{g \in G} \sum_{\sigma \in \hat{G}} \sqrt{\frac{d_\sigma}{|G|}} \sum_{j,k=1}^{d_\sigma} \Big[\sigma(g)\Big]_{j,k} \, |\sigma, j, k\rangle \, \langle g|$$

onde σ é uma representação irredutível de G, d_σ sua dimensão e j, k os índices da matriz.

A transformação de Fourier tem inúmeras aplicações, podendo ser empregada em diversos problemas. A transformação quântica de Fourier foi amplamente difundida após o artigo publicado por Peter Shor, que apresenta algoritmos quânticos polinomiais para o problema de fatoração e logaritmo discreto (Shor, 1997). A transformação quântica de Fourier para grupos abelianos nos permite construir algoritmos quânticos polinomiais para o problema do subgrupo

oculto onde o grupo é abeliano (Kitaev, 1995), que será discutido na Seção 3.2. No entanto, embora existam circuitos quânticos polinomiais que realizam a transformação quântica de Fourier para casos específicos de grupos não-abelianos, até o momento só temos conhecimento de algoritmos quânticos polinomiais para alguns desses grupos (Hallgren et al., 2003; Grigni et al., 2004; Gavinsky, 2004; Gonçalves et al., 2017). Um exemplo de grupo G para o qual conhecemos um algoritmo quântico polinomial para a transformação quântica de Fourier, mas até o momento não temos algoritmo quântico para resolver HSP, é o grupo simétrico $\mathbb{S}_n$.

3.2 MÉTODO PADRÃO

Como apontado anteriormente, para muitos casos de HSP ainda não conhecemos algoritmos polinomiais. No entanto, diferentemente do caso geral de HSP, o caso em que o grupo é abeliano (HSP abeliano) possui solução polinomial.

Para resolver HSP onde o grupo é abeliano, o melhor algoritmo conhecido atua de forma similar aos algoritmos de Simon (Simon, 1997) e Shor (Shor, 1997), e é comumente chamado de *método padrão* para resolver HSP (Kitaev, 1995; Childs e Van Dam, 2010).

Para um grupo abeliano $(G, +)$, iniciamos dois registradores, colocando o primeiro registrador em uma distribuição uniforme sobre os elementos de G e aplicamos a função f ao segundo registrador, gerando o estado indicado por $|G, f\rangle$, onde

$$|G, f\rangle = \frac{1}{\sqrt{|G|}} \sum_{g \in G} |g, f(g)\rangle \tag{3.1}$$

Após aplicar a caixa-preta U_f, medimos o segundo registrador e ignoramos o resultado, obtendo uma sobreposição sobre uma das coclasses do subgrupo oculto.

$$|b + H\rangle = \frac{1}{\sqrt{|H|}} \sum_{h \in H} |b + h\rangle \tag{3.2}$$

onde $b \in G$. Em seguida, aplicamos a QFT para grupos abelianos, apresentada na Definição 3.4, sobre o estado (3.2).

$$\begin{aligned} F_G |b + H\rangle &= \frac{1}{\sqrt{|G||H|}} \sum_{y \in G} \sum_{x \in G} \sum_{h \in H} \chi_y(x) |y\rangle \langle x| |b + h\rangle \\ &= \frac{1}{\sqrt{|G||H|}} \sum_{y \in G} \sum_{h \in H} \chi_y(b + h) |y\rangle \\ &= \sqrt{\frac{|H|}{|G|}} \sum_{y \in G} \chi_y(b) \chi_y(H) |y\rangle \end{aligned} \tag{3.3}$$

onde $\chi_y(H) = \frac{1}{|H|} \sum_{h \in H} \chi_y(h)$. Pela Proposição 2.41, temos que $\chi_y(H) = 1$ se e somente se $\chi_y(h) = 1$ para todo $h \in H$, caso contrário $\chi_y(H) = 0$. Portanto o resultado da aplicação da QFT sobre uma das coclasses de H é o estado (3.4), onde $H^\perp = \{y \in G : \chi_y(h) = 1 \text{ para todo } h \in H\}$.

$$F_G |b + H\rangle = \sqrt{\frac{|H|}{|G|}} \sum_{y \in H^\perp} \chi_y(b) |y\rangle \tag{3.4}$$

Realizamos uma medição do registrador, obtendo assim $y \in H^\perp$.

Com $O(\log|G|)$ amostras de $H^{\perp}$, escolhidas uniforme e independentemente ao acaso, conseguimos, com alta probabilidade de sucesso, construir um gerador para o subgrupo oculto H (Childs, 2021). Iremos demonstrar um resultado um pouco mais fraco, que com $2(\log|G|)^2$ amostras, escolhidas uniforme e independentemente ao acaso, conseguimos, com probabilidade ao menos $\frac{1}{\sqrt{2}}$, construir um conjunto gerador para H.

Proposição 3.6. Sejam $y_1, y_2, \cdots, y_j, y_{j+1}$ amostras de $H^{\perp}$ escolhidas uniforme e independentemente ao acaso, e $K_i = \cap_{j=1}^{i} \ker(\chi_{y_j})$, onde $\ker(\chi_{y_j}) = \{g \in G : \chi_{y_j}(g) = 1\}$. Então, se $K_j \neq H$,

$$\Pr[|K_j| \geq 2 \cdot |K_{j+1}|] \geq \frac{1}{2}.$$

Demonstração. Inicialmente note que $H \leq \ker(\chi_y) \leq G$, garantindo que $H \leq K_j$ para todo j. Note também que se $K_j \not\leq \ker(\chi_{y_{j+1}})$, então $K_j \cap \ker(\chi_{y_{j+1}}) < K_j$. Portanto, pelo teorema de Lagrange, $\frac{1}{2} \leq \frac{|K_{j+1}|}{|K_j|}$, desta forma garantindo uma redução significativa no tamanho do grupo.

Agora, iremos mostrar que este evento acontece com alta probabilidade. Podemos verificar que, se $K_j \neq H$ então $\Pr[K_j \leq \ker(\chi_{y_{j+1}})] \leq \frac{1}{2}$ da seguinte forma

$$\Pr[K_j \leq \ker(\chi_{y_{j+1}})] = \frac{|H|}{|G|}|\{y \in G : K_j \leq \ker(\chi_y)\}|.$$

Sabemos que se K_j fosse o subgrupo oculto, teríamos uma amostra uniforme de valores de y, cada qual com probabilidade $\frac{|K_j|}{|G|}$. Logo $|\{y \in G : K_j \leq \ker(\chi_y)\}| = \frac{|G|}{|K_j|}$ e com isso

$$\Pr[K_j \leq \ker(\chi_{y_{j+1}})] = \frac{|H|}{|G|}\frac{|G|}{|K_j|} \leq \frac{1}{2}.$$

Portanto

$$\Pr[|K_j| \geq 2 \cdot |K_{j+1}|] \geq \frac{1}{2}.$$

□

Imediatamente, temos que para $x \in \mathbb{Z}_{>0}$

$$\Pr[|K_j| \geq 2 \cdot |K_{j+x}|] \geq (1 - \frac{1}{2^x}).$$

Agora, sendo $n = \log|G|$,

$$\Pr[|K_j| \geq 2 \cdot |K_{j+2n}|] \geq (1 - \frac{1}{2^{2n}}).$$

Observe que no pior caso, $|H| = 1$, será necessário que este evento aconteça n vezes. Resultando em uma probabilidade de sucesso Pr tal que

$$\Pr \geq (1 - \frac{1}{2^{2n}})^n.$$

Agora, iremos mostrar que $\Pr \geq \frac{1}{\sqrt{2}}$. Note que para $x \in \mathbb{Z}_{>0}$ temos

$$2^{x+1/x} \geq 2^x + 2^{\frac{1}{x}}.$$

Reordenando os termos da inigualdade obtemos

$$1 - \frac{1}{2^x} \geq 2^{-\frac{1}{x}}.$$

Fazendo $x = 2n$, temos

$$1 - \frac{1}{2^{2n}} \geq 2^{-\frac{1}{2n}}.$$

Por fim, temos que

$$(1 - \frac{1}{2^{2n}})^n \geq 2^{-\frac{1}{2}}$$

Portanto, a probabilidade de sucesso do algoritmo é de ao menos $\frac{1}{\sqrt{2}}$.

Desta forma, podemos verificar que o método padrão resolve HSP para casos abelianos, e que se um problema Π pode ser reduzido a um caso de HSP em que G é abeliano, então $\Pi \in \mathsf{BQP}$, e assim possuindo algoritmo quântico polinomial. O método utilizado pode ser generalizado pelo Algoritmo 1.

Algoritmo 1 Método Padrão para HSP:

1: Inicia um estado com dois registradores, onde o primeiro registrador contém uma sobreposição de iguais probabilidades sobre os elementos de G.

$$|G, 0\rangle = \frac{1}{\sqrt{|G|}} \sum_{g \in G} |g, 0\rangle \tag{3.5}$$

2: Aplica a caixa-preta U_f sobre os elementos de G, armazenando o resultado no segundo registrador.

$$|G, f\rangle = \frac{1}{\sqrt{|G|}} \sum_{g \in G} |g, f(g)\rangle \tag{3.6}$$

3: Realiza uma medição no segundo registrador, deixando o primeiro registrador em uma coclasse $|gH\rangle$ de H. Como todas as coclasses possuem o mesmo tamanho, todas são igualmente prováveis de serem obtidas.

$$|gH\rangle = \frac{1}{\sqrt{|H|}} \sum_{h \in H} |gh\rangle \tag{3.7}$$

Uma vez que cada coclasse possui a mesma probabilidade de ser obtida, podemos analisar o estado do primeiro registrador, como um estado misto destas probabilidades, representado pela matriz de densidade ρ_H, o qual denominamos de estado do subgrupo oculto.

$$\rho_H = \frac{1}{|G|} \sum_{g \in G} |gH\rangle \langle gH| \tag{3.8}$$

4: Utilizando um conjunto de amostras de ρ_H, determinar o subgrupo H.

