

UNIVERSIDADE FEDERAL DO PARANÁ

HENRIQUE HEPP

COMPUTAÇÃO QUÂNTICA SEM COLAPSO E
PROBLEMAS DE CONHECIMENTO ZERO

CURITIBA PR

2025

HENRIQUE HEPP

COMPUTAÇÃO QUÂNTICA SEM COLAPSO E
PROBLEMAS DE CONHECIMENTO ZERO

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

Área de concentração: *Computação*.

Orientadores: Murilo V. G. da Silva e Leandro M. Zatesko.

CURITIBA PR

2025

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

Hepp, Henrique

Computação quântica sem colapso e problemas de conhecimento zero /
Henrique Hepp. – Curitiba, 2025.
1 recurso on-line : PDF.

Tese (Doutorado) - 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. Complexidade computacional. 2. Computação quântica. 3. Separação
oracular. 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: Elias Barbosa da Silva CRB-9/1894

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 tese de Doutorado de **HENRIQUE HEPP**, intitulada: **Computação Quântica sem Colapso e Problemas de Conhecimento Zero**, 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 doutor 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, 24 de Abril de 2025.

Assinatura Eletrônica

30/04/2025 20:37:11.0

MURILO VICENTE GONÇALVES DA SILVA

Presidente da Banca Examinadora

Assinatura Eletrônica

05/05/2025 11:24:58.0

FRANKLIN DE LIMA MARQUEZINO

Avaliador Externo (UNIVERSIDADE FEDERAL DO RIO DE JANEIRO)

Assinatura Eletrônica

29/04/2025 17:22:10.0

ANDRÉ LUIZ PIRES GUEDES

Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica

05/05/2025 14:40:40.0

UEVERTON DOS SANTOS SOUZA

Avaliador Externo (UNIVERSIDADE FEDERAL FLUMINENSE)

Assinatura Eletrônica

29/04/2025 18:08:52.0

NICOLLAS MOCELIN SDROIEVSKI

Avaliador Externo (UNIVERSIDADE ESTADUAL DO PARANÁ)

Assinatura Eletrônica

28/04/2025 15:09:56.0

LEANDRO MIRANDA ZATESKO

Coordenador(a) (UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ)

*À minha família e,
in memoriam, a meu pai.*

AGRADECIMENTOS

- A Deus.
- Aos meus orientadores de doutorado Leandro e Murilo.
- A todas pessoas que direta ou indiretamente contribuíram para a elaboração deste trabalho.

RESUMO

A classe de problemas para os quais existem algoritmos quânticos eficientes é denotada por BQP (*Bounded-error Quantum Polynomial*). Uma abordagem comum em teoria da computação para a compreensão dos limites de uma classe de complexidade é esclarecer relações desta com outras subclasses e superclasses próximas. Uma superclasse de BQP que tem sido foco de pesquisa recentemente é a classe naCQP (*non-adaptive Collapse-free Quantum Polynomial Time*), também conhecida como PDQP (*Product Dynamical Quantum Polynomial time*), introduzida por Aaronson et al. (2016), a qual corresponde a problemas que podem ser resolvidos eficientemente por um algoritmo quântico em um modelo hipotético de computação em que é possível realizar medições sem provocar o colapso do estado quântico. Provas interativas de conhecimento zero têm recebido muita atenção nas últimas décadas, em virtude de suas aplicações importantes em segurança de informação. A classe dos problemas que admitem provas interativas de conhecimento zero estatístico na computação clássica é a classe SZK (*Statistical Zero Knowledge*). Embora a relação entre BQP e SZK ainda seja desconhecida, Aaronson et al. (2016) mostraram que SZK está contida em naCQP. Uma das abordagens para investigação de separações entre classes é o estudo de suas relações relativas a um oráculo. Aaronson et al. (2016) mostraram que existe um oráculo A para o qual $NP^A \not\subseteq naCQP^A$. Nós provamos que existe um oráculo A para o qual $P^A = BQP^A = SZK^A = naCQP^A \neq (UP \cap coUP)^A = EXP^A$, onde UP (*Unambiguous Polynomial time*) é uma subclasse importante de NP. Ainda, mostramos que, relativo a um oráculo A escolhido uniformemente de modo aleatório, temos $(UP \cap coUP)^A \not\subseteq naCQP^A$ com probabilidade 1. Nossos resultados ampliam resultados de Bennett et al. (1997), Fortnow e Rogers (1999), Tamon e Yamakami (2001) e Aaronson et al. (2016). A classe SZK possui muitas subclasses notáveis, como NISZK e NIPZK₁. A classe NISZK é associada a sistemas de prova com somente um envio de mensagem do provador para o verificador. Para NIPZK₁, subclasse de NISZK, ainda é requerido conhecimento zero perfeito e que as instâncias positivas do problema sejam aceitas pelo verificador com probabilidade igual a 1. Mostramos um problema completo para NIPZK₁ e também que uma restrição não trivial do problema de subgrupo oculto está em NIPZK₁. Além disso, mostramos uma restrição não trivial de um problema completo para NISZK que está em BQP.

Palavras-chave: complexidade computacional. computação quântica. separação oracular.

ABSTRACT

The class of problems for which there are efficient quantum algorithms is denoted by BQP (*Bounded-error Quantum Polynomial*). A common approach in theoretical computer science to understand the limitations of a complexity class is to clarify its relationships with other nearby subclasses and superclasses. A superclass of BQP that has been the focus of research recently is the class naCQP (*non-adaptive Collapse-free Quantum Polynomial Time*), also known as PDQP (*Product Dynamical Quantum Polynomial time*), introduced by Aaronson et al. (2016), which corresponds to problems that can be solved efficiently by a quantum algorithm in a non-adaptive computing model where measurements can be made without causing the quantum state to collapse. Interactive statistical zero-knowledge proofs have received much attention in recent decades because of their important applications to information security. The class of problems that admit statistical zero-knowledge interactive proofs in classical computing is the class SZK (*Statistical Zero Knowledge*). Although the relationship between BQP and SZK is still unknown, Aaronson et al. (2016) have shown that SZK is contained in naCQP. A common approach for investigating separation between classes is the study of their relations relative to an oracle. Aaronson et al. (2016) showed that there is an oracle A for which $\text{NP}^A \not\subseteq \text{naCQP}^A$. We prove that there exists an oracle A for which $\text{P}^A = \text{BQP}^A = \text{SZK}^A = \text{naCQP}^A \neq (\text{UP} \cap \text{coUP})^A = \text{EXP}^A$, where UP (*Unambiguous Polynomial time*) is an important subclass of NP. Furthermore, we show that for an oracle A chosen uniformly at random, we have $(\text{UP} \cap \text{coUP})^A \not\subseteq \text{naCQP}^A$ with probability 1. Our results extend results from Bennett et al. (1997), Fortnow e Rogers (1999), Tamon e Yamakami (2001) and Aaronson et al. (2016). The class SZK has many notable subclasses, such as NISZK and NIPZK₁. The NISZK class is associated with proof systems with only one message sent from the prover to the verifier. For NIPZK₁, a subclass of NISZK, we still require perfect zero knowledge and that the positive instances of the problem are accepted by the verifier with probability equal to one. We show a complete problem for NIPZK₁ and also that a non-trivial restriction of the hidden subgroup problem is in NIPZK₁. Furthermore, we show a non-trivial constraint of a complete problem for NISZK which is in BQP.

Keywords: computational complexity. quantum computing. oracle separation.

SUMÁRIO

1	INTRODUÇÃO	8
2	PRELIMINARES	15
2.1	ÁLGEBRA LINEAR	15
2.2	MECÂNICA QUÂNTICA	20
2.3	CIRCUITOS QUÂNTICOS.	27
2.4	CONCEITOS ELEMENTARES DE COMPLEXIDADE COMPUTACIONAL. .	30
2.5	CLASSES DE PROVAS INTERATIVAS E DE CONHECIMENTO ZERO ESTATÍSTICO	34
3	REVISÃO DA LITERATURA	40
3.1	PROBLEMAS PARA CLASSES DE CONHECIMENTO ZERO ESTATÍSTICO	40
3.2	COMPUTAÇÃO QUÂNTICA SEM COLAPSO.	45
3.3	SEPARAÇÃO ORACULAR ENTRE $UP \cap coUP$ E BQP	52
3.4	SEPARAÇÃO ORACULAR ENTRE naCQP E NP	58
4	SEPARAÇÃO ORACULAR PARA naCQP	62
4.1	PROVA DO TEOREMA BBBV PARA naCQP	62
4.2	SEPARAÇÃO ORACULAR ALEATÓRIA ENTRE $UP \cap coUP$ E naCQP. . . .	64
4.3	SEPARAÇÃO ORACULAR GENÉRICA ENTRE $UP \cap coUP$ E naCQP	66
4.4	SEPARAÇÃO ORACULAR EXPONENCIAL ENTRE $UP \cap coUP$ E naCQP . .	67
5	PROBLEMAS RELACIONADOS A CLASSES DE CONHECIMENTO ZERO.	70
5.1	VERSÃO DE DECISÃO DE HSP E CLASSES DE COMPLEXIDADE DE CONHECIMENTO ZERO	71
5.2	UM PROBLEMA COMPLETO PARA $NISZK_1$	72
5.3	$BSDU \in BQP$	74
5.4	SOBRE A PERGUNTA $US \in coQMA$	76
6	CONSIDERAÇÕES FINAIS	79
	REFERÊNCIAS	82

1 INTRODUÇÃO

Os modelos clássico e quântico de computação definem classes diferentes de problemas computacionais que podem ser resolvidos em tempo polinomial. Para a computação clássica determinística, tal classe é a classe P, e para a computação quântica, é a classe BQP.

A classe P, definida na década de 60 em diversos artigos (Cobham, 1964; Edmonds, 1965; Rabin, 1960), é a classe dos problemas que podem ser resolvidos por um algoritmo determinístico polinomial. Em computação clássica, outra classe importante é a classe dos problemas que podem ser resolvidos por um algoritmo probabilístico polinomial com probabilidade de erro arbitrariamente pequena¹, denominada BPP (*Bounded-error Probabilistic Polynomial*), definida em Gill (1977), que claramente inclui P. Como essa probabilidade de erro é arbitrariamente pequena, pode-se considerar BPP como a classe dos problemas que são tratáveis para a computação clássica. Há problemas, como o problema da irreduzibilidade de polinômios sobre corpos finitos, que sabemos que estão em BPP, porém para os quais não se conhece algoritmo determinístico polinomial. Contudo, conjectura-se que $P = BPP$, embora ainda não tenha sido provado nem que $BPP \subseteq NP$ (Arora e Barak, 2009; Hemaspaandra e Ogihara, 2001).

A classe BQP (*Bounded-error Quantum Polynomial*), definida em Bernstein e Vazirani (1997), é a classe dos problemas que podem ser resolvidos por um algoritmo quântico polinomial com probabilidade de erro arbitrariamente pequena. Podemos considerar BQP como a classe dos problemas tratáveis para a computação quântica. Como todo algoritmo clássico probabilístico polinomial pode ser simulado por um algoritmo quântico polinomial com a mesma probabilidade de erro (Arora e Barak, 2009), BQP inclui BPP. Por outro lado, acredita-se que $BQP \neq BPP$. Existem algoritmos polinomiais quânticos para problemas que não parecem estar em BPP, como, por exemplo, os problemas da fatoração em números primos e do logaritmo discreto, os quais podem ser resolvidos pelo algoritmo de Shor.

A classe NP (*Nondeterministic Polynomial-Time*) é a classe dos problemas cujas instâncias positivas podem ser verificadas de modo eficiente. Isto é, se a instância é positiva, existe uma prova, ou certificado, de tamanho polinomial que pode ser verificada por um algoritmo determinístico polinomial. Se a instância é negativa, qualquer prova polinomial é rejeitada pelo verificador. Claramente, $P \subseteq NP$, contudo, ainda é um problema em aberto se $P \neq NP$ (Cook, 1971; Karp, 1972; Levin, 1973; Aaronson, 2016).

A classe PSPACE (*Polynomial-Space*) é a classe dos problemas que podem ser resolvidos por um algoritmo determinístico em espaço polinomial. A classe EXP (*Exponential Time*) é a classe dos problemas resolvidos por um algoritmo determinístico em tempo exponencial, i.e. $O(2^{n^k})$ para algum k inteiro.

Sabemos que $P \subseteq NP \subseteq PSPACE \subseteq EXP$, mas ainda são problemas em aberto se $P \neq PSPACE$, se $NP \neq PSPACE$ e se $PSPACE \neq EXP$ (Arora e Barak, 2009; Aaronson, 2016). Por outro lado, sabemos que $P \neq EXP$ (Stearns et al., 1965; Savage, 1997).

Uma subclasse importante de NP é a classe UP (*Unambiguous Polynomial-Time*), definida em Valiant (1976), em que se a instância é positiva existe exatamente uma prova, ou certificado, de tamanho polinomial que pode ser verificada por um algoritmo determinístico polinomial. Se a instância for negativa, qualquer prova é rejeitada pelo verificador. As principais

¹A probabilidade de erro, do algoritmo probabilístico, pode ser reduzida arbitrariamente repetindo a execução do algoritmo e tomando a resposta majoritária. Por limitante de Chernoff, podemos fazer a probabilidade de erro cair exponencialmente rápido com um número polinomial de rodadas. Com isso, alcançamos uma probabilidade de acerto tão próxima de 1 quanto quisermos.

classes estudadas neste trabalho (as que vimos e as que veremos ainda na introdução) e as relações conhecidas entre elas encontram-se na Figura 1.1.

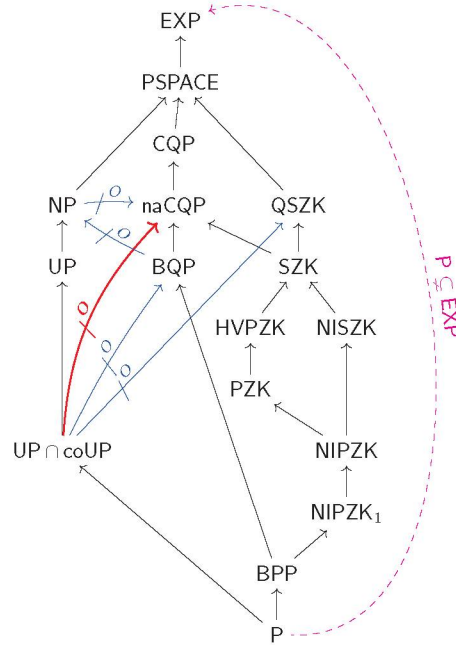


Figura 1.1: Relação entre as principais classes de complexidade estudadas neste trabalho. Uma seta $A \rightarrow B$ representa que $A \subseteq B$, sendo A e B classes. Uma seta $A \dashrightarrow B$ (tracejada em magenta) representa que $A \subsetneq B$. Uma seta $A \xrightarrow{O} B$ (em azul) representa que existe oráculo O tal que $A^O \not\subseteq B^O$. As setas anteriores são resultados da literatura. A seta $A \xrightarrow{O} B$ (em vermelho, mais grossa) representa o resultado do Teorema 1.2 e (para um oráculo aleatório) Teorema 1.1.

Sendo A um problema computacional (formalmente, uma linguagem), um algoritmo com consulta a um *oráculo* A é um algoritmo que pode, a qualquer momento, fazer consultas sobre instâncias do problema A e receber cada solução correspondente em um único passo. A classe dos problemas que podem ser resolvidos por um algoritmo que atende às restrições de uma classe de complexidade C , mas que pode fazer consultas a um oráculo A , é denotada por C^A .

Conforme Aaronson (2016), a maioria das técnicas existentes nas décadas de 60 e 70 para demonstrações de relações entre classes de complexidade, como $C \subseteq D$ ou $C \not\subseteq D$, eram tão gerais que, ao serem provadas, também provavam $C^A \subseteq D^A$ ou $C^A \not\subseteq D^A$ para todos possíveis oráculos A . Mais precisamente, em tais demonstrações, se cada algoritmo fosse trocado por um algoritmo que consulta oráculos, a argumentação ainda continua válida. Uma demonstração com essas características é dita ser *relativizante*, ou ser *possível em todos os mundos relativizados* ou ser *relativa para qualquer oráculo*. Um exemplo de demonstração relativizante é a de que $P \neq EXP$, i.e. a demonstração também implica que $P^A \neq EXP^A$ para todo oráculo A . Já Baker et al. (1975) demonstraram que existe um oráculo A para o qual $P^A = NP^A$ e outro oráculo B para o qual $P^B \neq NP^B$. Isso significa que uma demonstração para $P \neq NP$ (ou $P = NP$) deve usar técnicas mais sofisticadas que não relativizam. Um exemplo de uma técnica que não relativiza é a técnica da *aritmização*, utilizada na demonstração de que $IP = PSPACE$ (Shamir, 1992), sendo que existe um oráculo A para o qual $IP^A \neq PSPACE^A$. A classe IP será definida na Seção 2.5.