Note que no algoritmo utilizado para o caso abeliano de HSP aplicamos a transformação quântica de Fourier, deste modo realizando uma amostragem de Fourier. Na Seção 3.3 discutimos a aplicação da amostragem de Fourier para grupos não abelianos, apresentando resultados relevantes para diversos grupos, em especial para o grupo simétrico $\mathbb{S}_n$.

3.3 AMOSTRAGEM DE FOURIER

Uma vez que o caso abeliano de HSP é resolvido utilizando o método padrão, surge o questionamento se uma abordagem similar pode resolver casos não abelianos de HSP, entre eles o caso onde o grupo G é simétrico. Deste modo, discutimos nesta Seção a amostragem de Fourier, a qual atua de forma similar e utiliza a transformação de Fourier para grupos arbitrários, apresentada na Definição 3.5. Esta seção tem o intuito de apresentar uma ferramenta muito utilizada em estudos do HSP, para a qual muitas análises foram adaptadas de (Childs e Van Dam, 2010).

Sejam $L(\cdot)$ e $R(\cdot)$ representações regulares à esquerda e à direita respectivamente, de modo que $L(x)\,|y\rangle = |xy\rangle$ e $R(x)\,|y\rangle = \left|yx^{-1}\right\rangle$, para todo $x, y \in G$. Realizando uma análise de Fourier sobre a representação regular à direita, obtemos

$$\begin{aligned}\hat{R}(x) &= F_G R(x) F_G^{\dagger} \\ &= \sum_{\sigma\in\hat{G}} |\sigma\rangle\langle\sigma| \otimes I_{d_\sigma} \otimes \Big[\sigma(x)\Big]^{*}.\end{aligned} \tag{3.9}$$

conforme demonstrado no Apêndice A.

Observe que o estado de uma coclasse de H pode ser escrito utilizando a representação regular a direita na forma $|gH\rangle = \frac{1}{\sqrt{|H|}} \sum_{h\in H} R(h)\,|g\rangle$ e com isso

$$\begin{aligned}\rho_H &= \frac{1}{|G|} \sum_{g\in G} |gH\rangle\langle gH| \\ &= \frac{1}{|G|} \sum_{h\in H} R(h).\end{aligned}$$

Logo, pela Equação (3.9), a aplicação da transformação quântica de Fourier sobre o estado do subgrupo oculto é dada por:

$$\begin{aligned}\hat{\rho}_H &= F_G \rho_H F_G^{\dagger} \\ &= \frac{1}{|G|} \sum_{\sigma\in\hat{G}} |\sigma\rangle\langle\sigma| \otimes I_{d_\sigma} \otimes \sum_{h\in H} \Big[\sigma(h)\Big]^{*} \\ &= \frac{1}{|G|} \sum_{\sigma\in\hat{G}} |\sigma\rangle\langle\sigma| \otimes I_{d_\sigma} \otimes \Big[\sigma(H)\Big]^{*}.\end{aligned}$$

Note que o estado $\hat{\rho}_H$ é um estado que utiliza três registradores, o primeiro armazena o rótulo da representação irredutível, enquanto o segundo e o terceiro registradores, respectivamente, armazenam informações sobre a linha e a coluna das representações. Com isso podemos observar que o estado $\hat{\rho}_H$ é uma matriz bloco diagonal, onde os blocos são rotulados pelas representações irredutíveis.

Enquanto para o caso abeliano de HSP a transformação de Fourier resulta em uma matriz de densidade completamente diagonalizada, para casos não abelianos ela resulta em uma matriz bloco diagonal.

$$\hat{\rho}_H = \frac{1}{|G|}\begin{pmatrix} I_{d_{\sigma_1}} \otimes \left[\sigma_1(H)\right]^* & 0 & \cdots & 0 \\ 0 & I_{d_{\sigma_2}} \otimes \left[\sigma_2(H)\right]^* & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & I_{d_{\sigma_{|\hat{G}|}}} \otimes \left[\sigma_{|\hat{G}|}(H)\right]^* \end{pmatrix}.$$

Com a matriz organizada desta forma, ao medir o primeiro registrador obtemos uma representação σ. Uma vez que não temos informações de σ em outros blocos da matriz, não perdemos informações neste sentido, possibilitando outras análises após a medição.

A probabilidade de uma representação σ ser observada é dada por

$$\Pr[\sigma] = \frac{d_\sigma}{|G|}\chi_\sigma(H)^*.$$

onde $\chi_\sigma(H) = \sum_{h\in H}\chi_\sigma(h)$, ou ainda, podemos dizer que $\Pr[\sigma]$ é $\frac{d_\sigma|H|}{|G|}$ vezes a quantidade de vezes que a representação trivial aparece ao restringirmos a ação de σ a elementos de H, conforme mostrado em (Hallgren et al., 2003).

Note que a probabilidade de uma representação σ ser observada depende apenas dela mesma e do subgrupo oculto H, não dependendo das bases utilizadas para os espaços vetoriais das representações de G, pois não medimos os registradores que armazenam informações sobre as linhas e as colunas das representações. Este fato torna atrativo medir apenas informações sobre o rótulo das representações σ. Este procedimento é chamado de *amostragem de Fourier fraca*.

Utilizando a amostragem de Fourier fraca, é possível encontrar H se este é um subgrupo normal (Hallgren et al., 2003). O algoritmo atua de forma similar ao algoritmo utilizado para encontrar H para o caso abeliano de HSP. Adaptando o algoritmo também é possível determinar H em tempo polinomial quando $\kappa(G)$ é suficientemente grande, onde $\kappa(G)$ é a intersecção dos normalizadores de todos os subgrupos, mais especificamente se $\frac{|G|}{|\kappa(G)|} = poly(\log|G|)$ (Gavinsky, 2004). No entanto a amostragem de Fourier fraca se mostra insuficiente para resolver HSP para determinados grupos, como é o caso do grupo diedral $\mathbb{D}_n$ e para o caso do grupo simétrico $\mathbb{S}_n$ (Grigni et al., 2004) (Hallgren et al., 2003).

Desta forma, precisamos ir além da amostragem de Fourier fraca, medindo também informações sobre a linha e a coluna das representações. Observe que o segundo registrador não depende de H, logo, após uma amostragem de Fourier fraca podemos ignorar o segundo registrador, resultando no estado

$$\hat{\rho}_{H,\sigma} = \frac{\left[\sigma(H)\right]^*}{\chi_\sigma(H)^*}, \tag{3.10}$$

onde σ é a representação observada durante a amostragem de Fourier fraca.

O procedimento de realizar medições no estado $\hat{\rho}_{H,\sigma}$ é chamado de amostragem de Fourier forte. Para tal, podemos escolher uma base específica ou então utilizar uma base aleatória. Esta última, denominada de amostragem de Fourier forte aleatória, já é significativamente mais poderosa que a amostragem de Fourier fraca, produzindo informação suficiente para resolver HSP para o grupo de Heisenberg (Radhakrishnan et al., 2005) e outros grupos interessantes (Sen, 2006) (Moore et al., 2007).

Apesar de resultados promissores para diversos grupos, esta abordagem não resolve HSP para o caso simétrico $\mathbb{S}_n$. Mesmo para bases escolhidas cuidadosamente, a amostragem de Fourier forte não é capaz de distinguir se H é subgrupo trivial ou uma involução aleatória (Moore et al., 2005), isto é, uma permutação π tal que $\pi(\pi(\mathcal{G})) = \mathcal{G}$. Ainda, foi provado que para medições emaranhadas em pares de registradores a amostragem de Fourier necessita de $e^{\Omega(\sqrt{n}/\log n)}$ experimentos para distinguir tais subgrupos (Moore e Russell, 2005).

Nas palavras de Cristopher Moore, "nenhum algoritmo remotamente parecido com o algoritmo de Shor resolverá isomorfismo de grafos" (Moore, 2018), dado que isomorfismo de grafos é polinomialmente equivalente a #GA. Desta forma, é necessário buscar alternativas para tentar resolver o problema.

3.4 VERSÃO DE DECISÃO DE HSP

Uma vez que a amostragem de Fourier não consegue distinguir se $H = \{e\}$ ou não, voltamos nossa atenção para a versão de decisão do problema do subgrupo oculto (dHSP, de *decision Hidden Subgroup Problem*).

Definição 3.7 (Problema de decisão do Subgrupo Oculto)**.** Dado um grupo G e uma função $f : G \to S$, onde S um conjunto finito e f oculta um subgrupo H em G, o problema consiste em decidir se $|H| = 1$ (instância positiva) ou $|H| \geq 2$ (instância negativa).

Nesta versão, consideramos que a função f é dada por um circuito C_f de tamanho polinomial em $\log |G|$ ao invés de uma caixa-preta, que recebe como entrada codificações de elementos de G e tem como saída strings de m bits. Desta forma, trabalhamos com uma versão de HSP qual podemos relacionar as classes de problemas de promessa apresentadas no Capítulo 2.

Ainda que esta versão não seja a única versão de decisão de HSP (Sdroievski et al., 2019b; Ablayev e Vasiliev, 2009; Friedl et al., 2003; Wang, 2010), ela é a versão mais comumente encontrada na literatura, sendo estudada tanto na computação clássica (Sdroievski et al., 2019b) como na computação quântica (Kempe e Shalev, 2004; Ettinger et al., 2004; Hayashi et al., 2008).

Um caso particular de dHSP é a versão de decisão do problema de automorfismos de grafos (GA, de *Graph Automorphism*)

Definição 3.8 (Problema de Decisão de Automorfismos de Grafos - GA)**.** Dado um grafo $\mathcal{G}$, determinar se $\mathcal{G}$ possui um automorfismo não trivial.