Quanto à relação entre BQP e NP , conjectura-se que NP não contém BQP . Relativo a um oráculo A , sabemos que $BQP^A \not\subseteq NP^A$ (Bernstein e Vazirani, 1997) e, ainda mais forte, existe um oráculo A para o qual $BQP^A \not\subseteq PH^A$, sendo PH uma superclasse de NP (Raz e Tal,

2022). Também se conjectura que BQP não contém a classe NP, i.e. por mais poderosa que a computação quântica possa parecer, ainda assim acredita-se que não seja o suficiente para resolver problemas NP-completos de modo eficiente. Relativo a um oráculo A sabemos que $NP^A \not\subseteq BQP^A$ (Bennett et al., 1997).

Bennett et al. (1997) ainda provam que para um oráculo A sorteado uniformemente², $UP^A \cap coUP^A \not\subseteq BQP^A$ com probabilidade 1. Bennet et al. provam no artigo para $NP^A \cap coNP^A$, contudo, observamos que a mesma prova vale para $UP^A \cap coUP^A$. Fortnow e Rogers (1999) provam que existe um oráculo genérico³ A tal que $P^A = BQP^A \neq UP^A \cap coUP^A$.

A classe BQP contém vários problemas, como o problema da fatoração e do logaritmo discreto, que são candidatos a NP-*intermediários*. Do Teorema de Ladner (1975), se $P \neq NP$, então existem problemas NP-intermediários, isto é, problemas que não são NP-completos, mas que também não estão em P. Existem várias classes de problemas candidatos a NP-intermediários, como a classe dada por $UP \cap coUP$ e a classe SZK (*Statistical Zero-Knowledge*). A classe SZK é amplamente estudada por suas aplicações na área de segurança da informação (para definição técnica de SZK, suas subclasses e problemas completos, citamos como referência o trabalho de Vadhan (1999); essas classes assim como as outras classes apresentadas, serão vistas com mais detalhes no Capítulo 2).

A classe NP também pode ser definida com um sistema de prova interativa, em que um provador com poderes ilimitados envia uma mensagem (o certificado) a um verificador, um algoritmo determinístico polinomial. Podemos generalizar de dois modos essa interação entre provador e verificador. Primeiro, usando como verificador um algoritmo probabilístico polinomial. Depois, permitindo que provador e verificador façam um número polinomial de rodadas de troca de mensagens. Assim, obtemos sistemas de prova interativa polinomial em que, para as instâncias positivas do problema, existe uma prova que o verificador aceita com probabilidade grande, e, para as instâncias negativas, qualquer prova tem probabilidade grande de ser rejeitada.

A classe MA (*Merlin–Arthur*) é a classe dos problemas que admitem sistemas de prova interativa com uma mensagem do provador para o verificador, um algoritmo probabilístico. A classe AM (*Arthur–Merlin*), é a classe dos problemas que admitem sistemas de prova interativa com *duas* mensagens: a primeira do verificador probabilístico polinomial para o provador; e a segunda do provador para o verificador.

A classe SZK é a classe dos problemas que admitem sistemas de prova interativa polinomial de conhecimento zero estatístico. Isto é, dados um provador com poderes de computação ilimitados e um verificador, este um algoritmo clássico probabilístico polinomial, o verificador é convencido da instância positiva de um problema sem aprender nada mais com essa interação, a menos de um erro estatístico⁴. A classe HVPZK é a classe dos problemas que admitem sistemas de prova interativa com verificadores honestos⁵ de conhecimento zero perfeito, isto é, não é permitido um erro estatístico. A classe PZK (*Perfect Zero-Knowledge*), subclasse de HVPZK, é a classe dos problemas que admitem sistemas de prova interativa polinomial de conhecimento zero perfeito. A generalização de SZK quando o verificador é um algoritmo quântico, ao invés de um algoritmo probabilístico clássico, é denotada por QSZK. Naturalmente,

²Veja a definição de oráculo genérico na Seção 4.3.

³Oráculo aleatório amostrado com distribuição uniforme do conjunto de todas as linguagens em Σ^* no sentido que, para toda palavra $x \in \Sigma^*$, a probabilidade de x estar no oráculo é $1/2$. Como, nesse caso, estamos falando de uma distribuição de probabilidade sobre um espaço contínuo (2^{Σ^*}), a probabilidade de ocorrer qualquer evento (elementar) do espaço é igual a 0 (Dasgupta, 2011; Mitzenmacher e Upfal, 2017).

⁴O significado preciso de “sem aprender nada com essa interação” é apresentado na Seção 2.5.

⁵Verificador honesto: algoritmo clássico probabilístico uniforme polinomial. Verificador desonesto: algoritmo clássico probabilístico não uniforme polinomial. Para mais detalhes, veja Seção 2.5.

$SZK \subseteq QSZK$. Sobre a relação entre $QSZK$ e UP , Menda e Watrous (2018) mostraram que existe um oráculo A para o qual $UP^A \cap coUP^A \not\subseteq QSZK^A$. Desse modo, $UP^A \cap coUP^A \not\subseteq SZK^A$.

Uma outra subclasse de SZK é a classe $NISZK$ (*Non-Interactive Statistical Zero-Knowledge*), a qual é a classe dos problemas que admitem sistemas de prova *não-interativa* de conhecimento zero estatístico, i.e. o provador e o verificador clássico compartilham uma string de bits aleatórios e ocorre a troca de apenas uma mensagem, do provador para o verificador. A classe $NIPZK$ (*Non-Interactive Perfect Zero-Knowledge*) é a classe dos problemas que admitem sistemas de prova *não-interativa* de conhecimento zero perfeito. A classe $NIPZK_1$ é a classe dos problemas que admitem sistemas prova *não-interativa* de conhecimento zero perfeito em que as instâncias positivas são aceitas pelo verificador com probabilidade igual a um.

Aaronson et al. (2016) introduziram as classes de complexidade CQP (*Collapse-free Quantum Polynomial time*) e $naCQP$ (*non-adaptive Collapse-free Quantum Polynomial time*). A primeira é a classe dos problemas que podem ser resolvidos por um algoritmo quântico de tempo polinomial que permite que as medições não causem o colapso dos estados. A última é a classe CQP com a restrição de que as operações quânticas não dependem dos resultados de medições sem colapso. Remetemos o leitor à Seção 3.2 para obter detalhes técnicos sobre a definição de $naCQP$. Observe que o objetivo de definir essas classes de complexidade nesta linha de pesquisa não é propor modelos alternativos de computação fisicamente realizáveis, mas investigar classes de problemas que parecem ser maiores que BQP , mas não grandes o suficiente para incluir problemas NP -completos, procurando, com isso, entender melhor os poderes e as limitações dos aspectos da computação quântica. A classe $naCQP$ também é chamada de $PDQP$ (*Product Dynamical Quantum Polynomial time*) na versão pré-impressa (Aaronson et al., 2014) e em três artigos subsequentes (Aaronson, 2018; Hiromasa et al., 2025; Aaronson et al., 2024). Aaronson et al. (2016) mostram que SZK (*Statistical Zero Knowledge*) está contida em $naCQP$ e que existe um oráculo A para o qual $NP^A \not\subseteq naCQP^A$.

Provamos que:

Teorema 1.1 (Hepp et al., 2025). *Relativo a um oráculo aleatório A , temos $UP^A \cap coUP^A \not\subseteq naCQP^A$ com probabilidade 1.*

Note que:

- nosso resultado amplia o resultado de Bennett et al. concernente $UP \cap coUP$ e BQP ;
- nosso resultado também amplia o resultado de Aaronson et al. de que existe um oráculo A para o qual $NP^A \not\subseteq naCQP^A$.

Teorema 1.2. *Existe um oráculo genérico A , tal que $P^A = BQP^A = SZK^A = naCQP^A \neq UP^A \cap coUP^A$.*

Note que:

- nosso resultado amplia o resultado de Aaronson et al., concernente $naCQP$ e NP , e também o resultado de Fortnow e Rogers, concernente P , BQP , e $UP \cap coUP$;
- concernente SZK e $UP \cap coUP$, nosso resultado implica que não apenas existe um oráculo A para o qual $UP^A \cap coUP^A \not\subseteq SZK^A$, algo que já decorre do resultado de Menda e Watrous, mas também que $UP^A \cap coUP^A \not\subseteq P^A = SZK^A$.

Beigel et al. (1998) também mostraram que existe um oráculo A para o qual $P^A = \oplus P^A$ e $RP^A = EXP^A$ (lembre-se que $P^A \neq EXP^A$ para todo oráculo A). Como citado por Tamon e

Yamakami (2001), como $UP^A \subseteq \oplus P^A$ e $RP^A \subseteq BQP^A$, temos $P^A = UP^A$ e $BQP^A = EXP^A$, ou seja, existe um oráculo A tal que

$$P^A = UP^A \cap coUP^A = UP^A \neq BQP^A = naCQP^A = EXP^A.$$

Curiosamente, Tamon e Yamakami também afirmaram que existe outro oráculo A para o qual $P^A = BQP^A$ e $UP^A = EXP^A$, apresentado um esboço de prova para tal. Provamos o seguinte resultado mais forte.

Teorema 1.3 (Hepp et al., 2025). *Existe um oráculo A para o qual $P^A = naCQP^A$ e $UP^A \cap coUP^A = EXP^A$.*

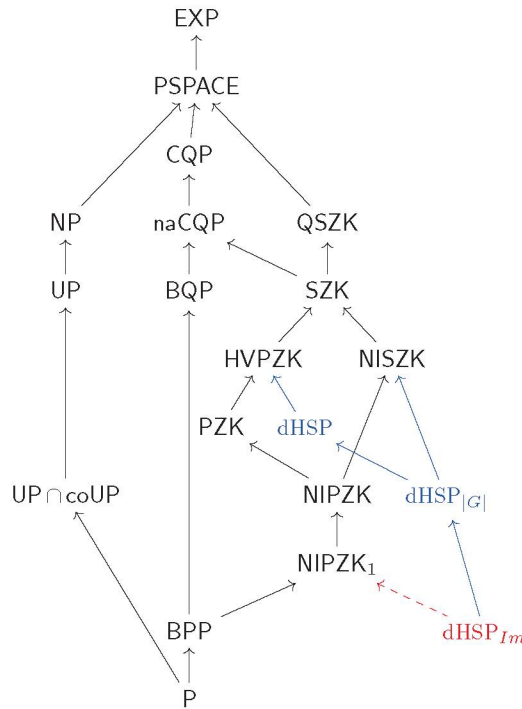


Figura 1.2: Relação entre as principais classes de complexidade estudadas neste trabalho e versões do problema dHSP. Uma seta $A \rightarrow B$ representa: que $A \subseteq B$ se A e B são classes; que $A \in B$, se A é um problema e B é uma classe; que A é uma restrição de B , se A e B são problemas. As setas azuis representam resultados da literatura referente às versões do problema dHSP. A seta vermelha, tracejada, representa o resultado do Teorema 1.4.

Um dos problemas candidatos para NP-intermediário é o problema do subgrupo oculto, denominado HSP (*Hidden Subgroup Problem*) que generaliza os problemas de fatoração e do logaritmo discreto. Estudamos a relação de problemas de decisão para o problema HSP (dHSP) com classes de provas interativas de conhecimento zero. Essas relações encontram-se na Figura 1.2. O problema dHSP consiste em decidir se o subgrupo oculto, de um grupo G , é o subgrupo trivial ou não. Esse problema está contido na classe HVPZK (Sdroievski et al., 2019). Caso se conheça o tamanho do grupo G , obtemos o problema dHSP $_{|G|}$ que está contido na classe NISZK (Sdroievski et al., 2019). Podemos também restringir o problema da seguinte forma: amostramos uniformemente ao acaso elementos de um grupo G e de um conjunto A , com a mesma dimensão do grupo G , tal que A contenha a imagem da função que esconde o subgrupo. Esse caso particular se aplica a casos importantes como quando $G = \mathbb{Z}_n$ com adição ou multiplicação modular. Provamos o seguinte.

Teorema 1.4 (Costa et al., 2023). $dHSP_{Im} \in NIPZK_1$.

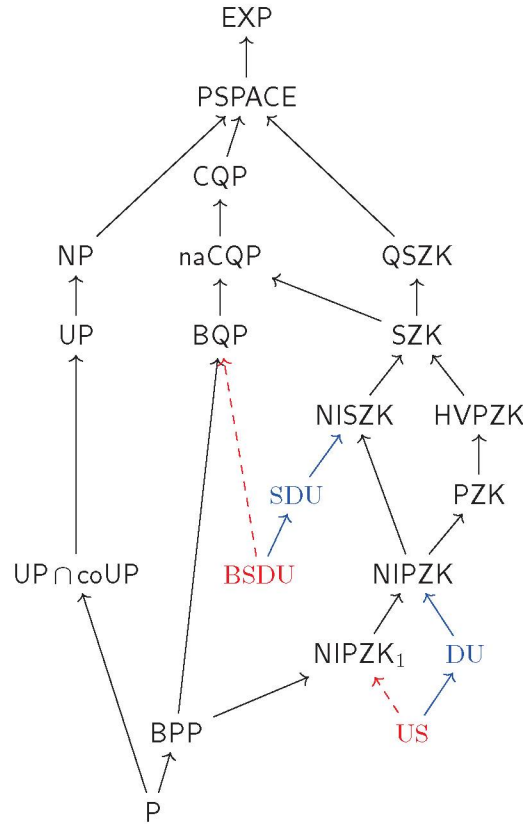


Figura 1.3: Relação entre as principais classes de complexidade estudadas neste trabalho e os problemas de distância estatística SDU, BSDU, DU e US. Uma seta $A \rightarrow B$ representa: que $A \subseteq B$ se A e B são classes; que $A \in B$, se A é um problema e B é uma classe; que A é uma restrição de B , se A e B são problemas. As setas azuis representam resultados da literatura referente aos problemas de distância estatística. As setas vermelhas tracejadas representam os resultados dos Teoremas 1.5 e 1.6.

Na Figura 1.3 mostramos a relação entre classes de conhecimento zero não interativo e variações do problema de distância estatística uniforme SDU (*Statistical Difference from Uniform*). Para SDU, precisamos comparar uma distribuição de probabilidade X com a distribuição uniforme U_n , ambas sobre todas as strings binárias com n bits, perguntando-nos o quão próximo a *distância estatística* $d_{VT}(X, U_n)$ está de 0 ou de 1. Esse problema é completo para NISZK (Goldreich et al., 1999). Malka (2008) mostrou uma variação do problema SDU, chamado de DU (de Distribuição Uniforme), que é completo para para a classe NIPZK. Uma restrição natural desse problema é o problema US (*Uniform-or-Small*), em que comparamos uma distribuição de probabilidade X com a distribuição uniforme U_n , ambas sobre todas as strings binárias com n bits, perguntando-nos a *distância estatística* $d_{VT}(X, U_n)$ é igual a 0 ou se está próxima de 1. Mostramos o seguinte.

Teorema 1.5 (Costa et al., 2023). *O problema US é completo para NIPZK₁.*

Apresentamos também uma restrição de SDU em que as probabilidades das $2^n/2$ primeiras strings (sob a ordenação lexicográfica usual) nunca são menores que as das $2^n/2$ últimas. Chamamos esse problema de Distância Estatística entre Distribuição Biparticionada e Distribuição Uniforme (BSDU). Provamos o seguinte.

Teorema 1.6 (Hepp et al., 2021). *BSDU está em BQP.*

O restante deste trabalho está organizado da seguinte maneira. O Capítulo 2 apresenta preliminares sobre álgebra linear, mecânica quântica, computação quântica e conceitos elementares de complexidade computacional. O Capítulo 3 apresenta uma revisão sobre as classes

de complexidade de conhecimento zero estatístico; sobre computação quântica sem colapso e sobre a separação oracular entre BQP e UP e entre naCQP e NP. O Capítulo 4 apresenta nossos resultados entre a separação oracular entre naCQP e UP. O Capítulo 5 apresenta nossos resultados obtidos sobre problemas relacionados a classes de conhecimento zero.

2 PRELIMINARES

Nesse capítulo apresentamos os preliminares necessários referentes a álgebra linear, mecânica quântica, computação quântica e complexidade. O conteúdo das Seções 2.1, 2.2 e 2.3 baseia-se principalmente nos livros de Kaye et al. (2007) e de Nielsen e Chuang (2011). Supomos que o leitor tenha familiaridade com conceitos da teoria da probabilidade, caso contrário, sugerimos como referência o livro de Mitzenmacher e Upfal (2017).