Apesar de não sabermos se dHSP está em NP, o problema está em coNP, uma vez que para $|H| \geq 2$, podemos usar um elemento $h \neq e \in H$ como um certificado para uma instância negativa. Para verificar o certificado, basta testar se $f(h) = f(e)$.

Foi apresentado em (Sdroievski et al., 2019b) um protocolo de conhecimento zero perfeito com um verificador honesto para dHSP, estabelecendo assim que dHSP $\in$ HVPZK. Além disso, (Sdroievski et al., 2019b) mostrou que se conhecemos o tamanho do grupo G, como é o caso para grupos de permutações (Seress, 2003), então existe uma redução de Karp polinomial de

dHSP para o Problema de Aproximação da Entropia (EA), um problema de promessa completo para NISZK.

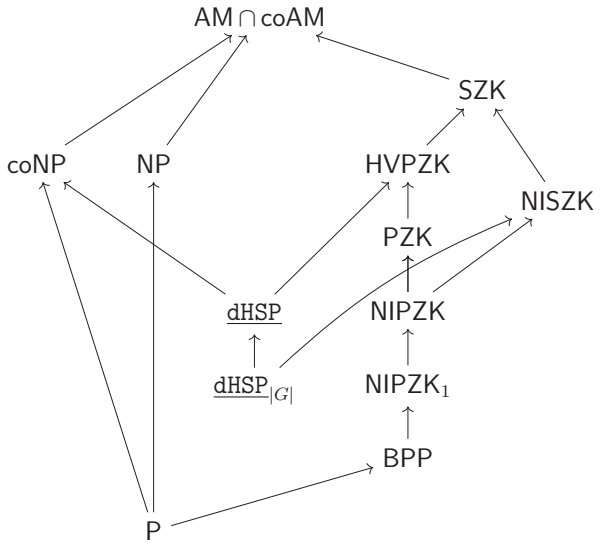


Figura 3.1: Relação conhecida entre classes de complexidade e dHSP. A seta A → B representa que: A ⊆ B se A e B são classes; A ∈ B, se A é um problema e B é uma classe; A é uma caso particular de B, se ambos A e B são problemas.

Em (Kempe e Shalev, 2004) é apresentado um estudo sobre amostragem de Fourier e dHSP para grupos de permutações, onde o objetivo é determinar casos nos quais a amostragem de Fourier é capaz de *distinguir* H do subgrupo trivial. Nesse contexto, é dito que H é *distinguível* se a amostragem fraca de Fourier ou a amostragem forte de Fourier, com bases aleatórias, nos fornece informação suficiente para distinguir H do subgrupo trivial. Um de seus resultados principais é o seguinte teorema

Teorema 3.9 (Kempe e Shalev, 2004)**.** Quando $H \leq \mathbb{S}_n$ e $|H| = \text{poly}(n)$, H é distinguível se e somente se $min\{sup(h) : h \neq e \in H\}$ é constante. □

Isto é, quando H tem tamanho polinomial, H é distinguível se e somente se H contém uma permutação na qual todos, exceto por c de pontos, são fixos, onde c é uma constante que independe de n. Deste modo, à medida que n aumenta, o número de subgrupos distinguíveis continua em função de c.

Adicionalmente, são apresentados outros teoremas que discutem casos onde $|H|$ não é polinomial, em particular onde H é primitivo. Ainda, (Kempe e Shalev, 2004) levanta a seguinte conjectura

Conjectura 3.10 (Kempe e Shalev, 2004)**.** Se $H \leq \mathbb{S}_n$ é distinguível, então $min\{sup(h) : h \neq e \in H\}$ é constante. □

Uma vez que existe apenas um número polinomial desses elementos em $\mathbb{S}_n$, a conjectura implica que a tanto a amostragem fraca de Fourier como a amostragem forte de Fourier, com bases aleatórias, não oferecem vantagens sobre a busca exaustiva clássica para resolver dHSP.

Desta forma, observamos que os algoritmos quânticos atuais não trazem vantagens em relação aos algoritmos clássicos para o problema de automorfismo de grafos, sendo o melhor algoritmo o algoritmo quasipolinomial de Babai (Babai, 2016), tendo custo computacional de tempo $\exp(O((\log n)^c)$ para alguma constante $c > 0$. Esse resultado foi posteriormente aprimorado, demonstrando-se que é possível resolver o problema com $c = 3$ (Helfgott et al., 2017).

4 RESULTADOS

Dedicamos este capítulo a apresentar resultados e análises originais obtidas neste trabalho. Na Seção 4.1 apresentamos uma heurística quântica desenvolvida para dHSP, uma alternativa aos algoritmos que utilizam amostragem de Fourier e métodos utilizando até então para abordar o problema.

Investigando classes de complexidade de provas de conhecimento zero, mostramos a relação de dHSP com a classe de provas de conhecimento zero NIPZK_1 (i.e. NIPZK com completude perfeita), discutida na Seção 4.3. Mais especificamente, mostramos que se conseguimos amostrar uniformemente os elementos de G e de um superconjunto A_f da imagem de f, tal que $|A_f| = |G|$, então dHSP admite uma prova não-interativa de conhecimento zero perfeito com completude perfeita (NIPZK_1).

Na Seção 4.4 apresentamos a prova de que NIPZK_1 possui um problema completo, que é uma versão restrita de um problema de promessa completo para a classe NIPZK.

Os resultados apresentados nas Seções 4.3 e 4.4 foram publicados em Costa et al. (2023).

4.1 MULTIPLICAÇÃO DE COCLASSES EM SOBREPOSIÇÃO

Uma vez que há indícios para acreditar que a amostragem de Fourier não oferece vantagens para resolver dHSP para o grupo simétrico e outros casos de grupos não abelianos, consideramos uma heurística que se beneficia quando um grupo é não abeliano.

Na Figura 4.1 apresentamos um circuito quântico que utiliza a heurística proposta, retornando um estado ρ^B. Para alguns casos de dHSP conseguimos mostrar que, quando o subgrupo oculto é o subgrupo trivial, ρ^B é um estado puro que representa um elemento $g \in G$, e quando o subgrupo oculto não é o subgrupo trivial, ρ^B é estado misto no qual todo elemento do grupo possui probabilidade próxima de $|G|^{-1}$ de ser obtido em uma medição.

Uma vez que assumimos que para um grupo $(G, *)$ conseguimos computar $*$ em tempo polinomial em $\log |G|$, conseguimos, pelo Lema 2.7, construir um operador quântico U_* tal que, para quaisquer $a, b \in G$, $U_* |a\rangle |b\rangle |0\rangle = |a\rangle |b\rangle |a * b\rangle$. Ainda em tempo polinomial em $\log |G|$, conseguimos construir o operador quântico $U_{*,m}$, sendo m polinomial em $\log |G|$, tal que, para quaisquer $a_1, a_2 \cdots a_m \in G$,

$$U_{*,m} |a_1\rangle |a_2\rangle \cdots |a_m\rangle |0\rangle = |a_1\rangle |a_2\rangle \cdots |a_m\rangle |a_1 * a_2 * \cdots * a_m\rangle .$$

Consideramos também que conseguimos gerar a sobreposição $|G\rangle = \frac{1}{\sqrt{|G|}} \sum_{g \in G} |g\rangle$ para o grupo G em questão.

Similarmente à matriz maximamente mista em n qubits $I/2^n$, para um conjunto $S \subseteq G$, denotamos por I/S a matriz *maximamente mista do conjunto S*, dada por

$$I/S = \frac{1}{|S|} \sum_{s \in S} |s\rangle \langle s| .$$

Dada uma sequência $(g_1, \cdots, g_m)$ de elementos de G, seja um passeio aleatório $\mathcal{W}$ no grafo de Cayley $\Gamma(ncl_G(H), H^G)$ que inicia no elemento identidade e no i-ésimo passo escolhe uniforme e independentemente ao acaso um elemento de H^{g_i}.

Dada uma sequência $(g_1, \cdots, g_m)$, a probabilidade de o passeio aleatório $\mathcal{W}$ terminar no elemento g é dada por

$$\frac{1}{|H|^m} \sum_{h_1 \in H^{g_1}} \cdots \sum_{h_m \in H^{g_m}} \begin{cases} 1 & \text{se } h_1 * \cdots * h_m = g; \\ 0 & \text{caso contrário.} \end{cases} \tag{4.1}$$

Dizemos que a sequência $(g_1, \cdots, g_m)$ é uma *sequência boa* se o passeio aleatório $\mathcal{W}$ gera uma distribuição próxima da uniforme em distância L_1, isto é, $L_1(X_{\mathcal{W}}, U_{ncl_G(H)}) \leq \epsilon$.

Teorema 4.1. Se existem um $m = \text{poly}(n)$ e um ϵ tais que ao menos $\frac{2}{3}$ das sequência são boas, então com probabilidade ao menos $\frac{2}{3}$, conseguimos produzir um estado ρ^B tal que

$$\begin{aligned} &\text{Se } |H| = 1, \text{ então} \quad D(\rho^B, I/G) = 1 - I/G; \\ &\text{Se } |H| \geq 2, \text{ então} \quad D(\rho^B, I/G) \leq 1 - \frac{|ncl_G(H)}{|G|} + \epsilon. \end{aligned}$$

Demonstração. Utilizando o método padrão, obtemos m sobreposições de coclasses de H, escolhidas uniforme e independentemente ao acaso, sejam $|a_1 H\rangle, |a_2 H\rangle, \cdots |a_m H\rangle$. Então aplicamos o operador $U_{*,m}$ sobre $|a_1 H\rangle |a_2 H\rangle \cdots |a_m H\rangle |0\rangle$, produzindo um estado ρ^B no último registrador. A Figura 4.1 apresenta o circuito quântico correspondente a este procedimento.