2.1 ÁLGEBRA LINEAR

Os objetos básicos da álgebra linear são os *espaços vetoriais*. Na computação quântica os espaços vetoriais nos quais estamos interessados são os espaços vetoriais complexos com dimensão finita com produto interno, que nomeamos como *espaços de Hilbert*. Neste texto usamos as letras $\mathcal{H}, \mathcal{V}, \mathcal{W}$ para denotar espaços de Hilbert. Para denotar os vetores de um espaço de Hilbert usamos a *notação de kets*, ou seja, um vetor v é denotado como $|v\rangle$.

Podemos descrever os vetores por meio de matrizes. Nesse caso, um vetor $|v\rangle = (z_1, z_2, \dots, z_N)$, em que $z_1, z_2, \dots, z_N$ são números complexos, é expresso como

$$|v\rangle = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix}.$$

Para denotar o vetor transposto conjugado de $|v\rangle$ usamos a notação *bra*, ou seja,

$$\langle v| = \begin{bmatrix} z_1^* \\ z_2^* \\ \vdots \\ z_N^* \end{bmatrix}^T = \begin{bmatrix} z_1^* & z_2^* & \cdots & z_N^* \end{bmatrix},$$

onde z_i^* são os complexos conjugados de z_i .

O *produto interno* de um vetor $|v\rangle = (\alpha_1, \alpha_2, \dots, \alpha_N)$ por um vetor $|w\rangle = (\beta_1, \beta_2, \dots, \beta_N)$ é definido e denotado por

$$\langle v|w\rangle = \alpha_1^* \beta_1 + \cdots + \alpha_N^* \beta_N.$$

Observe que $\langle v|w\rangle = \langle w|v\rangle^*$. Se o produto interno de dois vetores é igual a 0, esses dois vetores são *ortogonais*. A *norma de um vetor* $|v\rangle \in \mathcal{V}$ é

$$\| |v\rangle \| = \sqrt{\langle v|v\rangle}.$$

Um vetor é *unitário* se tiver norma igual a um. Vetores unitários ortogonais são chamados de *ortonormais*.

Dado um espaço de Hilbert $\mathcal{H}$ com dimensão N , um conjunto de N vetores $B = \{|b_m\rangle\} \subseteq \mathcal{H}$ é uma *base ortonormal* de $\mathcal{H}$ se

- $\langle b_n | b_m \rangle = \delta_{n,m} \quad \forall b_m, b_n \in B$ sendo $\delta_{n,m} = 1$ se $n = m$ e $\delta_{n,m} = 0$ se $n \neq m$;
- e todo $|v\rangle \in \mathcal{H}$ pode ser escrito como

$$|v\rangle = \sum_{b_n \in B} \alpha_n |b_n\rangle \quad \text{para } \{\alpha_n\} \subset \mathbb{C}.$$

Os valores de α_n satisfazem $\alpha_n = \langle b_n | \psi \rangle$ e são chamados de coeficientes de $|\psi\rangle$ com respeito à base $\{|b_n\rangle\}$.

Podemos, portanto, expressar um vetor $|v\rangle = \alpha_1 |v_1\rangle + \alpha_2 |v_2\rangle + \cdots + \alpha_N |v_N\rangle$, em que $|v_i\rangle$ são vetores unitários linearmente independentes e α_i são números complexos, como

$$|v\rangle = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{bmatrix}.$$

O vetor transposto conjugado de $|v\rangle$ pode ser expresso como

$$\langle v| = \begin{bmatrix} \alpha_1^* & \alpha_2^* & \cdots & \alpha_N^* \end{bmatrix},$$

onde os coeficientes α_i^* são os complexos conjugados de α_i .

É conveniente usarmos uma base de 2^n vetores unitários, chamada de *base computacional*, em que os vetores são representados por kets com strings binárias de tamanho n :

$$\underbrace{|00 \dots 00\rangle}_n, |00 \dots 01\rangle, \dots, |11 \dots 10\rangle, |11 \dots 11\rangle.$$

Podemos expressar um vetor $|v\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle + \cdots + \alpha_{2^n-1} |2^n - 1\rangle$, em que $|i\rangle$ compõe a base computacional e α_i são números complexos, como

$$|v\rangle = \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{2^n-1} \end{bmatrix} = \alpha_0 \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \alpha_1 \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} + \cdots + \alpha_{2^n-1} \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}.$$

A distância euclidiana entre dois vetores $|v\rangle = \sum_x \alpha_x |x\rangle$ e $|w\rangle = \sum_x \beta_x |x\rangle$ é

$$\| |v\rangle - |w\rangle \| = \| |v\rangle - |w\rangle \|_2 = \sqrt{\sum_x |\alpha_x - \beta_x|^2}.$$

Um *operador linear* entre $\mathcal{V}$ e $\mathcal{W}$ é qualquer função $A : \mathcal{V} \rightarrow \mathcal{W}$ que seja linear, ou seja,

$$A \left(\sum_i \alpha_i |v_i\rangle \right) = \sum_i \alpha_i A(|v_i\rangle).$$

Usualmente escrevemos $A |v\rangle$ para denotar $A(|v\rangle)$. Dado um espaço de Hilbert $\mathcal{H}$, denotamos por $\mathbf{L}(\mathcal{H})$ o conjunto de operadores lineares $A : \mathcal{H} \rightarrow \mathcal{H}$.

A norma de um operador $A \in \mathbf{L}(\mathcal{H})$, para todo vetor $|v\rangle \in \mathcal{H}$, é

$$\|A\| = \sup_{|v\rangle \neq 0} \frac{\|A|v\rangle\|}{\||v\rangle\|}.$$

Sejam $|v\rangle \in \mathcal{V}$ e $|w\rangle \in \mathcal{W}$. O *produto diádico* denotado por $|w\rangle\langle v|$ é um operador linear de $\mathcal{V}$ para $\mathcal{W}$ cuja ação é definida por:

$$(|w\rangle\langle v|)(|v'\rangle) = |w\rangle\langle v|v'\rangle.$$

Observe que: $|w\rangle\langle v|v'\rangle = \langle v|v'\rangle |w\rangle$. Em sua forma matricial, o *produto diádico* de dois vetores $|w\rangle = \alpha_1 |1\rangle + \dots + \alpha_N |N\rangle$ e $|v\rangle = \beta_1 |1\rangle + \dots + \beta_N |N\rangle$ para toda base ortonormal $\{|i\rangle\}$ de $\mathcal{H}$ é

$$|w\rangle\langle v| = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_N \end{bmatrix} \begin{bmatrix} \beta_1^* & \beta_2^* & \beta_3^* & \cdots & \beta_N^* \end{bmatrix}.$$

Seja $\{|i\rangle\}$ uma base ortonormal para $\mathcal{V}$, como um vetor $v \in \mathcal{V}$ pode ser escrito como $|v\rangle = \sum_i \alpha_i |i\rangle$, para algum conjunto de números complexos $\{\alpha_i\}$ e como $\langle i|v\rangle = \alpha_i$, temos:

$$\left(\sum_i |i\rangle\langle i| \right) |v\rangle = \sum_i |i\rangle\langle i|v\rangle = \sum_i \alpha_i |i\rangle = |v\rangle. \quad (2.1)$$

Como (2.1) vale para todo v , obtemos a equação conhecida como *relação de completude*:

$$\sum_i |i\rangle\langle i| = \mathbb{1}.$$

Dado um operador linear A em $\mathcal{H}$, o *operador adjunto* ou *Hermitiano conjugado* de A , denotado por $A^\dagger$, é definido como

$$A^\dagger = (A^*)^T.$$

Observe que

$$(\langle v|A|w\rangle)^* = \langle w|A^\dagger|v\rangle \quad \forall |v\rangle, |w\rangle \in \mathcal{H}.$$

Por convenção,

$$|v\rangle^\dagger = \langle v|.$$

A operação adjunta tem as seguintes propriedades:

- $(AB)^\dagger = B^\dagger A^\dagger$;
- $(|w\rangle\langle v|)^\dagger = |v\rangle\langle w|$;
- $(A^\dagger)^\dagger = A$;
- $(\sum_i \alpha_i A_i)^\dagger = \sum_i \alpha_i^* A_i^\dagger$.

Na mecânica quântica é comum o uso dos seguintes operadores lineares.

- Um operador N é *normal* se $NN^\dagger = N^\dagger N$. Denotamos o conjunto de todos operadores normais em $\mathcal{H}$ como $\mathbf{Norm}(\mathcal{H})$.

- Um operador H é *hermitiano* se $H = H^\dagger$. Denotamos o conjunto de todos operadores hermitianos em $\mathcal{H}$ como **Herm**($\mathcal{H}$).
- Um operador P é *positivo* ou *positivo semidefinido* se é hermitiano e para qualquer vetor $|v\rangle$, temos $\langle v|P|v\rangle \geq 0$, e $\langle v|P|v\rangle \in \mathbb{R}$. Denotamos o conjunto de todos operadores positivos em $\mathcal{H}$ como **Pos**($\mathcal{H}$).
- Um operador Π é *projedor ortogonal* se é hermitiano, e satisfaz $\Pi^2 = \Pi$. Denotamos o conjunto de todos operadores projetores ortogonais em $\mathcal{H}$ como **Proj**($\mathcal{H}$).
- Um operador $U \in \mathbf{L}(\mathcal{H})$ é *unitário* se $U^\dagger U = \mathbb{1}$. Denotamos o conjunto de todos operadores unitários em $\mathcal{H}$ como **U**($\mathcal{H}$). A definição também implica que:
 1. $U^\dagger = U^{-1}$, onde U^{-1} é a inversa de U ;
 2. U preserva o produto interno, $\langle Uv|Uw\rangle = \langle v|w\rangle$ para todo $|v\rangle, |w\rangle \in \mathcal{H}$;
 3. U preserva a norma: $\|U|v\rangle\| = \||v\rangle\|$ para todo $|v\rangle \in \mathcal{H}$;
 4. se $\||v\rangle\| = 1$ então $\|U|v\rangle\| = 1$ para todo $|v\rangle \in \mathcal{H}$.

A relação entre os operadores lineares é ilustrada na Figura 2.1.

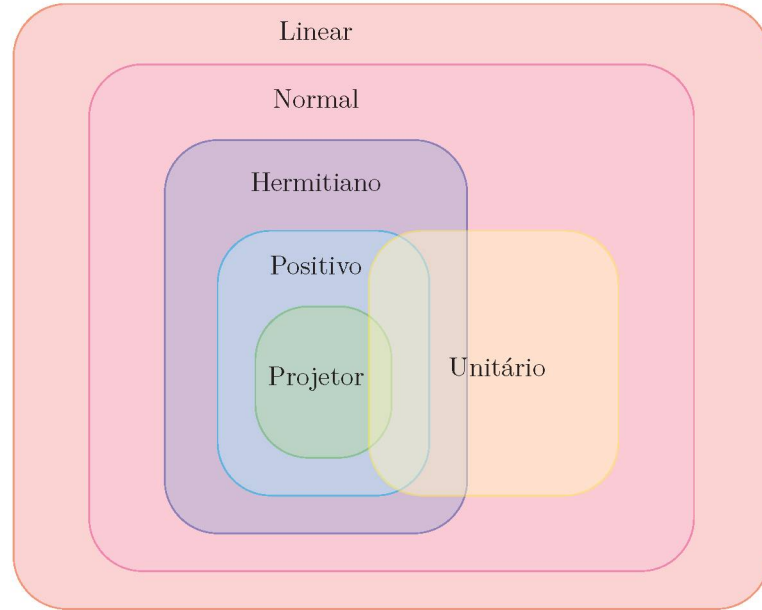


Figura 2.1: Tipos de operadores lineares.

Um vetor $|\psi\rangle$ é nomeado de *autovetor* de um operador T se, para alguma constante λ , temos:

$$T|\psi\rangle = \lambda|\psi\rangle.$$

A constante λ é nomeada como *autovalor* de T correspondente ao autovetor $|\psi\rangle$.

Teorema 2.1 (O Teorema Espectral). *Qualquer operador **normal** M sobre um espaço de Hilbert $\mathcal{H}$ pode ser escrito como:*

$$M = \sum_i \lambda_i |i\rangle \langle i|,$$

onde λ_i são os autovalores de M , $\{|i\rangle\}$ é uma base ortonormal para $\mathcal{H}$, e cada $|i\rangle$ é um autovetor de M com autovalor λ_i .

O produto tensorial entre dois vetores $|v\rangle \in \mathcal{V}$ e $|w\rangle \in \mathcal{W}$ é um vetor denotado por $|v\rangle \otimes |w\rangle$ em $\mathcal{V} \otimes \mathcal{W}$. O produto tensorial é caracterizado pelos seguintes axiomas.

1. Para quaisquer $c \in \mathbb{C}$, $|v\rangle \in \mathcal{V}$ e $|w\rangle \in \mathcal{W}$,

$$c(|v\rangle \otimes |w\rangle) = (c|v\rangle) \otimes |w\rangle = |v\rangle \otimes (c|w\rangle) ;$$

2. Para quaisquer $|v_1\rangle, |v_2\rangle \in \mathcal{V}$ e $|w\rangle \in \mathcal{W}$,

$$(|v_1\rangle + |v_2\rangle) \otimes |w\rangle = |v_1\rangle \otimes |w\rangle + |v_2\rangle \otimes |w\rangle ;$$

3. Para quaisquer $|v\rangle \in \mathcal{V}$ e $|w_1\rangle, |w_2\rangle \in \mathcal{W}$,

$$|v\rangle \otimes (|w_1\rangle + |w_2\rangle) = |v\rangle \otimes |w_1\rangle + |v\rangle \otimes |w_2\rangle .$$

Sejam $\mathcal{V}$ e $\mathcal{W}$ espaços de Hilbert com dimensões n e m respectivamente. O *produto tensorial* de $\mathcal{V}$ por $\mathcal{W}$, denotado por $\mathcal{V} \otimes \mathcal{W}$, é um espaço de Hilbert com dimensão $n \times m$. Sejam $\{|b_i\rangle\}_{i \in \{1, \dots, n\}}$ uma base ortonormal para $\mathcal{V}$ e $\{|c_j\rangle\}_{j \in \{1, \dots, m\}}$ uma base ortonormal para $\mathcal{W}$. Então,

$$\{|b_i\rangle \otimes |c_j\rangle\}_{i \in \{1, \dots, n\}, j \in \{1, \dots, m\}}$$

é uma base ortonormal para $\mathcal{V} \otimes \mathcal{W}$.

As seguintes notações são equivalentes:

$$\begin{aligned} |i\rangle \otimes |j\rangle &= |i, j\rangle = |ij\rangle ; \\ |i, j\rangle_{AB} &= |i\rangle_A \otimes |j\rangle_B = |i\rangle_A |j\rangle_B . \end{aligned}$$

Duas identidades úteis são:

$$\begin{aligned} |ij\rangle \langle kl| &= |i\rangle \langle k| \otimes |j\rangle \langle l| ; \\ \langle ij|kl\rangle &= \langle i|k\rangle \langle j|l\rangle . \end{aligned}$$

Sejam A e B operadores lineares em $\mathcal{V}$ e $\mathcal{W}$, respectivamente. Então $A \otimes B$ é o operador linear em $\mathcal{V} \otimes \mathcal{W}$ definido por

$$(A \otimes B)(|v\rangle \otimes |w\rangle) = A|v\rangle \otimes B|w\rangle .$$

O produto tensorial também pode ser representado matricialmente. Dadas as matrizes A com dimensão $m \times n$ e B com dimensão $p \times q$, o *produto de Kronecker*, denotado por $A \otimes B$, é a matriz $mp \times nq$ dada por

$$A \otimes B = \begin{bmatrix} A_{11}[B] & A_{12}[B] & \cdots & A_{1n}[B] \\ A_{21}[B] & A_{22}[B] & \cdots & A_{2n}[B] \\ \vdots & \vdots & \vdots & \vdots \\ A_{m1}[B] & A_{m2}[B] & \cdots & A_{mn}[B] \end{bmatrix}$$

$$A \otimes B = \begin{bmatrix} A_{11}B_{11} & \cdots & A_{11}B_{1q} & \cdots & \cdots & A_{1n}B_{11} & \cdots & A_{1n}B_{1q} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{11}B_{p1} & \cdots & A_{11}B_{pq} & \cdots & \cdots & A_{1n}B_{p1} & \cdots & A_{1n}B_{pq} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{m1}B_{11} & \cdots & A_{m1}B_{1q} & \cdots & \cdots & A_{mn}B_{11} & \cdots & A_{mn}B_{1q} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{m1}B_{p1} & \cdots & A_{m1}B_{pq} & \cdots & \cdots & A_{mn}B_{p1} & \cdots & A_{mn}B_{pq} \end{bmatrix} .$$

O *traço* de um operador A sobre um espaço de Hilbert $\mathcal{H}$ é

$$\text{tr}(A) = \sum_{b_n} \langle b_n | A | b_n \rangle ,$$

onde $\{b_n\}$ é qualquer base ortonormal para $\mathcal{H}$. Em sua representação matricial o traço é a soma dos elementos da diagonal principal.

Dados os operadores lineares $A, B \in \mathbf{L}(\mathcal{H})$ e $U \in \mathbf{U}(\mathcal{H})$, a função traço tem as seguintes propriedades:

- (Propriedade cíclica). $\text{tr}(AB) = \text{tr}(BA)$;
- (Linearidade) $\text{tr}(A + B) = \text{tr}(A) + \text{tr}(B)$;
- $\text{tr}(zA) = z \text{tr}(A)$ com $z \in \mathbb{C}$;
- $\text{tr}(UAU^\dagger) = \text{tr}(U^\dagger UA) = \text{tr}(A)$ (o traço não depende da base) ;
- $\text{tr}(A \otimes B) = \text{tr}(A) \text{tr}(B)$.

Dados dois vetores $|\psi\rangle \in \mathcal{H}$ e $|\phi\rangle \in \mathcal{H}$, o traço do produto diádico $|\psi\rangle \langle \phi|$ é:

$$\text{tr}(|\psi\rangle \langle \phi|) = \sum_i \langle i | \psi \rangle \langle \phi | i \rangle = \sum_i \langle \phi | i \rangle \langle i | \psi \rangle = \langle \phi | \psi \rangle .$$

2.2 MECÂNICA QUÂNTICA

A mecânica quântica pode ser vista como um arcabouço matemático e conceitual usado para descrever um sistema físico. Podemos nomear um conjunto de postulados como base da mecânica quântica, mas observamos que não existe um consenso sobre o conteúdo e forma dos postulados.

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

Um estado $|\psi\rangle$ em um espaço com N dimensões, $\mathbb{C}^N$, pode ser descrito pela combinação linear de N estados linearmente independentes

$$|\psi\rangle = \alpha_1 |1\rangle + \alpha_2 |2\rangle + \alpha_3 |3\rangle + \cdots + \alpha_N |N\rangle ,$$

onde $\alpha_1, \cdots, \alpha_N$ são números complexos. Como $|\psi\rangle$ é um vetor unitário, temos que $\sum_i |\alpha_i|^2 = 1$. Podemos também dizer que o estado quântico $|\psi\rangle$ é a *superposição* dos N estados, e os números $\alpha_1, \cdots, \alpha_N$ também são chamados de *amplitudes* dos estados.

Postulado 2. *A evolução de um sistema quântico isolado é descrita pela equação de Schrödinger,*

$$i\hbar \frac{d|\psi\rangle}{dt} = H |\psi\rangle ,$$

onde $\hbar$ é a constante de Planck e H é um operador hermitiano conhecido por Hamiltoniano.

Quando o sistema quântico é isolado, o Hamiltoniano é constante com relação ao tempo. Nesse caso a solução da equação de Schrödinger pode ser expressa por

$$|\psi(t_2)\rangle = \exp \left[\frac{-iH(t_2 - t_1)}{\hbar} \right] |\psi(t_1)\rangle = U(t_1, t_2) |\psi(t_1)\rangle ,$$

onde definimos

$$U(t_1, t_2) = \exp \left[\frac{-iH(t_2 - t_1)}{\hbar} \right] .$$

Como U é unitário e qualquer operador unitário pode ser expresso por $U = \exp(iK)$ para algum operador hermitiano K , podemos enunciar o segundo postulado como:

Postulado 2'. *A evolução de um sistema quântico isolado é descrita por uma **transformação unitária**. O estado $|\psi\rangle$ do sistema no tempo t_i está relacionado com o estado $|\psi'\rangle$ do sistema no tempo t_2 por meio de um operador unitário U que depende apenas dos tempos t_1 e t_2 ,*

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

Postulado 3. *O espaço de estado de um sistema físico composto é o produto tensorial dos espaços de estados dos subsistemas. Se os sistemas forem numerados de 1 a n , o estado composto do sistema total é $|\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_n\rangle$.*

Um estado puro $|\psi\rangle \in \mathcal{V} \otimes \mathcal{W}$ é um *estado produto* se pode ser escrito como um produto tensorial de dois estados,

$$|\psi\rangle = |\psi_V\rangle \otimes |\psi_W\rangle ,$$

sendo $|\psi_V\rangle \in \mathcal{V}$ e $|\psi_W\rangle \in \mathcal{W}$. Caso não seja possível, esse estado é *emaranhado*. Quatro exemplos de estados emaranhados são os *estados de Bell*:

$$\begin{aligned} & \frac{|00\rangle + |11\rangle}{\sqrt{2}}, \\ & \frac{|00\rangle - |11\rangle}{\sqrt{2}}, \\ & \frac{|10\rangle + |01\rangle}{\sqrt{2}}, \\ & \frac{|01\rangle - |10\rangle}{\sqrt{2}}. \end{aligned}$$

Postulado 4. As *medições quânticas gerais* são descritas por uma coleção $\{M_i\}$ de operadores, chamados de operadores de medição, os quais satisfazem a equação de completude

$$\sum_i M_i^\dagger M_i = \mathbb{1}.$$

Se o estado do sistema for $|\psi\rangle$ imediatamente antes da medição, então a probabilidade de ocorrer o resultado i é dado por

$$p(i) = \langle\psi| M_i^\dagger M_i |\psi\rangle,$$

e o estado do sistema após a medição é

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

A equação de completude expressa o fato de que a soma das probabilidades é igual a 1,

$$1 = \sum_i p(i) = \sum_i \langle\psi| M_i^\dagger M_i |\psi\rangle.$$

Se os operadores de medição M_i são operadores projetivos Π_i , temos que a coleção $\{M_i\}$ pode ser expressa em termos da matriz

$$M = \sum_i m_i \Pi_i = \sum_i m_i |\varphi_i\rangle \langle\varphi_i|.$$

A probabilidade de obtermos o evento i é dada por

$$p(i) = \langle\psi| \Pi_i |\psi\rangle = \langle\psi| \varphi_i\rangle \langle\varphi_i| \psi\rangle;$$

então, sendo $\alpha_i = \langle\varphi_i| \psi\rangle$,

$$p(i) = \langle\psi| \varphi_i\rangle \langle\varphi_i| \psi\rangle = \alpha_i^* \alpha_i = |\alpha_i|^2.$$

O estado do sistema após a medição é:

$$\frac{\Pi_i |\psi\rangle}{\sqrt{\langle\psi| \Pi_i |\psi\rangle}} = \frac{|\varphi_i\rangle \langle\varphi_i| \psi\rangle}{\sqrt{|\alpha_i|^2}} = \frac{\alpha_i |\varphi_i\rangle}{\alpha_i} = |\varphi_i\rangle.$$

Uma outra maneira de se enunciar o Postulado 4, portanto, é a seguinte:

Postulado 4'. *Dados uma base ortonormal $B = \{|\psi_i\rangle\}$ de um espaço de estado $\mathcal{H}_A$ e um estado $|\psi\rangle = \sum_i \alpha_i |\psi_i\rangle$, ao fazermos uma **medição de Von Neumann** de $|\psi\rangle$ com respeito à base B , obtemos o estado $|\psi_i\rangle$ com probabilidade $|\alpha_i|^2$ e perdemos todas as demais informações do estado original. Em outras palavras, ao medirmos $|\psi\rangle$, o estado **colapsa** em $|\psi_i\rangle$ com probabilidade $|\alpha_i|^2$.*

Considere um estado $|\psi\rangle = \sum_i \alpha_i |\varphi_i\rangle |\gamma_i\rangle$ de um espaço de estado $\mathcal{H}_A \otimes \mathcal{H}_B$ em que $|\varphi_i\rangle$ são ortonormais e $|\gamma_i\rangle$ são vetores unitários, a medição no sistema A colapsa o estado $|\psi\rangle$ em $|\varphi_i\rangle |\gamma_i\rangle$ com probabilidade $|\alpha_i|^2$.

Há casos em que o sistema é descrito por um conjunto de estados puros $\{|\psi_i\rangle\}$, associado cada um a uma probabilidade p_i , de modo que $\sum_i p_i = 1$:

$$\{(|\psi_1\rangle, p_1), (|\psi_2\rangle, p_2), \dots, (|\psi_k\rangle, p_k)\}.$$

Esse conjunto é chamado de *estado misto*. O *operador densidade*, também conhecido como matriz densidade, associado a esse estado misto é definido como

$$\rho = \sum_i p_i |\psi_i\rangle \langle \psi_i|.$$

Em particular, para um estado puro $|\psi\rangle$,

$$\rho = |\psi\rangle \langle \psi|.$$

Denotamos $\mathbf{D}(\mathcal{H})$ como o conjunto de operadores densidade no espaço de Hilbert $\mathcal{H}$.

Por exemplo, o operador densidade para o estado puro $|\psi\rangle = a|0\rangle + b|1\rangle$ é

$$\begin{aligned} \rho &= \sum_i p_i |\psi_i\rangle \langle \psi_i| = 1 \cdot |\psi\rangle \langle \psi| = (a|0\rangle + b|1\rangle)(a^* \langle 0| + b^* \langle 1|) \\ \rho &= aa^* |0\rangle \langle 0| + ab^* |0\rangle \langle 1| + ba^* |1\rangle \langle 0| + bb^* |1\rangle \langle 1|. \end{aligned}$$

A representação matricial de ρ é

$$\rho = \begin{bmatrix} aa^* & ab^* \\ ba^* & bb^* \end{bmatrix}.$$

O operador densidade para o estado misto em que há a probabilidade p de se ter o estado $|0\rangle$ e $1 - p$ de se ter o estado $|1\rangle$ é

$$\begin{aligned} \rho &= p |0\rangle \langle 0| + (1 - p) |1\rangle \langle 1| \\ \rho &= \begin{bmatrix} p & 0 \\ 0 & 1 - p \end{bmatrix}. \end{aligned}$$

Teorema 2.2. *Um operador ρ é o operador densidade associado a um conjunto $\{|\psi_i\rangle, p_i\}$ se e somente se satisfaz:*

1. $\text{tr}(\rho) = 1$;
2. $\rho \in \mathbf{Pos}(\mathcal{H})$.

Observe que, se o estado ao qual um operador densidade ρ está associado é puro, então

$$\text{tr}(\rho^2) = 1 ,$$

caso contrário

$$\text{tr}(\rho^2) < 1 .$$

Podemos agora reescrever os postulados para os operadores de densidade.

Postulado 1. *A todo sistema físico isolado está associado um espaço de Hilbert, conhecido por **espaço de estado** do sistema. O sistema é descrito por seu **operador densidade**, que é um operador ρ positivo com traço igual a um agindo sobre o espaço de estado do sistema. Se um sistema quântico está no estado ρ_i com probabilidade p_i , então o operador densidade para o sistema é $\sum_i p_i \rho_i$.*

Postulado 2. *A evolução de um sistema **isolado** é descrita por uma **transformação unitária**. O estado ρ do sistema no tempo t_1 está relacionado com o estado ρ' do sistema no tempo t_2 por meio de um operador unitário U que depende apenas dos tempos t_1 e t_2 :*

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

Postulado 3. *O espaço de estado de um sistema físico composto é o produto tensorial dos espaços de estados dos subsistemas. Se os sistemas forem numerados de 1 a n , o estado composto do sistema total é $\rho_1 \otimes \rho_2 \otimes \cdots \otimes \rho_n$.*

Um estado $\rho \in \mathbf{D}(\mathcal{V} \otimes \mathcal{W})$ é *separável* se pode ser escrito como uma combinação linear de produtos estados,

$$\rho = \sum_i p_i \rho_V^i \otimes \rho_W^i ,$$

sendo $0 \leq p_i \leq 1$, $\sum_i p_i = 1$, $\rho_V^i \in \mathbf{D}(\mathcal{V})$ e $\rho_W^i \in \mathbf{D}(\mathcal{W})$. Caso não seja possível, ele é *emaranhado*.

Postulado 4. *As medições quânticas são descritas por uma coleção $\{M_i\}$ de **operadores de medição**. Esses são os operadores lineares agindo sobre o espaço de estado do sistema. O índice i se refere aos possíveis resultados da medição. Se o estado do sistema for ρ imediatamente antes da medição, então a probabilidade de ocorrer o resultado i é dada por*

$$p(i) = \text{tr}(M_i^\dagger M_i \rho) ,$$

e o estado do sistema após a medição é

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

Os operadores de medição satisfazem a equação de completude

$$\sum_i M_i^\dagger M_i = \mathbb{1} .$$

A seguir, uma maneira alternativa de enunciar o Postulado 4 é enunciada a seguir.

Postulado 4'. Dada uma coleção $\{(|\psi_i\rangle, p_i)\}_{i=1}^N$ onde $\psi_i \in \mathcal{H}$ e p_i é a probabilidade de se obter $|\psi_i\rangle$, temos:

$$\begin{aligned} \Pr[\text{obter } |\phi\rangle] &= \sum_{i=1}^N p_i |\langle\psi_i|\phi\rangle|^2 \\ &= \sum_{i=1}^N p_i \langle\phi|\psi_i\rangle \langle\psi_i|\phi\rangle \\ &= \langle\phi| \left(\sum_{i=1}^N p_i |\psi_i\rangle \langle\psi_i| \right) |\phi\rangle \\ &= \langle\phi| \rho |\phi\rangle . \end{aligned}$$

Dados os sistemas quânticos $A = |a_1\rangle \langle a_2|$ e $B = |b_1\rangle \langle b_2|$, cujos estados são descritos por um operador densidade ρ^{AB} , o *operador de densidade reduzido* para o sistema A é definido como

$$\rho^A = \text{tr}_B(\rho^{AB}) ,$$

onde tr_B é o *traço parcial* sobre o sistema B , definido como

$$\text{tr}_B (|a_1\rangle \langle a_2| \otimes |b_1\rangle \langle b_2|) = |a_1\rangle \langle a_2| \text{tr}(|b_1\rangle \langle b_2|) .$$

Como

$$\text{tr}(|b_1\rangle \langle b_2|) = \text{tr}(\langle b_2|b_1\rangle) = \langle b_2|b_1\rangle ,$$

obtemos

$$\text{tr}_B (|a_1\rangle \langle a_2| \otimes |b_1\rangle \langle b_2|) = |a_1\rangle \langle a_2| \langle b_2|b_1\rangle .$$

Uma definição alternativa para o traço parcial é: dados dois espaços de Hilbert A e B e uma base ortonormal $\{|b_1\rangle, \dots, |b_n\rangle\}$ para B , definimos um mapeamento $\text{tr}_B : \mathbf{L}(A \otimes B) \rightarrow \mathbf{L}(A)$ tal que

$$\text{tr}_B \rho^{AB} = \sum_{j=1}^n (I \otimes \langle b_j|) \rho^{AB} (I \otimes |b_j\rangle) .$$

O traço parcial sobre B intuitivamente significa que descartamos ou ignoramos a parte B do sistema. Ao dizermos que uma parte do sistema quântico foi *traced out*, significa que foi executado o traço parcial ignorando essa parte do sistema.

Por exemplo, o operador densidade do estado

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

é

$$\rho = \frac{1}{2}(|00\rangle \langle 00| + |00\rangle \langle 11| + |11\rangle \langle 00| + |11\rangle \langle 11|) .$$

O operador de densidade reduzido do primeiro qubit sendo feito o traço parcial do segundo qubit é

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

Um estado puro ρ^{AB} é um estado produto, se e somente se os operadores de densidade reduzidos ρ^A e ρ^B são estados puros. Caso contrário, o estado ρ^{AB} é emaranhado.

O *rank* de uma matriz é o número de linhas ou colunas independentes. No caso de um operador de densidade, o rank corresponde ao menor número de estados puros necessários para formar um estado misto

$$\rho = \sum_i^{\text{rank}} p_i |\psi_i\rangle\langle\psi_i| .$$

A *purificação de um estado misto* $\rho \in \mathbf{D}(\mathcal{V})$ é qualquer estado puro $|\psi\rangle \in \mathcal{V} \otimes \mathcal{W}$ tal que, para algum $\mathcal{W}$, temos $\text{tr}_{\mathcal{W}} |\psi\rangle\langle\psi| = \rho$. Essa purificação sempre existe, desde que $\dim(\mathcal{W}) \geq \text{rank}(\rho)$.