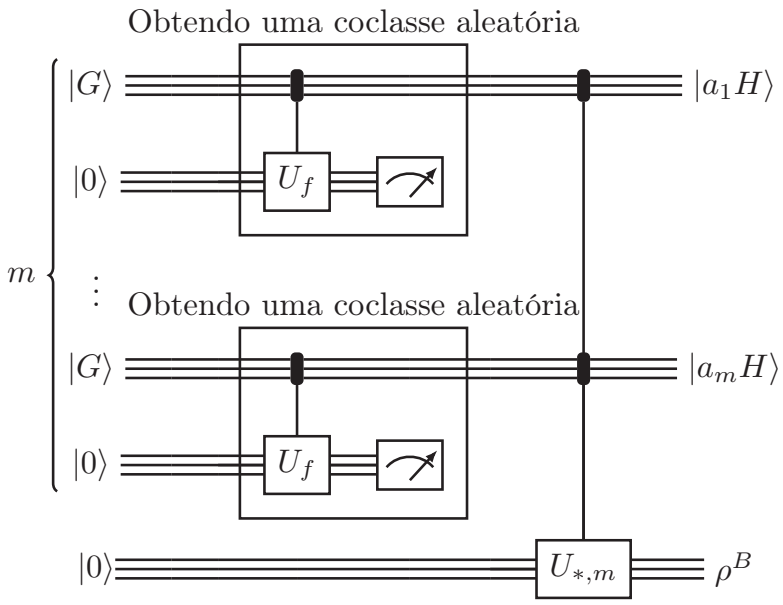


Figura 4.1: Circuito quântico de multiplicação de coclasses do subgrupo oculto.

É fácil observar que se o subgrupo oculto é o subgrupo trivial então $aH = a$, para todo $a \in G$, logo, quando $|H| = 1$, $\rho^B = |g\rangle \langle g|$, onde $g = a_1 * a_2 * \cdots * a_m$, e portanto

$$D(\rho^B, I/G) = 1 - \frac{1}{|G|}.$$

Analisamos agora o caso onde o subgrupo oculto não é o subgrupo trivial, isto é, $|H| \geq 2$. Após aplicar o operador $U_{*,m}$, o estado do sistema se encontra no estado

$$|\psi\rangle = \left(\frac{1}{\sqrt{|H|}} \sum_{h_1 \in H} |a_1 * h_1\rangle\right) \otimes \left(\frac{1}{\sqrt{|H|}} \sum_{h_2 \in H} |a_2 * h_2\rangle\right) \otimes \cdots \otimes \left(\frac{1}{\sqrt{|H|}} \sum_{h_m \in H} |a_m * h_m\rangle\right) \otimes \left(|a_1 * h_1 * a_2 * h_2 * \cdots * a_m * h_m\rangle\right).$$

Cuja matriz de densidade é dada por

$$\rho = \left(\frac{1}{|H|} \sum_{h_1, h_1' \in H} |a_1 * h_1\rangle \langle a_1 * h_1'|\right) \otimes \left(\frac{1}{|H|} \sum_{h_2, h_2' \in H} |a_2 * h_2\rangle \langle a_2 * h_2'|\right) \otimes \cdots \otimes \left(\frac{1}{|H|} \sum_{h_m, h_m' \in H} |a_m * h_m\rangle \langle a_m * h_m'|\right) \otimes \left(|a_1 * h_1 * a_2 * h_2 * \cdots * a_m * h_m\rangle \langle a_1 * h_1' * a_2 * h_2' * \cdots a_m * h_m'|\right).$$

Realizando o traço parcial no primeiro registrador, obtemos a matriz de densidade reduzida

$$\rho = \frac{1}{|H|^m} \sum_{h_1, h_1' \in H} \sum_{h_2, h_2' \in H} \cdots \sum_{h_m, h_m' \in H} \left(|a_1 * h_1\rangle \langle a_1 * h_1'|\right) \otimes \left(|a_2 * h_2\rangle \langle a_2 * h_2'|\right) \otimes \cdots \otimes \left(|a_m * h_m\rangle \langle a_m * h_m'|\right) \otimes \left(|a_1 * h_1 * a_2 * h_2 * \cdots * a_m * h_m\rangle \langle a_1 * h_1' * a_2 * h_2' * \cdots a_m * h_m'|\right).$$

Que pode ser reescrita como

$$\rho^{A_2, \cdots, A_m, B} = \frac{1}{|H|^m} \sum_{h_1, h_1' \in H} \sum_{h_2, h_2' \in H} \cdots \sum_{h_m, h_m' \in H} \left(\langle a_1 * h_1' | a_1 * h_1\rangle\right) \otimes \left(|a_2 * h_2\rangle \langle a_2 * h_2'|\right) \otimes \cdots \otimes \left(|a_m * h_m\rangle \langle a_m * h_m'|\right) \otimes \left(|a_1 * h_1 * a_2 * h_2 * \cdots * a_m * h_m\rangle \langle a_1 * h_1' * a_2 * h_2' * \cdots a_m * h_m'|\right).$$

Como $\langle a_1 * h_1' | a_1 * h_1\rangle = 1$ se e somente se $h_1 = h_1'$, e 0 caso contrário, temos

$$\rho^{A_2, \cdots, A_m, B} = \frac{1}{|H|^m} \sum_{h_1 \in H} \sum_{h_2, h_2' \in H} \cdots \sum_{h_m, h_m' \in H} \left(|a_2 * h_2\rangle \langle a_2 * h_2'|\right) \otimes \cdots \otimes \left(|a_m * h_m\rangle \langle a_m * h_m'|\right) \otimes \left(|a_1 * h_1 * a_2 * h_2 * \cdots * a_m * h_m\rangle \langle a_1 * h_1 * a_2 * h_2' * \cdots a_m * h_m'|\right).$$

Repetindo este processo, conseguimos obter a matriz de densidade reduzida referente ao último registrador.

$$\rho^B = \frac{1}{|H|^m} \sum_{h_1,h_2,\cdots,h_m \in H} |a_1 * h_1 * a_2 * h_2 * \cdots * a_m * h_m\rangle \langle a_1 * h_1 * a_2 * h_2 * \cdots a_m * h_m|$$

Para analisar o estado ρ^B, faça

$$S_g(a_1, \cdots, a_m) = \sum_{h_1 \in H} \cdots \sum_{h_m \in H} \begin{cases} 1 & \text{se } a_1 * h_1 * \cdots * a_m * h_m = g; \\ 0 & \text{caso contrário.} \end{cases}$$

Desta forma, temos $\rho^B = \sum_{g \in G} \frac{S_g(a_1,\cdots,a_m)}{|H|^m} |g\rangle \langle g|$. Observe que, como $\forall a, b \in G$, $ab = aba^{-1}a = b^a a$, temos

$$\begin{aligned} a_1 h_1 a_2 h_2 \cdots a_m h_m &= {h_1}^{a_1} a_1 a_2 h_2 \cdots a_m h_m \\ &= {h_1}^{a_1} {h_2}^{a_1 a_2} a_1 a_2 \cdots a_m H \\ &= {h_1}^{a_1} {h_2}^{a_1 a_2} \cdots {h_m}^{a_1 a_2 \cdots a_m} a_1 a_2 \cdots a_m \\ &= {h_1}^{g_1} {h_2}^{g_2} \cdots {h_m}^{g_m} g_m, \end{aligned}$$

onde $g_1 = a_1, g_2 = a_1 a_2, \cdots, g_m = a_1 a_2 \cdots a_m$. Logo podemos reescrever $S_g(a_1, \cdots, a_m)$ da seguinte forma

$$\begin{aligned} S_g(a_1, \cdots, a_m) &= \sum_{h_1 \in H} \cdots \sum_{h_m \in H} \begin{cases} 1 & \text{se } h_1^{g_1} * \cdots * h_m^{g_m} * g_m = g; \\ 0 & \text{caso contrário} \end{cases} \\ &= \sum_{h_1 \in H^{g_1}} \cdots \sum_{h_m \in H^{g_m}} \begin{cases} 1 & \text{se } h_1 * \cdots * h_m * g_m = g; \\ 0 & \text{caso contrário.} \end{cases} \end{aligned}$$

Ainda, temos

$$\begin{aligned} S_{g * g_m}(a_1, \cdots, a_m) &= \sum_{h_1 \in H^{g_1}} \cdots \sum_{h_m \in H^{g_m}} \begin{cases} 1 & \text{se } h_1 * \cdots * h_m = g; \\ 0 & \text{caso contrário.} \end{cases} \\ &= T_g(g_1, \cdots, g_m). \end{aligned}$$

Desta forma, o estado ρ^B pode ser descrito por

$$\begin{aligned} \rho^B &= \sum_{g \in G} \frac{S_{g * g_m}(a_1, \cdots, a_m)}{|H|^m} |g * g_m\rangle \langle g * g_m| \\ &= \sum_{g \in G} \frac{T_g(g_1, \cdots, g_m)}{|H|^m} |g * g_m\rangle \langle g * g_m| \end{aligned}$$

Isto torna-se particularmente útil para analisar a distância do traço entre ρ^B e I/G, dada por

$$D(\rho^B, I/G) = \frac{1}{2}\sum_{g\in G}\left|\frac{T_g(g_1,\cdots,g_m)}{|H|^m} - \frac{1}{|G|}\right|$$

Utilizando a propriedade da desigualdade do traço, temos

$$\begin{aligned} D(\rho^B, I/G) &\leq D(\rho^B, I/ncl_G(H)) + D(I/ncl_G(H), I/G)) \\ &= D(\rho^B, I/ncl_G(H)) + 1 - \frac{|ncl_G(H)|}{|G|} \end{aligned}$$

Observe que se $g \notin ncl_G(H)$, então $T_g(g_1,\cdots,g_m) = 0$, para quaisquer $g_1,\cdots,g_m \in G$. Pois, por definição $H_1^g H_2^g \cdots H_m^g \subseteq ncl_G(H)$, logo

$$D(\rho^B, I/ncl_G(H)) = \sum_{g\in ncl_G(H)}\left|\frac{T_g(g_1,\cdots,g_m)}{|H|^m} - \frac{1}{|ncl_G(H)|}\right|.$$

Dado que, com probabilidade ao menos $\frac{2}{3}$, $(g_1,\cdots,g_m)$ é um sequência boa, então, com probabilidade ao menos $\frac{2}{3}$, temos

$$D(\rho^B, I/ncl_G(H)) = \sum_{g\in ncl_G(H)}\left|\frac{T_g(g_1,\cdots,g_m)}{|H|^m} - \frac{1}{|ncl_G(H)|}\right| \leq \epsilon.$$

Portanto

$$D(\rho^B, I/G) \leq 1 - \frac{|ncl_G(H|)}{|G|} + \epsilon. \qquad \square$$

Quando H é um subgrupo normal de G, $ncl_G(H) = H$ e portanto

$$D(\rho^B, I/G) \leq 1 - \frac{|H|}{|G|} + \epsilon.$$

Ainda é possível mostrar que se $H \trianglelefteq G$, então $\rho^B = \frac{1}{|H|}\sum_{h\in H}|h * g_m\rangle\,|h * g_m\rangle$ e $D(\rho^B, I/G \leq 1 - \frac{|H|}{|G|})$.