Dados dois operadores de densidade ρ e σ , definimos a *distância de traço* como

$$T(\rho, \sigma) = \|\rho - \sigma\|_{\text{tr}} = \frac{1}{2} \text{tr} \left[\sqrt{(\rho - \sigma)^\dagger (\rho - \sigma)} \right] = \frac{1}{2} \sum_i |\lambda_i| ,$$

onde λ_i são os autovalores da matriz $\rho - \sigma$.

A distância de traço pode ser interpretada como sendo a maior distância que pode ser medida entre dois estados quânticos usando um projetor Π .

$$T(\rho, \sigma) = \max_{\Pi} \text{tr} (\Pi(\rho - \sigma)) .$$

Duas propriedades da distância de traço são:

- $0 \leq T(\rho, \sigma) \leq 1$;
- $T(\rho, \sigma) = 0$ se e somente se $\rho = \sigma$.

Se ρ e σ comutam, então a distância de traço entre ρ e σ é igual à distância estatística, definida a seguir, entre as distribuições de probabilidade representadas pelos conjuntos de autovalores de ρ e σ . A *distância estatística*, ou *distância de variação total*, entre duas distribuições de probabilidade (ou variáveis aleatórias) X e Y definidas sobre o mesmo espaço amostral $\mathcal{U}$ é

$$d_{VT}(X, Y) = \Delta(X, Y) = \max_{S \subseteq \mathcal{U}} \{|\Pr[X \in S] - \Pr[Y \in S]|\} = \frac{1}{2} \sum_{x \in \mathcal{U}} |\Pr[X = x] - \Pr[Y = x]| .$$

Teorema 2.3 (Montanaro e de Wolf, 2016, Sec. 4.2, p. 44). *Seja $T(\rho, \sigma) \geq \epsilon$, sendo $\rho, \sigma \in \mathbf{D}(\mathcal{H})$ onde $\mathcal{H}$ tem dimensão N . Então, é possível distinguir ρ de σ com probabilidade maior que $2/3$ usando $O(N^2/\epsilon^2)$ cópias de ρ e em tempo polinomial em N/ϵ .*

Teorema 2.4 (Bernstein e Vazirani, 1997). *Sejam X e Y as distribuições dos resultados das medições de dois estados quânticos $|\psi\rangle$ e $|\phi\rangle$, respectivamente. Se $\| |\psi\rangle - |\phi\rangle \| = \epsilon$, então $d_{VT}(X, Y) \leq 4\epsilon$.*

Uma variável aleatória $v = (v_0, v_1, \dots, v_T)$ obedece a uma *distribuição de Markov* se v_i depende apenas de v_{i-1} . Isto não significa que v_i é independente de $v_0, v_1, \dots, v_{i-2}$, mas que qualquer dependência de v_i é capturada pelo valor de v_{i-1} .

Lema 2.5 (Aaronson et al., 2016). *Dados $T \geq 1$, e duas variáveis aleatórias $v = (v_0, \dots, v_T)$ e $w = (w_0, \dots, w_T)$ governadas por uma distribuição de Markov. A distância estatística entre v e w é*

$$d_{VT}(v, w) \leq 2 \sum_{i=1}^T d_{VT}((v_{i-1}, v_i), (w_{i-1}, w_i)) .$$

2.3 CIRCUITOS QUÂNTICOS

Além das referências que já indicamos na abertura do capítulo, esta seção baseia-se também no trabalho de Aharonov (2003).

Um *qubit* é um estado quântico em um espaço de Hilbert com dimensão 2,

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle ,$$

onde $\alpha, \beta \in \mathbb{C}$ e $|\alpha|^2 + |\beta|^2 = 1$.

Podemos medir o qubit com relação à base $\{|0\rangle, |1\rangle\}$. A probabilidade de obtermos o estado $|0\rangle$ é

$$\begin{aligned} \Pr[0] &= |\langle 0|\psi\rangle|^2 = |\langle 0| \cdot (\alpha |0\rangle + \beta |1\rangle)|^2 \\ &= |\alpha \langle 0|0\rangle + \beta \langle 0|1\rangle|^2 = |\alpha \cdot 1 + \beta \cdot 0|^2 = |\alpha|^2 . \end{aligned}$$

Calcula-se de modo similar a probabilidade para o estado $|1\rangle$. Isso significa que, obtemos ou com probabilidade $|\alpha|^2$ o estado $|0\rangle$ ou com probabilidade $|\beta|^2$ o estado $|1\rangle$.

Para um sistema com n qubits, o estado quântico do sistema pode ser descrito como

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

onde $\alpha_0, \dots, \alpha_{2^n-1} \in \mathbb{C}$ e $\sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1$. A base $\{|i\rangle\}$ é a base computacional.

Uma *porta quântica* para n qubits é um operador que faz uma transformação unitária de n qubits. Um *circuito quântico* é uma rede acíclica de portas quânticas agindo sobre qubits. Como toda rede acíclica possui ordenação topológica, por vezes representamos circuitos quânticos por uma sequência de operadores. Observe que em um circuito quântico o número de qubits de entrada é sempre igual ao número de qubits da saída. No entanto, por vezes nos referimos a circuitos com m qubits de entrada e $n < m$ qubits de saída, presumindo uma operação *trace-out* dos $m - n$ qubits que não fazem parte da saída. Para todo circuito quântico em que medições ocorrem em estágios intermediários, é possível construir um circuito quântico equivalente com

medições apenas no final, o que também é conhecido como *Princípio da Medição Postergada*, tratado formalmente em Gurevich e Blass (2022). Portanto, supomos que as medições ocorrem apenas no final do circuito.

Na Tabela 2.1 apresentamos alguns dos operadores quânticos, suas respectivas representações matriciais e diagramas no circuito.

Tabela 2.1: Diagrama e representação matricial de operadores quânticos.

Descrição	Circuito	Representação matricial
Um qubit		
Múltiplos qubits (número indeterminado)		
n qubits		
Um bit		
Uma medição		
Porta Hadamard		$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Porta CNOT		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$
Porta Toffoli (CCNOT)		$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$
U controlado		$cU = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & U \end{bmatrix}$

Um conjunto de portas quânticas S é *estritamente universal* se, para qualquer porta U para $n \geq 1$ qubits e qualquer $\epsilon > 0$, podemos construir um circuito quântico C com n qubits formado apenas por portas em S , tal que C implementa uma porta $\tilde{U}$ de modo que $\|\tilde{U} - U\| \leq \epsilon$.

Dado um circuito C formado com apenas portas de um conjunto estritamente universal, o *tamanho* de C é definido como o número de portas quânticas mais o número de qubits de C .

O *Teorema de Solovay–Kitaev* implica que qualquer porta quântica para $O(1)$ qubits pode ser aproximada com uma precisão arbitrária ϵ usando $O(\log^c(1/\epsilon))$ portas, onde c é uma constante, de um conjunto estritamente universal.

Um conjunto de portas quânticas S é *computacionalmente universal* se, para todo circuito quântico C com n qubits e t portas de um conjunto estritamente universal, é possível construir

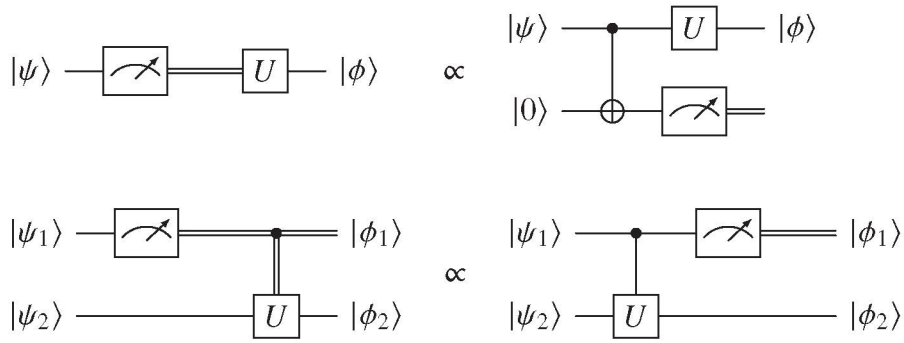


Figura 2.2: Os circuitos à esquerda (com medições intermediárias) são equivalentes aos circuitos à direita (com medições no final).

um circuito para simular C dentro de um erro ϵ com um *overhead* apenas polilogarítmico em $(n, t, 1/\epsilon)$. A definição de conjunto *computacionalmente universal* é uma generalização da definição de conjunto *estritamente universal* em que se permite, por exemplo, o uso de qubits auxiliares. Dentre os conjuntos de portas quânticas computacionalmente universais citamos o conjunto com as portas Hadamard e Toffoli.

Podemos aplicar o princípio da Medição Postergada em um circuito usando as transformações mostradas na Figura 2.2.

Se aplicarmos a porta Hadamard em n qubits, todos inicialmente iguais a 0, obtemos todas as 2^n possíveis strings binárias em sobreposição, todas com igual amplitude:

$$H^{\otimes n} |0\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle .$$

É comum, em algoritmos quânticos, o uso de operações *caixa-preta*. Sendo $f : \{0,1\}^n \rightarrow \{0,1\}^m$ uma função clássica, definimos U_f como a transformação unitária, para quaisquer $x \in \{0,1\}^n$ e $y \in \{0,1\}^m$,

$$U_f : |x, y\rangle \rightarrow |x, y \oplus f(x)\rangle ,$$

onde a operação $\oplus$ indica a adição módulo 2 bit a bit. Podemos aplicar a função $f(x)$ para diversos valores diferentes de x de modo simultâneo, o que é chamado de *paralelismo quântico*. Por exemplo, podemos colocar $f(x)$ em sobreposição para todos valores possíveis para x da seguinte forma:

$$U_f(H^{\otimes n} |0\rangle^{\otimes n}) |0\rangle^m = U_f \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle |0\rangle^m = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle |f(x)\rangle .$$

Um dos algoritmos quânticos mais famosos é o *algoritmo de Grover* (Grover, 1996). Ele resolve o problema de busca não estruturada no modelo caixa-preta de único elemento:

Busca não estruturada de único elemento

Entrada: um operador U_f que implementa uma função $f : \{0,1\}^n \rightarrow \{0,1\}$;

Saída: o único valor $w \in \{0,1\}^n$, tal que $f(w) = 1$.

O algoritmo de Grover, que tem um tempo de execução $O(\sqrt{N})$, é ótimo, i.e., não é possível resolver o problema de busca não estruturada usando um algoritmo quântico no modelo caixa-preta em tempo menor que $O(\sqrt{N})$ (Bennett et al., 1997).

2.4 CONCEITOS ELEMENTARES DE COMPLEXIDADE COMPUTACIONAL

Assumimos familiaridade do leitor com conceitos elementares da teoria da computação. O conteúdo desta seção baseia-se nos trabalhos de Papadimitriou (1994), Watrous (2009), Goldreich (2005) e Bennett et al. (1997).

A teoria de complexidade computacional classifica problemas computacionais de acordo com recursos, como tempo e espaço, necessários para resolvê-los. Costuma-se definir uma classe de complexidade como o conjunto de problemas que podem ser resolvidos com um modelo formal de computação dadas certas restrições.

Uma *máquina de Turing* é um modelo formal de computação definida por uma tupla $M = (K, \Sigma, \delta, s)$, onde:

- K é um conjunto finito de estados;
- s é o estado inicial;
- Σ é um conjunto de símbolos, denominado de alfabeto da máquina; e
- δ é uma função de transição que especifica o *programa* da máquina.

A máquina pode ser visualizada como tendo uma fita com um número infinito de posições, cada uma contendo um símbolo. A máquina tem um cabeçote de leitura e escrita sobre uma das posições da fita. De acordo com o estado e o símbolo lido a cada passo da execução da máquina, a função de transição δ determina:

- o símbolo a ser sobrescrito sob o cabeçote;
- para qual outra posição o cabeçote se moverá; e
- o próximo estado da máquina.

Se a máquina alcançar o estado “sim”, ela para e dizemos que a máquina *aceita* a entrada. Se a máquina alcançar o estado “não”, ela para e dizemos que a máquina *rejeita* a entrada.

Seja n o tamanho da entrada. Dizemos que a *máquina é polinomial*, se para toda entrada com tamanho n , a máquina para após $O(n^k)$ passos (em tais casos escrevemos simplesmente $\text{poly}(n)$ passos). Se ela parar sem ter aceitado a entrada, por definição, ela rejeita a entrada.

A máquina de Turing descrita é *determinística*, pois dados um estado e um símbolo, a ação da máquina é unicamente determinada. Em uma *Máquina de Turing não determinística*, dados um estado e um símbolo, a máquina poderá executar uma entre várias ações possíveis, em outras palavras, poderá usar um de diversos caminhos computacionais. Essa máquina aceita uma entrada, se houver ao menos um caminho que resulta no estado “sim”. Rejeita, se todos os caminhos resultarem no estado “não”. Já uma *máquina de Turing probabilística*, que também pode executar uma entre várias ações possíveis, recebe uma string adicional de bits aleatórios como entrada que determina qual ação será executada a cada passo. Essa máquina aceita se e somente se a maioria dos caminhos computacionais resultarem no estado “sim”.

Seja A uma linguagem. Uma máquina de Turing M^A com oráculo A é uma máquina de Turing com uma fita adicional de consulta. Quando a máquina escreve x nessa fita e entra no estado de consulta $q_?$, ela imediatamente segue para o estado q_s se $x \in A$, ou q_n se $x \notin A$.

Em teoria de complexidade computacional, é comum definir classes de complexidade como sendo conjuntos de linguagens (i.e. problemas de decisão). No entanto, em trabalhos da área que envolvem conceitos como *conhecimento zero* e *computação quântica*, é conveniente definir classes de complexidade usando *problemas de promessa*, uma generalização de problemas de decisão. Um *problema de promessa* A é um par de conjuntos disjuntos (A_S, A_N) tal que $A_S, A_N \subseteq \{0, 1\}^*$ e $A_S \cap A_N = \emptyset$. As strings A_S são as *instâncias positivas* e as strings A_N são as *instâncias negativas*. O conjunto $A_S \cup A_N$ é a *promessa*. O *complemento* do problema de promessa A é o problema de promessa $\bar{A}$ onde $\bar{A}_S = A_N$ e $\bar{A}_N = A_S$. Observe que um problema de *decisão* é um problema de promessa em que $A_S \cup A_N = \{0, 1\}^*$.

Um problema de promessa A tem uma *redução de Karp* para um problema de promessa B se há uma função $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$ computável por uma máquina de Turing de tempo polinomial de modo que

- se $x \in A_S$, então $f(x) \in B_S$;
- se $x \in A_N$, então $f(x) \in B_N$.

Se C é uma classe de problemas de promessa, dizemos que o problema de promessa A é *difícil* para C , se todo problema de promessa em C reduz para A . O problema de promessa A é *completo* para C se $A \in C$ e é difícil para C .

O principal motivo pelo qual trabalhamos com problemas de promessa quando estudamos diversas classes de interesse neste trabalho, como BPP, BQP, SZK, é porque não se conhecem problemas de decisão que sejam completos para essas classes, mas se conhecem problemas de promessa que o sejam. Alguns autores diferem as versões dessas classes para problemas de promessa acompanhando-as do prefixo *promise*, como em Promise BPP, Promise BQP, Promise SZK (Janzing e Wocjan, 2007). Nós optamos, neste trabalho, por omitir esse prefixo, assim como diversos autores que trabalham apenas com problemas de promessa. Observamos, contudo, que se trata de um abuso de notação e que classes de problemas de decisão são subclasses de classes de problemas de promessa.

A *classe* P é a classe dos problemas $A = (A_S, A_N)$ que admitem uma máquina de Turing determinística de tempo polinomial que aceita toda string $x \in A_S$ e rejeita toda string $x \in A_N$.

A *classe* NP é a classe dos problemas $A = (A_S, A_N)$ para os quais existe um polinômio $p(\cdot)$ e uma máquina de Turing determinística de tempo polinomial de modo que:

- se $x \in A_S$, então M aceita (x, y) para alguma string $y \in \{0, 1\}^{p(|x|)}$;
- se $x \in A_N$, então M rejeita (x, y) para toda string $y \in \{0, 1\}^{p(|x|)}$.

Equivalentemente, A está em NP se e somente se A é decidida por uma máquina de Turing não determinística polinomial.

A *classe* $PSPACE$ ($NPSpace$) é a classe dos problemas $A = (A_S, A_N)$ que admitem uma máquina de Turing determinística (não determinística) de espaço polinomial que aceita toda string $x \in A_S$ e rejeita toda string $x \in A_N$. Do Teorema de Savitch, $PSPACE = NPSpace$ (Savitch, 1970).

A *classe* EXP é a classe dos problemas $A = (A_S, A_N)$ que admitem uma máquina de Turing determinística de tempo exponencial que aceita toda string $x \in A_S$ e rejeita toda string $x \in A_N$. É fácil ver que

$$P \subseteq NP \subseteq PSPACE \subseteq EXP.$$

Teorema 2.6 (Baker et al., 1975). *Existe um oráculo A , tal que $P^A = NP^A$.*

Demonstração. Sendo A qualquer linguagem PSPACE-completa, temos

$$P^A = \text{PSPACE} = \text{NPSPACE} = \text{NP}^A .$$

□

Teorema 2.7 (Baker-Gill-Solovay, Baker et al.1975). *Existe um oráculo B , tal que $P^B \neq \text{NP}^B$.*

Demonstração. Para o oráculo B que vamos construir, definimos uma linguagem unária L :

$$L = \{0^n : \text{existe } x \in B \text{ tal que } |x| = n\} .$$

Note que $L \in \text{NP}^B$, pois, dada a entrada 0^n , uma máquina não-determinística com oráculo B pode encontrar um x com tamanho n e verificar que $x \in B$.

Queremos definir $B \subseteq \{0, 1\}^*$ tal que $L \notin P^B$. Fazemos isso usando um argumento de *diagonalização* sobre todas máquinas de Turing determinísticas polinomiais com oráculo. Considere que temos uma enumeração $M_1^?, M_2^?, \dots$ de todas essas máquinas. Assumimos que toda máquina aparece nessa enumeração um número infinito de vezes. Podemos assumir isso por um argumento de *padding*: qualquer máquina pode ser arbitrariamente preenchida com estados inúteis de modo a não mudar a linguagem decidida, e essas variantes precisam aparecer na enumeração.

Construímos B em estágios. Denotamos $B_0 = \emptyset$ e B_i como o conjunto de strings de tamanho menor ou igual a i em B . Assumimos que no início de cada estágio i , B_{i-1} já foi determinado. Definimos também um conjunto auxiliar $X \subseteq \Sigma^*$ de strings que não poderão ser colocadas em B . Inicialmente, $X = \emptyset$.

Decidimos quais strings de tamanho i devem estar em B simulando $M_i^B(0^i)$ para $i^{\log i}$ passos. O número de passos foi escolhido de modo a ser uma função menor que exponencial, mas ainda assintoticamente maior que um polinômio.

Durante a simulação de $M_i^B(0^i)$, poderá ser feita a consulta $x \in B$. Se $|x| < i$, a resposta será positiva se $x \in B_{i-1}$ (negativa se $x \notin B_{i-1}$), de modo que M_i^B irá para o estado q_S (q_N). Agora, se $|x| \geq i$, então M_i irá para o estado q_N e adicionamos x ao conjunto X , de modo a sempre lembrarmos que $x \notin B$.

Suponha que M_i^B termine a computação em até $i^{\log i}$ passos e rejeite 0^i . Nesse caso, definimos B_i como sendo $B_{i-1} \cup \{x \in \{0, 1\}^* : |x| = i, x \notin X\}$. Desse modo, asseguramos que $0^i \in L$ (pois temos uma string de tamanho i em B) e, ao mesmo tempo, que M_i^B rejeita 0^i . Portanto, M_i^B não decide L .

Precisamos mostrar que o conjunto $\{x \in \{0, 1\}^* : |x| = i, x \notin X\}$ não é vazio. Ora, o número total de passos simulados para todas as máquinas é $m = \sum_{j=1}^i j^{\log j}$, portanto X não contém mais que m strings de tamanho i . Esse número é menor que 2^i (número de strings em $\{0, 1\}^i$).

Se M_i^B aceita 0^i em até $i^{\log i}$ passos, definimos $B_i = B_{i-1}$. Isso assegura que $0^i \notin L$, e, portanto, M_i^B não decide L .

Precisamos ainda analisar o caso em que $M_i^B(0^i)$ não termina a computação em $i^{\log i}$ passos, ou seja, se $i^{\log i} < p(i)$. Nesse caso, definimos $B_i = B_{i-1}$. Ora, máquinas equivalentes a essa aparecem de modo infinito na enumeração, de modo que haverá ao menos uma máquina M_I^B equivalente tal que $I^{\log I} \geq p(I)$. Isso assegura que M_I^B não decide L .

Terminando a construção de B , concluímos que todas máquinas de Turing determinísticas polinomiais com oráculo não decidem L , portanto, $L \notin P^B$. □

A classe UP é a classe dos problemas $A = (A_S, A_N)$ para os quais existe um polinômio $p(\cdot)$ e uma máquina de Turing determinística de tempo polinomial de modo que:

- se $x \in A_S$, então M aceita (x, y) para exatamente uma string $y \in \{0, 1\}^{p(|x|)}$;
- se $x \in A_N$, então M rejeita (x, y) para toda string $y \in \{0, 1\}^{p(|x|)}$.

A *classe* PP (tempo polinomial probabilístico) é a classe dos problemas $A = (A_S, A_N)$ que admitem uma máquina de Turing probabilística de tempo polinomial M de modo que:

- se $x \in A_S$, então $\Pr[M \text{ aceita } x] > 1/2$;
- se $x \in A_N$, então $\Pr[M \text{ aceita } x] \leq 1/2$.

A *classe* BPP (*bounded-error probabilistic polynomial-time*) é a classe dos problemas $A = (A_S, A_N)$ que admitem uma máquina de Turing probabilística de tempo polinomial M de modo que:

- se $x \in A_S$, então $\Pr[M \text{ aceita } x] \geq 2/3$;
- se $x \in A_N$, então $\Pr[M \text{ aceita } x] \leq 1/3$.

Para todo $\epsilon > 0$, é possível ter a probabilidade de erro da máquina M menor que ϵ executando-se M para a entrada $\text{polylog}(1/\epsilon)$ vezes. Nesse sentido, as constantes $2/3$ e $1/3$ são arbitrárias, e o mesmo raciocínio pode ser reproduzido para outras classes que virão a seguir, embora não possa ser reproduzido para PP.

A *classe* SBP (*Small Bounded-Error Probability*), introduzida por Böhler et al. (2006), é a classe dos problemas $A = (A_S, A_N)$ para os quais existe uma constante $\epsilon > 0$, um polinômio $p(\cdot)$ e uma máquina de Turing probabilística de tempo polinomial de modo que:

- se $x \in A_S$, então $\Pr[M \text{ aceita } x] \geq (1 + \epsilon)/2^{p(|x|)}$;
- se $x \in A_N$, então $\Pr[M \text{ aceita } x] \leq 1/2^{p(|x|)}$.

Uma família $\mathcal{Q} = \{Q_n : n \in \mathbb{N}\}$ de circuitos quânticos é *gerada uniformemente em tempo polinomial* se existe uma máquina de Turing determinística polinomial, que, ao receber como entrada 1^n , retorna uma descrição de Q_n . A uniformidade se refere ao fato de que a família infinita é descrita de modo finito por uma máquina de Turing.

Uma maneira mais geral, segundo Watrous (2009), para se definir uma *família uniforme polinomial de circuitos quânticos* é: sendo $S \subseteq \{0, 1\}^*$, 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, também chamada de *algoritmo* BQP, tal que, para toda entrada $x \in S$, retorna uma descrição de Q_x . Dado um polinômio $p(\cdot)$, 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* BQP é a classe dos problemas $A = (A_S, A_N)$ que admitem uma família uniforme polinomial 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 $x \in A_S$, então $\Pr[Q_x |0\rangle^{\otimes p(|x|)} = 1] \geq 2/3$;
- se $x \in A_N$, então $\Pr[Q_x |0\rangle^{\otimes p(|x|)} = 1] \leq 1/3$.

De modo mais geral, definimos *algoritmo* $\text{BQTime}(T(|x|))$ como uma máquina de Turing determinística de tempo limitado por uma função $T(|x|)$ (no lugar de um polinômio $p(|x|)$), tal que, para toda entrada $x \in S$, retorna uma descrição de um circuito quântico (não necessariamente polinomial) Q_x . De modo análogo, definimos a *classe* $\text{BQTime}(T(|x|))$.

Um *algoritmo* BQP com consulta a um oráculo é um algoritmo BQP em que cada um dos operadores unitários $U_1, \dots, U_R$ pode estar associado a uma função $f : \{0, 1\}^n \rightarrow \{0, 1\}$, isso é, definimos para a construção de $U_1, \dots, U_R$, para quaisquer $x \in \{0, 1\}^n$ e $b \in \{0, 1\}$,

$$U_f : |x, b\rangle \rightarrow |x, b \oplus f(x)\rangle ,$$

onde a operação $\oplus$ indica a adição módulo 2, ou, de modo equivalente,

$$U_f |x\rangle = (-1)^{f(x)} |x\rangle , \quad \text{para } x \in \{0, 1\}^n.$$

Durante esse texto, assumimos sem perda de generalidade que cada uma das transformações unitárias $U_1, \dots, U_R$, de um algoritmo BQP com consulta a um oráculo f , é ou uma cópia de U_f , ou um circuito de n qubits construído usando apenas portas de um conjunto universal fixo. Sendo A uma linguagem, um *algoritmo* BQP com oráculo A é um algoritmo BQP com consulta a oráculo para a função f_A definida por $f_A(x) = 1$ se e somente se $x \in A$.

Definição 2.8. Seja y uma string binária, definimos $q_y(|\phi_t\rangle)$ como o módulo ao quadrado da amplitude de $|y\rangle$ em $|\phi_t\rangle$ se U_t é uma cópia de U_f , e 0 caso contrário.

Note que, como $|\phi_t\rangle$ pode ser vista como a superposição de todos os estados no tempo t , temos que $q_y(|\phi_t\rangle)$ é o módulo ao quadrado da amplitude em $|\phi_t\rangle$ correspondendo apenas ao estado do tempo t que consulta o oráculo A quanto a string y . Por exemplo, seja $|\phi_t\rangle = \alpha |01\rangle |0\rangle + \beta |10\rangle |0\rangle$, então, $q_{01}(|\phi_t\rangle) = |\alpha|^2$.

2.5 CLASSES DE PROVAS INTERATIVAS E DE CONHECIMENTO ZERO ESTATÍSTICO

Nesta seção, as definições e resultados referentes a provas interativas e conhecimento zero baseiam-se nos textos de Vadhan (1999), Goldreich (2007) e Sdroievski (2018); as definições e resultados referentes a provas interativas quânticas e de conhecimento zero baseiam-se nos textos de Watrous (2002), Kobayashi (2003) e Yan (2012).

Um sistema de *prova clássica* para um problema de promessa A é dado por um algoritmo de verificação V e um polinômio $p(\cdot)$ tal que:

1. (Eficiência) o algoritmo V executa de modo determinístico em tempo polinomial.
2. (Abrangência) se $x \in A_S$, então existe uma string u com tamanho no máximo $p(|x|)$ tal que V aceita (x, u) . A string u é chamada de prova, testemunha ou certificado de x .
3. (Credibilidade) se $x \in A_N$, então V rejeita (x, u) para todo u .

Observe que a classe NP pode ser vista como a classe de problemas que admitem provas clássicas.

O termo *abrangência* é a tradução de *completeness*. Um algoritmo goza de *abrangência* se o conjunto de strings aceitas pelo algoritmo *abrange* é o conjunto das instâncias positivas. Outras traduções encontradas são *integralidade* e *completude*. O termo *credibilidade* é a tradução de *soundness*. Um algoritmo goza de *credibilidade* se não abranger nada mais do que deveria. Outra tradução encontrada é *corretude*.

Um *protocolo interativo* (A, B) é um par de funções de $\{0, 1\}^*$ em $\{0, 1\}^*$. Sendo r_A e r_B seqüências de bits aleatórios e uniformes, a *interação* entre A e B com k mensagens, dada uma entrada em comum x , é o processo $(A, B)(x)$ de troca de k mensagens m_i definido por:

- para $0 \leq i \leq k$, onde i é ímpar, $m_{i+1} = A(x, m_1, \dots, m_i; r_A)$;
- para $0 < i \leq k$, onde i é par, $m_{i+1} = B(x, m_1, \dots, m_i; r_B)$;
- $m_{k+1} \in \{\text{aceite}, \text{rejeite}\}$.

A última mensagem m_{k+1} não é contada como parte da troca de mensagens. Se m_{k+1} é *aceite* (ou *rejeite*), dizemos que A (ou B) aceita (ou rejeita) x . Um protocolo é polinomialmente limitado se $k = \text{poly}(|x|)$ mensagens são trocadas, tendo cada uma tamanho $\text{poly}(|x|)$.

Seja (P, V) um protocolo interativo, sendo V um algoritmo probabilístico de verificação, e A um problema de promessa. Dizemos que (P, V) é um *sistema de prova interativa* (com k mensagens) para A se:

1. (Eficiência) (P, V) é polinomialmente limitado e V é computável em tempo polinomial;
2. (Abrangência) se $x \in A_S$, então V aceita x com probabilidade pelo menos $2/3$ na interação $(P, V)(x)$ (com k mensagens);
3. (Credibilidade) se $x \in A_N$, então para qualquer P^* , o verificador V rejeita x com probabilidade pelo menos $2/3$ na interação $(P^*, V)(x)$ (com k mensagens).

Se, para todo $x \in A_S$, o verificador V aceita x com probabilidade igual a um na interação $(P, V)(x)$, dizemos que esse sistema de prova interativa tem *abrangência perfeita*. A classe $IP(k)$ é a classe dos problemas que admitem sistemas de prova interativa com k mensagens; a classe $IP(0)$ é igual à classe BPP ; a classe $IP(2)$ é também conhecida como a classe **AM** (Arthur–Merlin); a classe IP é a classe dos problemas $\cup_{c \geq 1} IP(k^c)$.

Um *protocolo não-interativo* (P, V) é um par qualquer de funções $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$. Sendo r_P e r_V seqüências de bits aleatórios e uniformes, a *comunicação* entre P e V , dada uma entrada em comum x , é o processo $(P, V)(x)$ definido por:

1. seja $m = P(x, r_P)$;
2. a resposta é $m = V(x, r_V)$.

Se a resposta é *aceite* (ou *rejeite*) dizemos que V aceita (ou rejeita) x . O protocolo é polinomialmente limitado se $|m| = \text{poly}(|x|)$.

Seja (P, V) um protocolo não-interativo e A um problema de promessa. Dizemos que (P, V) é um *sistema de prova não-interativo* para A se:

1. (Eficiência). (P, V) é polinomialmente limitado e V é computável em tempo polinomial.
2. (Abrangência). Se $x \in A_S$, então existe um provador P que convence V a aceitar x com probabilidade pelo menos $2/3$ na interação $(P, V)(x)$.
3. (Credibilidade). Se $x \in A_N$, então para qualquer P^* , o verificador V rejeita x com probabilidade pelo menos $2/3$ na interação $(P, V)(x)$.

A classe MA (Merlin–Arthur) é a classe dos problemas de promessa que admitem provas não-iterativas.

Um *protocolo não-iterativo com string compartilhada* (P, V) é um par qualquer de funções $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$ mais uma função computável em tempo polinomial $l : \{0, 1\}^* \rightarrow \mathbb{N}$. A *comunicação* de P para V dada uma entrada em comum x é o processo $(P, V)(x)$ definido por:

1. selecione a string aleatória compartilhada $r \leftarrow \{0, 1\}^{l(x)}$;
2. seja $m = P(x, r)$;
3. a resposta é $m = V(x, r, m)$.

Se a resposta é *aceite* (ou *rejeite*) dizemos que B aceita (ou rejeita) x . O protocolo é polinomialmente limitado se $l(x)$ e $|m|$ são limitados por $\text{poly}(|x|)$.

Seja (P, V) um protocolo não-iterativo com string compartilhada e A um problema de promessa. Dizemos que (P, V) é um *sistema de prova não-iterativa com string compartilhada* para A se:

1. (Eficiência) (P, V) é polinomialmente limitado e V é computável em tempo polinomial.
2. (Abrangência) se $x \in A_S$, então existe P que convence V a aceitar x com probabilidade pelo menos $2/3$ na interação $(P, V)(x)$.
3. (Credibilidade) se $x \in A_N$, então para qualquer P , o verificador V rejeita x com probabilidade pelo menos $2/3$ na interação $(P, V)(x)$.

A visão de um protocolo iterativo consiste em todas as informações que o verificador tem durante a interação. Seja (P, V) um protocolo iterativo. A *visão* de V com relação a (P, V) sobre a entrada em comum x é a variável aleatória

$$\langle P, V \rangle(x) = (m_1, \dots, m_k; r)$$

sendo $m_1, \dots, m_k$ as mensagens trocadas entre P e V e r a substring de r_v contendo todos os bits aleatórios lidos por V durante a interação.