Por outro lado, se $H \not\trianglelefteq G$, $H \neq ncl_G(H)$. Se $|ncl_G(H)|$ for suficientemente grande, a distancia do traço será pequena. Um exemplo é quando G é o grupo simétrico $\mathbb{S}_n$, para o qual, quando $n > 4$, o único subgrupo normal não trivial é o grupo alternante $\mathbb{A}_n$.

Corolário 4.2. Se G é o grupo simétrico $\mathbb{S}_n$ e existem um $m = \text{poly}(n)$ e um ϵ tais que ao menos $\frac{2}{3}$ das sequência são boas, então com probabilidade ao menos $\frac{2}{3}$, conseguimos produzir um estado ρ^B tal que

$$\begin{aligned} &\text{Se } |H| = 1, \text{ então} \quad D(\rho^B, I/G) = 1 - \frac{1}{|G|}; \\ &\text{Se } |H| \geq 2, \text{ então} \quad D(\rho^B, I/G) \leq \frac{1}{2} + \epsilon. \end{aligned}$$

Demonstração. Dado que o único subgrupo normal não trivial de $\mathbb{S}_n$, para $n > 4$, é o grupo alternante $\mathbb{A}_n$, se $|H| \geq 2$, então $\frac{|ncl_G(H)|}{|G|} \geq \frac{1}{2}$, pois $|\mathbb{S}_n| = 2|\mathbb{A}_n|$. □

Conjectura 4.3. Se G é o grupo alternante $\mathbb{A}_n$, então conseguimos, com alta probabilidade, produzir um estado ρ^B tal que

$$\text{Se } |H| = 1, \text{ então} \quad D(\rho^B, I/G) = 1 - \frac{1}{|G|};$$
$$\text{Se } |H| \geq 2, \text{ então} \quad D(\rho^B, I/G) \leq \frac{1}{2} + \epsilon.$$

4.2 INDÍCIOS DA CONJECTURA 4.3

Destacamos a semelhança do algoritmo apresentado na Seção 4.1 com um passeio aleatório em $R = \bigcup_{H^g \in Cl_G(H)} H^g$, no qual iniciamos no elemento identidade e e, a cada passo, escolhemos uniforme e independentemente ao acaso um elemento de R. Note que no algoritmo apresentado, a cada passo, escolhemos uniforme e independentemente ao acaso $H^g \in Cl_G(H)$.

Caso seja possível formalmente estabelecer uma relação entre os passos do algoritmo apresentado e tal passeio aleatório, poderíamos utilizar o resultado de Lulov e Pak (2002) (ver Teorema 4.4 a seguir), que é um desdobramento de análises presentes nos trabalhos de (Lulov, 1996; Pak, 1997; Aldous e Fill, 2002), para obtermos casos interessantes em que o algoritmo se aplica, como a Conjectura 4.3.

Teorema 4.4 (Lulov e Pak, 2002)**.** Sejam G um grupo finito e R um conjunto gerador simétrico para G. Então um passeio aleatório em R gera uma distribuição próxima da uniforme (em distância L_1) em $diam(G, R) \log |G|$ passos, onde $diam(G, R)$ é o diâmetro do grafo de Cayley $\Gamma(G, R)$. □

Uma vez que H^g, para todo $g \in G$, é um subgrupo, o conjunto $R = \bigcup_{H^g \in Cl(H)} H^g$ é um conjunto gerador simétrico para $ncl_G(H)$. Utilizando do fato que o diâmetro de um grafo k-regular de n vértices é $O(n/k)$, e o grafo de Cayley é um grafo $|R|$-regular quando R é simétrico, temos a relação

$$diam(G, R) = O(\frac{|G|}{|R|}).$$

Por definição, $|Cl(H)| \leq |R| \leq |H| \cdot |Cl(H)|$, logo $\frac{|G|}{|R|} \leq \frac{|G|}{|Cl(H)|}$. Pelo Lema 2.24 temos $|Cl(H)| = \frac{|G|}{|N_G(G)|}$ e portanto

$$diam(G, R) = O(|N_G(H)|).$$

Isto nos mostra que quanto menor o tamanho do conjunto $N_G(H)$ (H distante de ser um subgrupo normal), menor será o número de passos necessários para que o passeio aleatório gere uma distribuição próxima da uniforme.

Sendo possível estabelecer uma relação entre o algoritmo apresentado na Seção 4.1 e tal passeio aleatório, os seguintes indícios poder ser utilizados para demonstrar a Conjectura 4.3.

Uma vez que temos o Corolário 4.2, o problema GA torna-se um caso interessante de se analisar, pois independe da ordem de $ncl_G(H)$. Para o contexto de passeios aleatórios e $ncl_G(H) = \mathbb{A}_n$, o seguinte teorema nos auxilia a determinar o número de passos necessário.

Teorema 4.5 (Arad e Herzog, 2006)**.** Seja $C \neq \{e\}$ uma classe de conjugação de $\mathbb{A}_n$, então $C^{\lceil \frac{n}{2} \rceil} = \mathbb{A}_n$. □

Utilizando o Teorema 4.4, podemos observar que $diam(G, C) \leq \lceil \frac{n}{2} \rceil$ e portanto um passeio aleatório em R gera uma distribuição próxima da uniforme em $O(n \log |\mathbb{A}_n|) = O(n^2 \log n)$ passos. Deste modo, com $m = \text{poly}(n)$ passos conseguimos uma distribuição próxima da uniforme em um passeio aleatório em R.

Ainda para o caso do grupo alternante, temos o seguinte teorema que nos indica que existe uma sequência $(g_1, \cdots, g_m)$, com $m = \text{poly}(n)$, tal que todo elemento $g \in \mathbb{A}_n$ possui probabilidade positiva.

Teorema 4.6 (Liebeck et al., 2010; Liebeck et al., 2012)**.** Sejam G um grupo simples finito e $A \subseteq G$ tal que $|A| \geq 2$, então existe uma constante absoluta c tal que G é o produto de N conjugados de A, para algum $N \leq c \log |G|$. □

Apesar do Teorema 4.4 demonstrar a existência de uma sequência com pelo menos m elementos, neste caso m é linear $\log(|\mathbb{A}_n|)$ e não $\text{poly}\log(|\mathbb{A}_n|)$. Isto nos permite flexibilizar este termo em outras análises, na tentativa de alcançar o resultado desejado para a Conjectura 4.3.

4.3 DHSP E CLASSES DE CONHECIMENTO ZERO ESTATÍSTICO

Para o caso clássico de dHSP, é dado um circuito C_f de tamanho polinomial que recebe como entrada elementos de G e retorna strings de m bits, de modo que C_f computa a função f que oculta um subgrupo H em G.

Observando que o conjunto imagem $Im(f)$ da função f tem no máximo $|G|$ elementos, apontamos que o circuito C_f tem no máximo $|G|$ saídas diferentes possíveis. Utilizando esta observação, definimos por dHSP_{Im} como o caso particular de dHSP no qual podemos amostrar uniformemente elementos de G e de um conjunto $A_f \subseteq \{0,1\}^m$, de forma que $Im(f) \subseteq A_f$ e $|A_f| = |G|$. Embora este caso particular possa parecer artificial, observe que ele se aplica a casos importantes, como quando $G = \mathbb{Z}_n$ com adição ou multiplicação modular. O Corolário 4.8 explora mais alguns casos que satisfazem as condições de dHSP_{Im}. A Figura 4.2 apresenta o resultado obtido para dHSP_{Im}.

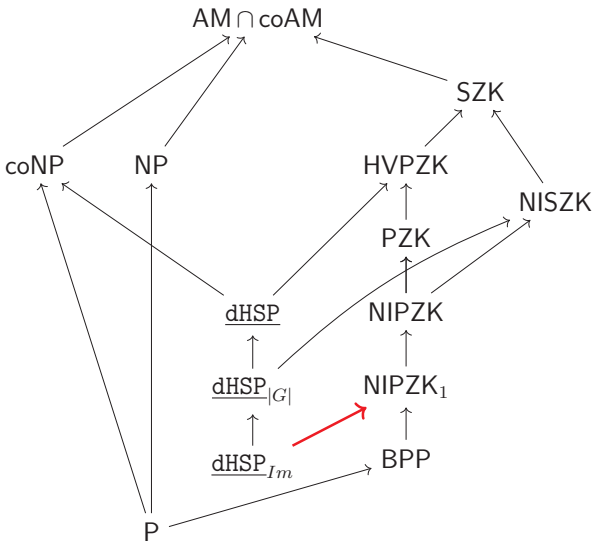


Figura 4.2: Relação entre as classes de complexidade estudas e o resultado obtido para dHSP. A seta A → B representa que: A ⊆ B se A e B são classes; A ∈ B, se A é um problema e B é uma classe; A é uma caso particular de B, se ambos A e B são problemas. Nosso resultado é destacado por uma seta vermelha em negrito.