Uma função $\mu : \mathbb{N} \rightarrow [0, 1]$ é *negligenciável* se, para todo polinômio $p : \mathbb{N} \rightarrow \mathbb{N}$, temos $\mu(k) < 1/p(k)$ para k suficientemente grande. Um *simulador* S de um sistema de prova interativa (P, V) para um problema de promessa A é um algoritmo probabilístico polinomial que, dada uma instância x do problema, retorna um conjunto de strings, o qual queremos que seja o mais parecido possível com a visão $\langle P, V \rangle(x)$. Por definição, permite-se que o simulador falhe, tendo, nesse caso, como saída a string *falha*. Chamamos um simulador S de *útil* se $\Pr[S(x) = \text{falha}] \leq 1/2$ para todo $x \in A_S$, ou, equivalentemente, se $\Pr[S(x) = \text{falha}] \leq \mu(|x|)$ para todo $x \in A_S$ e para uma função negligenciável μ . Observe que não impomos restrição à probabilidade de falha do simulador para $x \notin A_S$. Definimos $\tilde{S}(x)$ como sendo a saída de S dada a entrada x , *condicionado* a que $S(x) \neq \text{falha}$.

Um sistema de prova interativa (P, V) (não-iterativa com string compartilhada) *para verificadores honestos* para um problema de promessa A é de *conhecimento zero estatístico* se existe um simulador útil com probabilidade uniforme S e uma função negligenciável μ tal que, para quaisquer $x \in A_S$ e $k > 0$, temos

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

A função negligenciável μ é chamada de *desvio do simulador*. Se $\mu = 0$, então (P, V) é um sistema de prova interativa de *conhecimento zero perfeito*.

Sejam o simulador útil S e o verificador V^* algoritmos probabilísticos *não uniformes* polinomiais.¹ Um sistema de prova interativa (P, V) para verificadores desonestos para um problema de promessa A é de *conhecimento zero estatístico* se existe um simulador útil S e uma função negligenciável μ tal que, para quaisquer $x \in A_S$ e $k > 0$, temos

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

A função negligenciável μ é chamada de *desvio do simulador*. Se $\mu = 0$, então (P, V) é um sistema de prova interativa de *conhecimento zero perfeito*.

A classe SZK é a classe dos problemas que admitem sistemas de prova interativa de conhecimento zero estatístico. Caso se requeira que os verificadores sejam honestos, a classe é denotada por HVSZK. Entretanto, como HVSZK = SZK (Goldreich et al., 1998) não fazemos distinção entre verificadores honestos e desonestos quando estamos lidando com a classe SZK.

A classe PZK é a classe dos problemas que admitem sistemas de prova interativa com verificadores desonestos de conhecimento zero perfeito. A classe HVPZK é a classe dos problemas que admitem sistemas de prova interativa com verificadores honestos de conhecimento zero perfeito.

A classe PZK₁ é a classe dos problemas que admitem sistemas de prova interativa de conhecimento zero perfeito com abrangência perfeita. A classe NISZK é a classe dos problemas que admitem sistemas de prova não-interativa com string compartilhada de conhecimento zero estatístico. A classe NIPZK é a classe dos problemas que admitem sistemas de prova não-interativa com string compartilhada de conhecimento zero perfeito. A classe NIPZK₁ é a classe dos problemas que admitem sistemas de prova não-interativa com string compartilhada de conhecimento zero perfeito com abrangência perfeita.

A classe SZK é fechada sob complemento, isto é, SZK = coSZK (Okamoto, 1996). Não se sabe se NISZK é fechada sob complemento, mas se o for, então SZK = NISZK; e se SZK $\neq$ BPP então NISZK $\neq$ BPP (Goldreich et al., 1999). A classe SZK está contida em $AM \cap coAM$ (Fortnow, 1987; Aiello e Hastad, 1991). A classe NIPZK está contida em coSBP (Dixon et al., 2020).

Um *protocolo interativo quântico* é um par (P, V) que descreve a interação entre um provador quântico P e um verificador quântico V . Sendo $k(\cdot)$ um polinômio, o *verificador quântico* é descrito por uma família uniforme polinomial de circuitos quânticos

$$V = \left\{ V_x^j : x \in \{0, 1\}^*, 1 \leq j \leq k(|x|) \right\} .$$

Seja $n = |x|$, cada circuito V_x^j tem como entrada e saída $q_V(n) + q_M(n)$ qubits, sendo $q_V(n)$ qubits privados para o verificador e $q_M(n)$ qubits enviados, ou recebidos, no j -ésimo *turno* na interação entre o verificador e o provador. O *provador quântico* é uma família arbitrária de transformações unitárias

$$P = \left\{ P_x^j : x \in \{0, 1\}^*, 1 \leq j \leq l(|x|) \right\} .$$

Cada circuito P_x^j tem como entrada e saída $q_P(n) + q_M(n)$ qubits, sendo $q_P(n)$ qubits privados para o provador e $q_M(n)$ qubits enviados, ou recebidos, no j -ésimo turno da interação. A

¹Um algoritmo não uniforme M , também conhecido como máquina de Turing com conselho, é um algoritmo que recebe uma string adicional, chamada de conselho, dada por uma função $A : \mathbb{Z}_+ \rightarrow \Sigma^*$. O algoritmo M decide uma linguagem L com o conselho $A(n)$ se $x \in L$ implica em $M(x, A(|x|))$ aceitar, e $x \notin L$ implica em $M(x, A(|x|))$ rejeitar.

interação entre o verificador e o provador por meio de mensagens é exemplificada na Figura 2.3. Sendo $m(n)$ o número de mensagens trocadas entre o provador e o verificador, observe que é sempre o verificador a receber a última mensagem, diferentemente do que ocorre com os protocolos interativos clássicos. Desse modo, $k(n) = \lfloor m(n)/2 \rfloor + 1$ e $l(n) = \lceil m(n)/2 \rceil$. O circuito quântico (P, V) , com $q_V(n) + q_M(n) + q_P(n)$ qubits de entrada, inicialmente iguais a $|0\rangle$, e um qubit de saída, é dado pela sequência, se $m(n)$ é par,

$$\overbrace{V_x^1, P_x^1, \dots, P_x^{l(n)}, V_x^{k(n)}}^{1^\circ \text{ turno}},$$

e, se $m(n)$ é ímpar,

$$\overbrace{P_x^1, V_x^1, \dots, P_x^{l(n)}, V_x^{k(n)}}^{1^\circ \text{ turno}}.$$

Um dos qubits privados para o verificador é o qubit de saída que, ao ser medido, indica se o verificador aceita ou rejeita x .

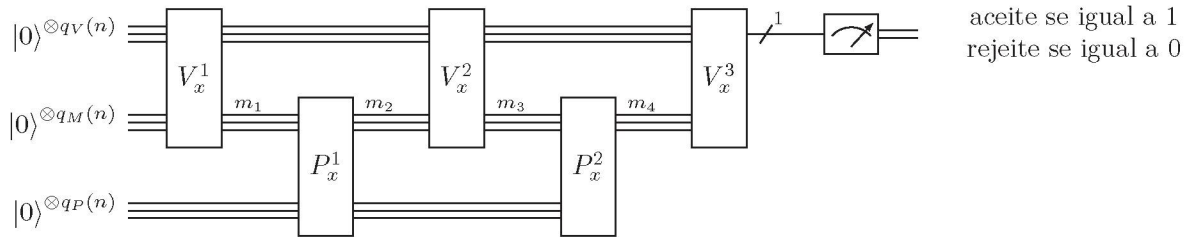


Figura 2.3: Um protocolo interativo quântico com 4 mensagens nomeados como m_1, m_2, m_3 e m_4 . O circuito tem como entrada $q_V(n) + q_M(n) + q_P(n)$ qubits inicializados em $|0\rangle$ e como saída um qubit que é medido. Se a medição resultar em 1 dizemos que o verificador aceita x , caso contrário ele rejeita x .

Sejam (P, V) um protocolo interativo quântico de $q(n)$ qubits e A um problema de promessa. Dizemos que (P, V) é um *sistema de prova interativa quântica* (com m mensagens) para A se:

1. (Eficiência) (P, V) é polinomialmente limitado e V é computável em tempo polinomial;
2. (Abrangência) se $x \in A_S$, então V aceita x com probabilidade pelo menos $2/3$ na ação do circuito (P, V) (com m mensagens) sobre $|0\rangle^{\otimes q(n)}$;
3. (Credibilidade) se $x \in A_N$, então para qualquer P^* , o verificador V rejeita x com probabilidade pelo menos $2/3$ na ação do circuito (P^*, V) (com m mensagens) sobre $|0\rangle^{\otimes q(n)}$.

A classe $\text{QIP}(m)$ é a classe dos problemas que admitem provas interativas quânticas com m mensagens; a classe $\text{QIP}(0)$ é igual a BQP ; a classe $\text{QIP}(1)$ é também conhecida como QMA ; a classe coQMA é o complemento de QMA ; a classe $\text{QIP}(2)$ é também conhecida como a classe QAM ; a classe QIP é a classe dos problemas $\cup_{c \geq 1} \text{QIP}(m^c)$.

Um *protocolo não-interativo quântico com estados de Bell compartilhados* é um par (P, V) que descreve a comunicação entre um provador quântico P e um verificador quântico V que compartilham entre si estados de Bell. O *provador quântico* é descrito por uma família arbitrária de transformações unitárias

$$P = \{P_x : x \in \{0, 1\}^*\}.$$

O circuito P recebe como entrada $q_P(n) + q_M(n)$ qubits inicializados em $|0\rangle$ mais o primeiro qubit de cada um de $q_S(n)$ estados de Bell $(|00\rangle + |11\rangle)/\sqrt{2}$. O circuito P envia para V uma mensagem com $q_M(n)$ qubits. O *verificador quântico* é descrito por uma família uniforme polinomial de circuitos quânticos

$$V = \{V_x : x \in \{0, 1\}^*\} .$$

O circuito V recebe como entrada $q_V(n)$ qubits inicializados em $|0\rangle$, os $q_M(n)$ qubits enviados por P , mais o segundo qubit de cada um de $q_S(n)$ estados de Bell $(|00\rangle + |11\rangle)/\sqrt{2}$. O circuito V tem como saída um qubit que ao ser medido indica se o verificador aceita ou rejeita x .

Sejam (P, V) um protocolo não-interativo quântico de $q(n)$ qubits com estados de Bell compartilhados e A um problema de promessa. Dizemos que (P, V) é um *sistema de prova não-interativa quântica* para A se:

1. (Eficiência) (P, V) é polinomialmente limitado e V é computável em tempo polinomial;
2. (Abrangência) se $x \in A_S$, então V aceita x com probabilidade pelo menos $2/3$ na ação do circuito (P, V) sobre $|0\rangle^{\otimes q(n)}$;
3. (Credibilidade) se $x \in A_N$, então para qualquer P^* , o verificador V rejeita x com probabilidade pelo menos $2/3$ na ação do circuito (P^*, V) sobre $|0\rangle^{\otimes q(n)}$.

Dada uma coleção de $\{\rho_y\}$ de estados quânticos mistos, dizemos que essa coleção é *preparada em tempo polinomial* se existe uma família uniforme polinomial $\{Q_y\}$ de circuitos quânticos, de modo, que para cada y , o estado ρ_y é o estado misto obtido com a ação Q_y sobre $|0\rangle^{\text{poly}(|y|)}$ e fazendo-se o trace-out dos qubits que não fazem parte da saída.

Dado um provador P e um verificador V , definimos a *visão* $\text{visão}_{V,P}(x, m)$ como o estado dos qubits privados do verificador e da mensagem, após m mensagens serem enviadas na execução do sistema de prova (V, P) com a entrada x . Em outras palavras, esse é o estado do sistema obtido após serem trocadas m mensagens e tendo sido feito o trace-out dos qubits privados do provador.

Um sistema de prova interativa (não-interativa) quântica (P, V) para um problema de promessa A é de *conhecimento zero estatístico* se existe um conjunto preparável $\sigma_{x,m}$, dado por uma família uniforme polinomial de circuitos quânticos, e uma função negligenciável μ tais que, para todo $x \in A_S$ e para cada mensagem de número m temos

$$\|\sigma_{x,m} - \text{visão}_{V,P}(x, m)\|_{tr} < \mu(n) .$$

Se $\mu = 0$, então (P, V) é um sistema de prova interativa quântico de *conhecimento zero perfeito*. A classe QSZK é a classe dos problemas que admitem sistemas de prova interativa quântica de conhecimento zero estatístico. A classe QSZK é fechada sob complemento. A classe QPZK é a classe dos problemas que admitem sistemas de prova interativa quântica de conhecimento zero perfeito. Por sua vez, QPZK₁ é a classe dos problemas que admitem sistemas de prova interativa quântica de conhecimento zero perfeito com abrangência perfeita. A classe NIQSZK é a classe dos problemas que admitem sistemas de prova não-interativa quântica de conhecimento zero estatístico. A classe NIQPZK é a classe dos problemas que admitem sistemas de prova não-interativa quântica de conhecimento zero perfeito. A classe NIQPZK₁ é a classe dos problemas que admitem sistemas de prova não-interativa quântica de conhecimento zero perfeito com abrangência perfeita.

3 REVISÃO DA LITERATURA

Neste capítulo apresentamos uma breve revisão sobre problemas para classes de provas de conhecimento zero estatístico, conceitos sobre computação quântica sem colapso, e separação oracular entre UP e BQP, assim como entre naCQP e BQP.

3.1 PROBLEMAS PARA CLASSES DE CONHECIMENTO ZERO ESTATÍSTICO

Dada uma distribuição de probabilidade X sobre todas as 2^n strings com n bits, dizemos que um circuito C *codifica* X se, dada uma entrada $x \in \{0, 1\}^m$ escolhida de modo uniforme aleatório, a probabilidade de se obter na saída de C a string $y \in \{0, 1\}^n$ é dada por X .

Um problema completo para SZK é o problema *Distância Estatística* (SD, de *Statistical Difference*) (Sahai e Vadhan, 2003), em que precisamos comparar duas distribuições de probabilidade. Sendo α e β dois números reais tais que $0 \leq \alpha < \beta^2 \leq 1$, esse problema é definido como:

(α, β) -SD

Dados: dois inteiros não-negativos m e n , e circuitos booleanos C e D de tamanho $\text{poly}(m + n)$ com m bits de entrada e n bits de saída, os quais codificam, respectivamente, distribuições de probabilidade X e Y sobre todas as strings com n bits;

decidir entre: *instância positiva:* $d_{VT}(X, Y) \leq \alpha$;
instância negativa: $d_{VT}(X, Y) \geq \beta$;

prometido que vale um dos casos.

Se ambos os circuitos C e D tiverem apenas um bit de saída, o problema de promessa (α, β) -SD é completo para BPP. Como SZK = coSZK, o complemento de SD também é um problema completo para SZK.

Podemos redefinir de modo equivalente o problema (α, β) -SD usando o Lema da Polarização.

Lema 3.1 (Lema da Polarização (Sahai e Vadhan, 2003)). *Sejam $\alpha, \beta \in [0, 1]$ duas constantes tais que $\beta^2 > \alpha$ (por exemplo, $\beta = 2/3$ e $\alpha = 1/3$). Sejam C_0 e C_1 circuitos que codificam duas distribuições de probabilidade X_0 e X_1 , respectivamente. Existe uma função computável em tempo polinomial que dada a tripla $(C_0, C_1, 1^k)$, devolve dois circuitos D_0 e D_1 , codificando duas distribuições de probabilidade Y_0 e Y_1 , de modo que*

- *se $d_{VT}(X_0, X_1) \geq \beta$ então $d_{VT}(Y_0, Y_1) \geq 1 - 2^{-k}$;*
- *se $d_{VT}(X_0, X_1) \leq \alpha$ então $d_{VT}(Y_0, Y_1) \leq 2^{-k}$.*

Assim, podemos restringir o problema (α, β) -SD a $(2^{-\text{poly}(n)}, 1 - 2^{-\text{poly}(n)})$ -SD, chamando esse problema apenas de SD.

Dizemos que as distribuições de probabilidade são *distribuições caixa-preta* se as descrições dos circuitos que codificam as distribuições não puderem ser usadas. Bouland et al. (2019) observam que ao polarizar as distribuições, usando o Lema 3.1, o tamanho das distribuições de saída também aumentam do seguinte modo: o tamanho de Y_0 e Y_1 é uma função polinomial do tamanho de X_0 e X_1 e do valor k . Portanto, existe uma restrição para o valor k em termos do tamanho da distribuição. Bouland et al. demonstram a seguinte restrição.