Teorema 4.7 (colab. H. Hepp)**.** $\text{dHSP}_{Im} \in \mathsf{NIPZK}_1$.

Demonstração. Sejam $r \in A_f$, escolhido uniformemente ao acaso a CRS e $B_r = \{g \in G : f(g) = r\}$.

O provador P escolhe $g \in B_r$ se $B_r \neq \emptyset$, e $g \in G$ caso contrario, e envia para o verificador V, que aceita se $f(g) = r$ e rejeita caso contrário.

Como C_f é um circuito de tamanho polinomial, o verificador V computa $f(g)$ em tempo polinomial. Se $|H| = 1$, então V sempre aceita, garantindo a propriedade de completude. Agora, mostramos a propriedade de corretude. Se $|H| \geq 2$, então $|Im(f)|/|G| \leq 1/2$. Como $r \in A_f$ é escolhido uniformemente ao acaso, a probabilidade de $r \in Im(f)$ é no máximo $1/2$. Logo, a probabilidade de existir $g \in G$ que P possa enviar para V tal que $f(g) = r$ é no máximo $1/2$.

Para mostrar a propriedade de conhecimento zero, seja S o simulador que escolhe $g' \in G$ uniformemente ao acaso e computa $r' = f(g')$. Como $|H| = 1$, a função f é uma bijeção, obtendo $r' \in A_f$ também uniformemente ao acaso. Portanto, as transcrições $\langle C_f, r', g' \rangle$ do simulador S e as transcrições $\langle C_f, r, g \rangle$ do protocolo são identicamente distribuídas sempre que $|H| = 1$. □

Os Corolários 4.8 e 4.9 seguem imediatamente do Teorema 4.7.

Corolário 4.8 (colab. H. Hepp)**.** Se amostramos uniformemente ao acaso elementos de G, e se A_f é o conjunto das primeiras $|G|$ strings em ordem lexicográfica, então $\mathtt{dHSP} \in \mathsf{NIPZK}_1$. □

Corolário 4.9 (colab. H. Hepp)**.** Se amostramos uniformemente ao acaso elementos de G, e o circuito C_f tem como saída strings de tamanho $\log |G|$, então $\mathtt{dHSP} \in \mathsf{NIPZK}_1$. □

4.4 UM PROBLEMA COMPLETO PARA NIPZK COM COMPLETUDE PERFEITA

A classe NIPZK foi definida por Malka (2008), no qual também foi apresentado o problema Uniforme (UN, de *Uniform*) e mostrado que este é completo para NIPZK. Ainda para a classe NIPZK, foi mostrado em Morrison e Groce (2020) que existe um oráculo O tal que $\mathsf{NIPZK}^O \not\subseteq \mathsf{BQP}^O$.

A seguir, apresentamos a definição do problema UN, onde U_m denota a distribuição uniforme sobre todas as strings de m bits, e $sup(X) = \{y \in \{0,1\}^m : \Pr[X = y] \neq 0\}$ é o *suporte* da distribuição X.

Definição 4.10 (Problema Uniforme)**.** Dado um circuito polinomial $C : \{0,1\}^n \to \{0,1\}^{m+1}$, onde n e m são inteiros não negativos, que codifica um distribuição de probabilidade X, decidir entre:

$$\mathtt{UN}_Y = \{X : \Delta(X[1..m], U_m) = 0 \text{ e } \Pr[X[m+1] = 1] \geq 2/3\};$$
$$\mathtt{UN}_N = \{X : |sup(X) \cap \{0,1\}^m 1| \leq 2^m/3\},$$

prometido que vale um dos casos.

Embora não tenha explicitamente provado por Malka (2008), foi sugerido que uma versão restrita do problema também seria completo para a classe NIPZK_1.

Definimos o problema Uniforme-ou-Menor (US, de *Uniform-or-Small*). Este problema também foi estudado por Dixon et al. (2020), que mostrou que existe um oráculo O relativo ao qual US não está na classe probabilística SBP (*Small Bounded-Error Probability*).

Definição 4.11 (Problema Uniforme-ou-Pequeno)**.** Dado um circuito polinomial $C : \{0,1\}^n \to \{0,1\}^m$ que codifica uma distribuição X, decidir entre:

$$\texttt{US}_Y = \{X : \Delta(X, U_m) = 0\};$$
$$\texttt{US}_N = \{X : |sup(X)| \leq 2^m/3\},$$

prometido que vale um dos casos.

Teorema 4.12 (colab. H. Hepp)**.** US é completo para NIPZK_1.

Prova de que US $\in \mathsf{NIPZK}_1$.

Faça $r \in \{0,1\}^m$ escolhido uniformemente ao acaso como a CRS e $B_r = \{\pi \in \{0,1\}^n : X(\pi) = r\}$. O provador P escolhe $\pi \in B_r$ se $B_r \neq \emptyset$, e $\pi \in \{0,1\}^n$ caso contrario, e envia para o verificador V, que aceita se $X(\pi) = r$ e rejeita caso contrário.

Como C é um circuito polinomial, o verificador V computa $X(\pi)$ em tempo polinomial. Se $X \in \texttt{US}_Y$, então V sempre aceita, garantindo a propriedade de completude. Agora, para a propriedade de corretude, se $X \in \texttt{US}_N$ então, como $|sup(X)| \leq 2^m/3$ e r é escolhido uniformemente ao acaso, a probabilidade de $r \in sup(X)$ é no máximo $1/3$. Portanto, a probabilidade de que existe $\pi \in \{0,1\}^n$ tal que $X(\pi) = r$ é no máximo $1/3$.

Para garantir a propriedade de conhecimento zero, faça $X \in \texttt{US}_Y$ e S o simulador que escolhe $\pi' \in \{0,1\}^n$ uniformemente ao acaso e computa $r' = X(\pi')$. Quando $X \in \texttt{US}_Y$, obtemos $r' \in \{0,1\}^m$ uniformemente ao acaso. Portanto, as transcrições $\langle X, r', \pi' \rangle$ do simulador S e as transcrições $\langle X, r, \pi \rangle$ do protocolo são identicamente distribuídas sempre que $X \in \texttt{US}_Y$. □

Prova de que US é NIPZK_1*-difícil.*

Sejam $\Pi = \langle \Pi_Y, \Pi_N \rangle$ um problema NIPZK_1 e (P, V) seu respectivo protocolo. Para o protocolo (P, V) de Π, sejam r sua CRS e S seu simulador, onde r_S é string aleatória de entrada de S.

Para mostrar que Π possui uma redução Karp polinomial para US iremos mostrar que existe uma máquina de Turing polinomial que, na entrada $x \in \Pi_Y \cup \Pi_N$, devolve um circuito $C : \{0,1\}^{|r_S|} \to \{0,1\}^{|r|}$ que codifica uma distribuição X, tal que se $x \in \Pi_Y$, então $X \in \texttt{US}_Y$, e se $x \in \Pi_N$, então $X \in \texttt{US}_N$. Dado x, o circuito C é construído de modo que, na entrada uma string r_S, emula a computação de S em x sob a aleatoriedade r_S, produzindo a visão $\langle x, r, \pi \rangle$, onde r é a CRS de (P, V) e π é a mensagem enviada de P para V. Então, C devolve r se $V(x, r, \pi) = \texttt{accept}$, e $0^{|r|}$ caso contrário.

Agora, nós analisamos a redução. Se $x \in \Pi_Y$, então $V(x, r, \pi) = \texttt{accept}$. Por construção, C devolve r. Como r_S é a CRS para US, uniformemente distribuída, o simulador S retorna a string r também de maneira uniformemente distribuída. Deste modo C codifica uma distribuição uniforme X. Se $x \in \Pi_N$, mostraremos que $|sup(X)| \leq 2^{|r|}/3$. Pela condição de corretude, a probabilidade do verificador $V(x)$ aceitar é menor que $1/3$. Logo, por construção, em mais de $2/3$ dos casos, temos a string $0^{|r|}$ como saída de C. Portanto, $|sup(X)| \leq 2^{|r|}/3$. □

Obviamente, a expressão $|sup(X)| \leq 2^m/3$ na definição de US (Definição 4.11) pode ser substituída por $|sup(X)| \leq \beta \cdot 2^m$, para qualquer constante positiva $\beta < 1$, e o Teorema 4.12 também será válido. Em Dixon et al. (2020) US foi definido de maneira equivalente, com $\beta = 1/2$.

5 CONSIDERAÇÕES FINAIS

Neste trabalho, realizamos uma análise sobre um algoritmo quântico de multiplicação de coclasses para a versão de decisão do problema do subgrupo oculto (dHSP). Isso nos permitiu identificar casos específicos de grupos não abelianos nos quais a multiplicação de um número polinomial de coclasses em sobreposição resulta em um estado com distância de traço, em relação ao estado maximamente misto, próxima de 1 quando o subgrupo oculto é o subgrupo trivial, e distante de 1 caso contrário.

Ao analisar o algoritmo apresentado, observamos que o resultado sobre a distância do traço é bastante distinto quando o subgrupo oculto é um subgrupo normal, sendo, neste caso, relacionado à ordem do subgrupo oculto, e não distante de 1, quando o subgrupo oculto não é o subgrupo trivial. Além disso, foram apresentados indícios de que para o grupo alternante $\mathbb{A}_n$, resultados mais robustos podem ser enunciados.

Atualmente, estamos investigando algoritmos para estimar a distância de traço, de modo que possamos aplicar nos resultados obtidos e, possivelmente, encontrar casos particulares de dHSP que podem ser resolvidos por meio da multiplicação de coclasses aleatórias. Uma vez que o algoritmo apresentado possui semelhanças com um passeio aleatório, também consideramos, como continuidade deste trabalho, o estudo de passeios quânticos em grafos, os quais são similares a passeios aleatórios em grafos, mas em um contexto quântico.

Ao investigar provas não-interativas de conhecimento zero, mostramos condições suficientes para que casos particulares de dHSP admitam provas não-interativas de conhecimento zero perfeito com completude perfeita, demonstrando que esses casos particulares pertencem à classe NIPZK_1.

O estudo da relação entre as classes de protocolos não-interativos e a classe BQP também é um tópico de interesse que pode ser aprofundado em pesquisas futuras. Uma vez que se acredita que a classe SZK não está contida em BQP e que as classes não-interativas são menores que SZK, torna-se interessante investigar se existem algoritmos quânticos polinomiais para problemas que se encontram nessas classes de complexidade.

REFERÊNCIAS

Ablayev, F. e Vasiliev, A. (2009). Algorithms for quantum branching programs based on fingerprinting. *arXiv preprint arXiv:0911.2317*.

Aldous, D. e Fill, J. (2002). Reversible markov chains and random walks on graphs.

Arad, Z. e Herzog, M. (2006). *Products of conjugacy classes in groups*, volume 1112. Springer.

Arora, S. e Barak, B. (2009). *Computational complexity: a modern approach*. Cambridge University Press.

Babai, L. (2016). Graph isomorphism in quasipolynomial time. Em *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, páginas 684–697.

Babai, L. e Szemerédi, E. (1984). On the complexity of matrix group problems I. Em *25th Annual Symposium on Foundations of Computer Science, 1984*, páginas 229–240. IEEE.

Beals, R. (1997). Quantum computation of fourier transforms over symmetric groups. Em *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, páginas 48–53.

Cheung, K. K. e Mosca, M. (2001). Decomposing finite abelian groups. *arXiv preprint cs/0101004*.

Childs, A. M. (2021). Lecture notes on quantum algorithms. *Lecture notes at University of Maryland*.

Childs, A. M. e Van Dam, W. (2010). Quantum algorithms for algebraic problems. *Reviews of Modern Physics*, 82(1):1.

Cleve, R. (1994). A note on computing fourier transforms by quantum programs. *preprint*.

Cleve, R. e Watrous, J. (2000). Fast parallel circuits for the quantum fourier transform. Em *Proceedings 41st Annual Symposium on Foundations of Computer Science*, páginas 526–536. IEEE.

Coppersmith, D. (1994). An approximate fourier transform useful in quantum factoring. *IBM Research Report*, páginas RC–19642.

Costa, A. F., Hepp, H., da Silva, M. V. e Zatesko, L. M. (2023). The hidden subgroup problem and non-interactive perfect zero-knowledge proofs. Em *Anais do VIII Encontro de Teoria da Computação*, páginas 99–103. SBC.

Dixon, P., Gayen, S., Pavan, A. e Vinodchandran, N. (2020). Perfect zero knowledge: New upperbounds and relativized separations. Em *Theory of Cryptography: 18th International Conference, TCC 2020, Durham, NC, USA, November 16–19, 2020, Proceedings, Part I*, páginas 684–704. Springer.

Ettinger, M., Høyer, P. e Knill, E. (2004). The quantum query complexity of the hidden subgroup problem is polynomial. *Information Processing Letters*, 91(1):43–48.

Friedl, K., Magniez, F., Santha, M. e Sen, P. (2003). Quantum testers for hidden group properties. Em *International Symposium on Mathematical Foundations of Computer Science*, páginas 419–428. Springer.

Fulton, W. e Harris, J. (2013). *Representation theory: a first course*, volume 129. Springer Science & Business Media.

Gavinsky, D. (2004). Quantum solution to the hidden subgroup problem for poly-near-hamiltonian groups. *Quantum Information & Computation*, 4(3):229–235.

Goldreich, O., Sahai, A. e Vadhan, S. (1998). Honest-verifier statistical zero-knowledge equals general statistical zero-knowledge. Em *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, páginas 399–408.

Gonçalves, D. N., Fernandes, T. D. e Cosme, C. (2017). An efficient quantum algorithm for the hidden subgroup problem over some non-abelian groups. *TEMA (São Carlos)*, 18:215–223.

Grigni, M., Schulman, J., Vazirani, M. e Vazirani, U. (2004). Quantum mechanical algorithms for the nonabelian hidden subgroup problem. *Combinatorica*, 24(1):137–154.

Hales, L. e Hallgren, S. (2000). An improved quantum fourier transform algorithm and applications. Em *Proceedings 41st Annual Symposium on Foundations of Computer Science*, páginas 515–525. IEEE.

Hales, L. R. (2002). *The quantum Fourier transform and extensions of the abelian hidden subgroup problem*. University of California, Berkeley.

Hallgren, S., Moore, C., Rötteler, M., Russell, A. e Sen, P. (2010). Limitations of quantum coset states for graph isomorphism. *Journal of the ACM (JACM)*, 57(6):1–33.

Hallgren, S., Russell, A. e Ta-Shma, A. (2003). The hidden subgroup problem and quantum computation using group representations. *SIAM Journal on Computing*, 32(4):916–934.

Hayashi, M., Kawachi, A. e Kobayashi, H. (2008). Quantum measurements for hidden subgroup problems with optimal sample complexity. *Quantum Information & Computation*, 8(3):345–358.

Helfgott, H. A., Bajpai, J. e Dona, D. (2017). Graph isomorphisms in quasi-polynomial time. *arXiv preprint arXiv:1710.04574*.

Holt, D. F., Eick, B. e O'Brien, E. A. (2005). *Handbook of computational group theory*. CRC Press.

Hoyer, P. (1997). Efficient quantum transforms. *arXiv preprint quant-ph/9702028*.

Kempe, J. e Shalev, A. (2004). The hidden subgroup problem and permutation group theory. *arXiv preprint quant-ph/0406046*.

Kitaev, A. Y. (1995). Quantum measurements and the abelian stabilizer problem. *arXiv preprint quant-ph/9511026*.

Ladner, R. E. (1975). On the structure of polynomial time reducibility. *Journal of the ACM (JACM)*, 22(1):155–171.

Liebeck, M. W., Nikolov, N. e Shalev, A. (2010). A conjecture on product decompositions in simple groups. *Groups, Geometry, and Dynamics*, 4(4):799–812.

Liebeck, M. W., Nikolov, N. e Shalev, A. (2012). Product decompositions in finite simple groups. *Bulletin of the London Mathematical Society*, 44(3):469–472.

Lulov, N. e Pak, I. (2002). Rapidly mixing random walks and bounds on characters of the symmetric group. *Journal of Algebraic Combinatorics*, 16:151–163.

Lulov, N. A. M. (1996). *Random walks on the symmetric group generated by conjugacy classes*. Harvard University.

Lund, C., Fortnow, L., Karloff, H. e Nisan, N. (1992). Algebraic methods for interactive proof systems. *Journal of the ACM (JACM)*, 39(4):859–868.

Malka, L. (2008). How to achieve perfect simulation and a complete problem for non-interactive perfect zero-knowledge. Em *Theory of Cryptography: Fifth Theory of Cryptography Conference, TCC 2008, New York, USA, March 19-21, 2008. Proceedings 5*, páginas 89–106. Springer.

Moore, C. (2018). The hunt for a quantum algorithm for graph isomorphism. `https://www.youtube.com/watch?v=m8uXQmo2A0U`. Acessado em 05/04/2022.

Moore, C., Rockmore, D. e Russell, A. (2006). Generic quantum fourier transforms. *ACM Transactions on Algorithms (TALG)*, 2(4):707–723.

Moore, C., Rockmore, D., Russell, A. e Schulman, L. J. (2007). The power of strong fourier sampling: Quantum algorithms for affine groups and hidden shifts. *SIAM Journal on Computing*, 37(3):938–958.

Moore, C. e Russell, A. (2005). The symmetric group defies strong fourier sampling: Part ii. *arXiv preprint quant-ph/0501066*.

Moore, C., Russell, A. e Schulman, L. J. (2005). The symmetric group defies strong fourier sampling: Part i.

Morrison, B. e Groce, A. (2020). Oracle separations between quantum and non-interactive zero-knowledge classes. *Information Processing Letters*, 154:105864.

Mosca, M. (1999). *Quantum computer algorithms*. Tese de doutorado, University of Oxford. 1999.

Nielsen, M. A. e Chuang, I. (2002). Quantum computation and quantum information.

Pak, I. (1997). *Random walks on permutation: strong uniform time approach*. Tese de doutorado, Ph. D. Thesis, Harvard University.

Pearce-Crump, E. (2020). *Quantum Computing*. Tese de doutorado, Imperial College London.

Püschel, M., Rötteler, M. e Beth, T. (1999). Fast quantum fourier transforms for a class of non-abelian groups. Em *International Symposium on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes*, páginas 148–159. Springer.

Radhakrishnan, J., Rötteler, M. e Sen, P. (2005). On the power of random bases in fourier sampling: Hidden subgroup problem in the heisenberg group. Em *International Colloquium on Automata, Languages, and Programming*, páginas 1399–1411. Springer.

Regev, O. (2004). Quantum computation and lattice problems. *SIAM Journal on Computing*, 33(3):738–760.

Rotman, J. J. (2005). *A First Course in Abstract Algebra*. Pearson, Upper Saddle River, NJ, 3 edition.

Sdroievski, N. M., da Silva, M. V. e Vignatti, A. L. (2019a). Conhecimento zero estatístico e reduções eficientes para o problema mktp. Em *Anais do XXXII Concurso de Teses e Dissertações*. SBC.

Sdroievski, N. M., da Silva, M. V. e Vignatti, A. L. (2019b). The hidden subgroup problem and MKTP. *Theoretical Computer Science*, 795:204–212.

Sen, P. (2006). Random measurement bases, quantum state distinction and applications to the hidden subgroup problem. Em *21st Annual IEEE Conference on Computational Complexity (CCC'06)*, páginas 14–pp. IEEE.

Seress, Á. (2003). *Permutation group algorithms*. Número 152. Cambridge University Press.

Serre, J.-P. (1977). *Linear representations of finite groups*, volume 42. Springer.

Shamir, A. (1992). Ip= pspace. *Journal of the ACM (JACM)*, 39(4):869–877.

Shor, P. W. (1994). Algorithms for quantum computation: discrete logarithms and factoring. Em *Proceedings 35th annual symposium on foundations of computer science*, páginas 124–134. IEEE.

Shor, P. W. (1997). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509.

Simon, D. R. (1997). On the power of quantum computation. *SIAM journal on computing*, 26(5):1474–1483.

Steinberg, B. (2009). Representation theory of finite groups. *School of Mathematics and Statistics, Carleton University*.

Vadhan, S. P. (1999). *A study of statistical zero-knowledge proofs*. Tese de doutorado, Massachusetts Institute of Technology.

Wang, F. (2010). The hidden subgroup problem. *arXiv preprint arXiv:1008.0010*.

West, D. B. et al. (2001). *Introduction to graph theory*, volume 2. Prentice hall Upper Saddle River.

APÊNDICE A – LEMATA

.

Dedicamos este apêndice a apresentar e demonstrar Proposições e lemas que utilizamos nos capítulos principais da dissertação.

Lema A.1. Seja G um grupo e $x \in G$, então

$$\hat{R}(x) = F_G R(x) F_G^{\dagger} = \sum_{\sigma \in \hat{G}} |\sigma\rangle \langle\sigma| \otimes \mathbb{I}_{d_\sigma} \otimes \Big[\sigma(x)\Big]^* \tag{A.1}$$

Demonstração. A demonstração apresentada a seguir foi retirada e adaptada de (Pearce-Crump, 2020).

$$\begin{aligned} \hat{R}(x) &= F_G R(x) F_G^{\dagger} \\ &= \left(\sum_{z \in G} |\hat{z}\rangle \langle z| \right) R(x) \left(\sum_{y \in G} |\hat{y}\rangle \langle y| \right) \\ &= \sum_{z,y \in G} |\hat{z}\rangle \langle z|yx^{-1}\rangle \langle \hat{y}| \\ &= \sum_{y \in G} |\widehat{yx^{-1}}\rangle \langle \hat{y}| \end{aligned} \tag{A.2}$$

Logo

$$\hat{R}(x) = \sum_{y \in G} \sum_{\sigma,\sigma' \in \hat{G}} \sum_{j,k=1}^{d_\sigma} \sum_{j',k'=1}^{d_{\sigma'}} \frac{\sqrt{d_\sigma d_{\sigma'}}}{|\,G\,|} \Big[\sigma(yx^{-1})\Big]_{j,k} \Big[\sigma'(y)\Big]^*_{j',k'} |\sigma, j, k\rangle \langle \sigma', j', k'| \tag{A.3}$$

Como

$$\Big[\sigma(yx^{-1})\Big]_{j,k} = \sum_{l}^{d_\sigma} \Big[\sigma(y)\Big]_{j,l} \Big[\sigma(x^{-1})\Big]_{l,k}$$

Então

$$\hat{R}(x) = \sum_{y \in G} \sum_{\sigma,\sigma' \in \hat{G}} \sum_{j,k,l=1}^{d_\sigma} \sum_{j',k'=1}^{d_{\sigma'}} \frac{\sqrt{d_\sigma d_{\sigma'}}}{|\,G\,|} \Big[\sigma(y)\Big]_{j,l} \Big[\sigma(x^{-1})\Big]_{l,k} \Big[\sigma'(y)\Big]^*_{j',k'} |\sigma, j, k\rangle \langle \sigma', j', k'| \tag{A.4}$$

Pela ortogonalidade de reduções irredutíveis temos

$$\frac{d_\sigma}{|\,G\,|} \sum_{y \in G} \Big[\sigma(y)\Big]_{j,l} \Big[\sigma'(y)\Big]^*_{j',k'} = \delta_{\sigma,\sigma'} \delta_{j,l} \delta_{j',k'} \tag{A.5}$$

onde $\delta_{\sigma,\sigma'} = 1$ se σ é uma representação isomórfica a σ', caso contrário $\delta_{\sigma,\sigma'} = 0$.

Substituindo na Equação (A.4) obtemos

$$\hat{R}(x) = \sum_{\sigma \in \hat{G}} \sum_{j,k,l=1}^{d_\sigma} \left[\sigma(x^{-1})\right]_{l,k} |\sigma, j, k\rangle \langle \sigma, j, l| \tag{A.6}$$

Como

$$\left[\sigma(x^{-1})\right]_{l,k} = \left[\sigma(x)\right]^{-1}_{l,k} = \left[\sigma(x)\right]^{\dagger}_{l,k} = \left[\sigma(x)\right]^{*}_{k,l}$$

Portanto

$$\begin{aligned}
\hat{R}(x) &= \sum_{\sigma \in \hat{G}} \left(|\sigma\rangle \langle \sigma| \otimes \left(\sum_{j=1}^{d_\sigma} |j\rangle \langle j| \right) \otimes \left(\sum_{k=1}^{d_\sigma} \sum_{l=1}^{d_\sigma} \left[\sigma(x)\right]^{*}_{k,l} |k\rangle \langle l| \right) \right) \\
&= \sum_{\sigma \in \hat{G}} \left(|\sigma\rangle \langle \sigma| \otimes \mathbb{I}_{d_\sigma} \otimes \left[\sigma(x)\right]^{*} \right)
\end{aligned}$$

□

APÊNDICE B – TEORIA DE REPRESENTAÇÃO

Para melhor compreender representações irredutíveis, observamos que toda representação irredutível é *indecomponível*, isto é, não pode ser escrita pela *soma direta* de duas outras representações.

Dizemos que um espaço vetorial V é a soma direta de dois subespaços vetoriais V_1 e V_2 de V, denotada por $V = V_1 \oplus V_2$, se todo $v \in V$ pode ser escrito unicamente por $v = v_1 + v_2$, onde $v_1 \in V_1$ e $v_2 \in v_2$. A soma direta de duas representações $\sigma_1 : G \to GL(V_1)$ e $\sigma_2 : G \to GL(V_2)$, que denotamos como $\sigma_1 \oplus \sigma_2 : G \to GL(V_1 \oplus V_2)$, para todos os elementos $g \in G$, vetores $v_1 \in V_1$, e vetores $v_2 \in V_2$, é definida da seguinte forma:

$$(\sigma_1 \oplus \sigma_2)(g)(v_1, v_2) = (\sigma_1(g)(v_1), \sigma_2(g)(v_2)).$$

Se escolhermos bases para os espaços vetoriais V_1 e V_2, então para todo $g \in G$, temos

$$\left[(\sigma_1 \oplus \sigma_2)(g)\right] = \begin{bmatrix} \left[\sigma_1(g)\right] & 0 \\ 0 & \left[\sigma_2(g)\right]. \end{bmatrix}.$$

Uma representação $\sigma : G \to GL(V)$ é dita *indecomponível* se para quaisquer duas representações $\sigma_1 : G \to GL(V_1)$ e $\sigma_2 : G \to GL(V_2)$, onde V_1 e V_2 são subespaços não triviais de V, então $\sigma \ncong \sigma_1 \oplus \sigma_2$. Isto é, se não existem duas representações tais que sua soma direta é isomorfa a σ.

O Teorema B.1 introduz o Lema de Schur, um resultado importante na teoria de representações.

Teorema B.1 (Lema de Schur)**.** Sejam $\sigma_1 : G \to GL(V_1)$ e $\sigma_2 : G \to GL(V_2)$ duas representações irredutíveis de um grupo G e seja $T : V_1 \to V_2$ uma transformação linear tal que $T \circ \sigma_1(g) = \sigma_2(g) \circ T$ para todo $g \in G$.

- Se $\sigma_1 \ncong \sigma_2$ então $T = 0$.
- Se $V_1 = V_2$ e $\sigma_1 = \sigma_2$ então T é um múltiplo escalar da identidade. □

Utilizando o Lema se Schur, é possível inferir os teorema da ortogonalidade de representações irredutíveis e o teorema da ortogonalidade de caracteres.

Teorema B.2 (Ortogonalidade de representações irredutíveis)**.** Sejam σ_1, σ_2 duas representações irredutíveis de um grupo G, então

$$\frac{d_{\sigma_1}}{|G|} \sum_{g \in G} \sigma_1(g)^*_{j_1,k_1} \sigma_2(g)_{j_2,k_2} = \delta_{\sigma_1,\sigma_2} \delta_{j_1,j_2} \delta_{k_1,k_2}$$

onde $\delta_{\sigma_1,\sigma_2} = 1$ se $\sigma_1 \cong \sigma_2$, e $\delta_{\sigma_1,\sigma_2} = 0$ se $\sigma_1 \ncong \sigma_2$. □

Teorema B.3 (Ortogonalidade de caracteres)**.** Sejam σ_1, σ_2 duas representações irredutíveis de um grupo G, então

$$\frac{1}{|G|} \sum_{g \in G} \chi_{\sigma_1}(g)^* \chi_{\sigma_2}(g) = \delta_{\sigma_1,\sigma_2}$$

onde $\delta_{\sigma_1,\sigma_2} = 1$ se $\sigma_1 \cong \sigma_2$, e $\delta_{\sigma_1,\sigma_2} = 0$ se $\sigma_1 \ncong \sigma_2$. □