Teorema 3.2 (Boulund et al., 2019). *Sejam D_0 e D_1 duas distribuições caixa-preta, sobre strings com n bits, que obedecem $d_{VT}(D_0, D_1) \leq 1/3$ ou $d_{VT}(D_0, D_1) \geq 2/3$. Então **não** existe um algoritmo que em tempo $\text{poly}(n)$ produz duas distribuições D'_0 e D'_1 , sobre strings de tamanho $n' = \text{poly}(n)$, tal que $d_{VT}(D_0, D_1) \leq \epsilon$ ou $d_{VT}(D_0, D_1) \geq 1 - \epsilon$ onde $\epsilon \leq 2^{-n'/2-1}$.*

Um problema completo para PZK_1 (PZK com abrangência perfeita) é o problema SD_1 (Sahai e Vadhan, 2003).

SD_1

Dados: dois inteiros não-negativos m e n , e circuitos booleanos C e D de tamanho $\text{poly}(m + n)$ com m bits de entrada e n bits de saída, os quais codificam, respectivamente, distribuições de probabilidade X e Y sobre todas as strings com n bits;

decidir entre: *instância positiva:* $d_{VT}(X, Y) = 0$;
instância negativa: $d_{VT}(X, Y) \geq 1/2$;

prometido que vale um dos casos.

Um problema difícil para PZK é o problema *Distribuições Idênticas (DI)* (Malka, 2015), definido a seguir. Não se sabe se DI está em PZK , isto é, se é PZK -completo. Todos os problemas completos para PZK que conhecemos são artificialmente construídos.

DI

Dados: dois inteiros não-negativos m e n , e circuitos booleanos C e D de tamanho $\text{poly}(m + n)$ com m bits de entrada e n bits de saída, os quais codificam, respectivamente, distribuições de probabilidade X e Y sobre todas as strings com n bits, mais um circuito booleano D de tamanho polinomial em n com um bit de saída que codifica uma distribuição de probabilidade Z sobre um bit;

decidir entre: *instância positiva:* $d_{VT}(X, Y) = 0$ e $\Pr[Z = 1] \geq 2/3$;
instância negativa: $d_{VT}(X, Y) \geq 1/2$ ou $\Pr[Z = 1] \leq 1/3$;

prometido que vale um dos casos.

Um problema completo para NISZK é o problema *Distância Estatística para Distribuição Uniforme (SDU)*, de *Statistical Difference from Uniform* (Goldreich et al., 1999).

SDU

Dados: dois inteiros não-negativos m e n e um circuito booleano C de tamanho $\text{poly}(m + n)$ com m bits de entrada e n bits de saída, o qual codifica uma distribuição de probabilidade X sobre todas as strings com n bits;

decidir entre: *instância positiva:* $d_{VT}(X, U_n) < 1/n$;
instância negativa: $d_{VT}(X, U_n) > 1 - 1/n$;

prometido que vale um dos casos.

Segue o lema de polarização para SDU . Definimos *suporte* de X como

$$\text{suporte}(X) = \{x \in \{0, 1\}^n : \Pr[X = x] \neq 0\}.$$

Lema 3.3 (Lema da Polarização para SDU (Vadhan, 1999)). *Sejam $\alpha, \beta : \mathbb{N} \rightarrow \mathbb{N}$ duas funções tais que $\alpha(n)$ e $\beta(n)$ são computáveis em tempo $\text{poly}(n)$ e, para alguma constante c ,*

$$n\alpha(n) \leq \log \left(\frac{1}{1 - \beta(n)} \right) + \frac{1}{n^c}.$$

Seja C um circuito que codifica a distribuição de probabilidade X . Existe uma função computável em tempo polinomial que, dado $(C, 1^k)$, devolve um circuito D codificando uma distribuição de probabilidade Y , de modo que

- *se $d_{VT}(X, U_n) \leq \alpha(n)$ então $d_{VT}(Y, U_{n'}) \leq 2^{-k}$;*
- *se $d_{VT}(X, U_n) \geq \beta(n)$ então $d_{VT}(Y, U_{n'}) \geq 1 - 2^{-k}$.*

No segundo caso, o suporte de Y é no máximo uma fração 2^{-k} de $\{0, 1\}^{n'}$.

Um problema completo para NIPZK é o problema *Distribuição Uniforme (DU)* (Malka, 2015). Na definição do problema, a saída do circuito C tem $n + 1$ bits, $X[1..n]$ representa a distribuição sobre todas as strings com os n primeiros bits, e $X[n + 1]$ a distribuição sobre o último bit. Ainda, o conjunto $\{0, 1\}^n 1$ representa o conjunto das strings com $n + 1$ bits que terminam em 1.

DU

Dados: dois inteiros não-negativos m e n e um circuito booleano C de tamanho $\text{poly}(m + n)$ com m bits de entrada e $n + 1$ bits de saída, o qual codifica uma distribuição de probabilidade X sobre todas as strings com $n + 1$ bits;

decidir entre: *instância positiva:* $d_{VT}(X[1..n], U_n) = 0$ e $\Pr[X[n + 1] = 1] \geq 2/3$;
instância negativa: $|\text{suporte}(X) \cap \{0, 1\}^n 1| \leq 2^n/3$;

prometido que vale um dos casos.

Um problema completo para QSZK é o problema *Proximidade entre Estados Quânticos* (QSC, de *Quantum State Closeness*) (Watrous, 2002). Sendo α e β dois números reais tais que $0 \leq \alpha < \beta^2 \leq 1$, esse problema é definido como:

(α, β) -QSC

Dados: dois inteiros não-negativos m e n tais que $m \geq n$ e circuitos quânticos Q_0 e Q_1 de tamanho $\text{poly}(m)$ com m qubits de entrada e n qubits de saída;

sendo: ρ_0 a saída de Q_0 ao receber $|0\rangle^{\otimes m}$ e ρ_1 a saída de Q_1 ao receber $|0\rangle^{\otimes m}$;

decidir entre: *instância positiva:* $\|\rho_0 - \rho_1\|_{\text{tr}} \leq \alpha$;
instância negativa: $\|\rho_0 - \rho_1\|_{\text{tr}} \geq \beta$;

prometido que vale um dos casos.

Como $\text{QSZK} = \text{coQSZK}$, o complemento de QSC, conhecido como o problema de distinção de estados quânticos (*Quantum State Distinguishability* — QSD), também é um problema completo para QSZK. Se ρ_0 e ρ_1 forem operadores de densidade diagonalizados, obtemos um problema equivalente ao problema SD. De modo análogo ao caso clássico, temos o Lema da Polarização para o caso quântico.

Lema 3.4 (Watrous, 2002). *Considere que α e β satisfazem $0 \leq \alpha \leq \beta^2 \leq 1$. Sejam Q_0 e Q_1 descrições de circuitos quânticos tendo como saída os estados mistos ρ_0 e ρ_1 , respectivamente. Existe, então, um procedimento determinístico de tempo polinomial que, com a entrada $(Q_0, Q_1, 1^n)$, tem como saída (R_0, R_1) , onde R_0 e R_1 são descrições de circuitos quânticos, tendo cada um tamanho polinomial em n e no tamanho de Q_0 e Q_1 , de modo que os estados mistos ξ_0 e ξ_1 obtidos na saída de R_0 e R_1 satisfazem o seguinte:*

- se $\frac{1}{2}\|\rho_0 - \rho_1\|_1 \leq \alpha$ então $\frac{1}{2}\|\xi_0 - \xi_1\|_1 \leq 2^{-n}$;
- se $\frac{1}{2}\|\rho_0 - \rho_1\|_1 \geq \beta$ então $\frac{1}{2}\|\xi_0 - \xi_1\|_1 \geq 1 - 2^{-n}$.

Assim, podemos restringir o problema (α, β) -QSC a $(1 - 2^{-\text{poly}(m)}, 2^{-\text{poly}(m)})$ -QSC, chamando esse problema apenas de QSC.

Se restringirmos o problema (α, β) -QSC de modo que os circuitos tenham apenas um qubit, esse problema está em BQP. Sendo α e β dois números reais tais que $0 \leq \alpha < \beta^2 \leq 1$, esse problema é definido como:

(α, β) -1QSC

- Dados:* um inteiro não-negativo m e circuitos quânticos Q_0 e Q_1 de tamanho $\text{poly}(m)$ com m qubits de entrada e um qubit de saída;
- sendo:* ρ_0 a saída de Q_0 ao receber $|0\rangle^{\otimes m}$ e ρ_1 a saída de Q_1 ao receber $|0\rangle^{\otimes m}$;
- decidir entre:* *instância positiva:* $\|\rho_0 - \rho_1\|_{\text{tr}} \leq \alpha$;
- instância negativa:* $\|\rho_0 - \rho_1\|_{\text{tr}} \geq \beta$;

prometido que vale um dos casos.

Proposição 3.5. *Para $0 \leq \alpha < \beta \leq 1$ com $1/(\beta - \alpha)^2 = \text{poly}(m)$, o problema (α, β) -1QSC está em BQP.*

Demonstração. Segue do Teorema 2.3. □

Um problema completo para QPZK₁ (QPZK com abrangência perfeita) é o problema *Igualdade de Estados Quânticos* (QSI, de *Quantum State Identicalness*) (Yan, 2012).

QSI

- Dados:* dois inteiros não-negativos m e n tais que $m \geq n$ e dois circuitos quânticos Q_0 e Q_1 de tamanho $\text{poly}(m)$ com m qubits de entrada e n de saída;
- sendo:* ρ_0 a saída de Q_0 ao receber $|0\rangle^{\otimes m}$ e ρ_1 a saída de Q_1 ao receber $|0\rangle^{\otimes m}$;
- decidir entre:* *instância positiva:* $\|\rho_0 - \rho_1\|_{\text{tr}} = 0$;
- instância negativa:* $\|\rho_0 - \rho_1\|_{\text{tr}} \geq 2/3$;

prometido que vale um dos casos.

Um problema completo para QPZK é o problema *QSI'* (Yan, 2012).

QSI'

Dados: dois inteiros não-negativos m e n tais que $m \geq n$ e dois circuitos quânticos Q_0 e Q_1 de tamanho $\text{poly}(m)$ com n qubits de saída, mais um circuito quântico Q_2 de tamanho polinomial em m , com k qubits de entrada e um qubit de saída;

sendo: ρ_0 a saída de Q_0 ao receber $|0\rangle^{\otimes m}$, ρ_1 a saída de Q_1 ao receber $|0\rangle^{\otimes m}$ e b o resultado da medição da saída de Q_2 ao receber $|0\rangle^{\otimes k}$;

decidir entre: *instância positiva:* $\|\rho_0 - \rho_1\|_{\text{tr}} = 0$ e $\Pr[b = 1] \geq 2/3$;
instância negativa: $\|\rho_0 - \rho_1\|_{\text{tr}} \geq 2/3$ ou $\Pr[b = 1] \leq 1/3$;

prometido que vale um dos casos.

Conjectura-se que um problema completo para NIQSZK é o problema *Proximidade de um Estado Quântico ao Estado Maximalmente Misto* (QSCMM, de *Quantum State Closeness to Maximally Mixed*) (Kobayashi, 2003). Sendo α e β dois números reais tais que $0 \leq \alpha < \beta \leq 1$, esse problema é definido como:

(α, β) -QSCMM

Dados: dois inteiros não-negativos m e n tais que $m \geq n$ e um circuito quântico Q de tamanho $\text{poly}(m)$ com m qubits de entrada e n qubits de saída;

sendo: ρ a saída de Q ao receber $|0\rangle^{\otimes m}$;

decidir entre: *instância positiva:* $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} \leq \alpha$;
instância negativa: $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} \geq \beta$;

prometido que vale um dos casos.

Se ρ for um operador de densidade diagonalizado obtemos um problema equivalente ao problema SDU.

Um problema completo para NIQPZK₁ (NIQPZK com abrangência perfeita) é o problema *Igualdade de um Estado Quântico com o Estado Maximalmente Misto* (QSIMM, de *Quantum State Identicalness to Maximally Mixed*) (Kobayashi, 2003).

QSIMM

Dados: dois inteiros não-negativos m e n tais que $m \geq n$ e um circuito quântico Q de tamanho $\text{poly}(m)$ com m qubits de entrada e n qubits de saída;

sendo: ρ a saída de Q ao receber $|0\rangle^{\otimes m}$;

decidir entre: *instância positiva:* $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} = 0$;
instância negativa: $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} \geq 2/3$;

prometido que vale um dos casos.

Um problema completo para NIQPZK é o problema *QSIMM'* (Yan, 2012).

QSIMM'

Dados: dois inteiros não-negativos m e n tais que $m \geq n$ um circuito quântico Q_0 de tamanho $\text{poly}(m)$ com n qubits de saída, mais um circuito quântico Q_1 de tamanho polinomial em m , com k qubits de entrada e um qubit de saída;

sendo: ρ a saída de Q_0 ao receber $|0\rangle^{\otimes m}$ e b o resultado da medição da saída de Q_1 ao receber $|0\rangle^{\otimes k}$;

decidir entre: *instância positiva:* $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} = 0$ e $\Pr[b = 1] \geq 2/3$;
instância negativa: $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} \geq 1/2$ ou $\Pr[b = 1] \leq 1/3$;

prometido que vale um dos casos.

3.2 COMPUTAÇÃO QUÂNTICA SEM COLAPSO

Nesta seção, as definições e resultados referentes à computação quântica sem colapso baseiam-se no texto de Aaronson et al. (2016).

Diferentemente de como definimos circuitos quânticos na Seção 2.3, na presente seção consideraremos circuitos quânticos em que as medições ocorrem em estágios intermediários, pois queremos explorar o poder da computação quântica sem colapso nas medições. Consideremos um circuito quântico C de n qubits, cuja construção depende de uma string x com n bits, definido pela seguinte sequência de operadores $C = (U_1, M_1, U_2, M_2, \dots, U_T, M_T)$, onde cada U_t é um operador unitário de n qubits, e cada M_t é uma medição quântica na base computacional de m_t qubits, sendo $0 \leq m_t \leq n$. O estado inicial do circuito é $|\psi_0\rangle = |0\rangle^{\otimes n}$, e após a t -ésima medição é $|\psi_t\rangle = M_t U_t |\psi_{t-1}\rangle$, de forma que os estados nas diferentes etapas do circuito são dados pela sequência de variáveis aleatórias $\{|\psi_t\rangle\}_{t=0}^R$, sujeitas a uma distribuição de probabilidade que depende do circuito C . Consideremos um procedimento que recebe como entrada o circuito C e amostra a sequência $\{|\psi_t\rangle\}_{t=1}^R$ seguindo a distribuição de probabilidade conforme definida acima. Então, o procedimento mede os estados $|\psi_t\rangle$ para cada t de forma independente na base computacional. São retornados os R resultados dessas medições, nomeados de $v_1, \dots, v_R$. Supondo que esse procedimento seja feito em apenas um passo, chamamos esse procedimento de *oráculo* Q . Observe que se fosse possível ocorrer medições sem colapso, o resultado do oráculo seria equivalente ao caso em que para cada t , os m_t qubits medidos em M_t colapsam, e os demais $n - m_t$ qubits são medidos sem colapso. Lembre-se que é possível que as medições $M_1, M_2, \dots, M_R$ sejam de menos que n (ou ainda zero) qubits (se $m_t = 0$, então a medição M_t é totalmente sem colapso). Observe que o oráculo amostra todos os estados $\{|\psi_t\rangle\}$ de uma só vez, antes de realizar as medições na base computacional, resultando em um modelo *não-adaptativo* de medições sem colapso.

Podemos agora definir **naCQP** (*non-adaptive Collapse-free Quantum Polynomial*) como a classe dos problemas de promessa que podem ser resolvidos por um *algoritmo* **naCQP**, i.e. uma máquina de Turing polinomial com probabilidade de erro menor ou igual a $1/3$ que, tendo como entrada uma string x de n bits, escreve a descrição de um circuito quântico C e faz uma única consulta ao oráculo Q como definido acima, veja Figura 3.1. De modo semelhante, definimos um *algoritmo* **naCQTime**($T(n)$) como uma máquina de Turing limitada por uma função $T(n)$ com probabilidade de erro menor ou igual a $1/3$ que, tendo como entrada uma string x de n bits, escreve a descrição de um circuito quântico C e faz uma única consulta ao oráculo Q . A classe **naCQTime**($T(n)$) é definida de modo análogo.

Podemos também definir **CQP**, uma classe mais geral, em que a máquina base pode fazer de maneira *adaptativa* mais de uma consulta ao oráculo, isto é, ao construir o circuito $C = (U_1, M_1, U_2, M_2, \dots, U_T, M_T)$ e receber as amostras $v_1, \dots, v_t$, usa essas amostras para

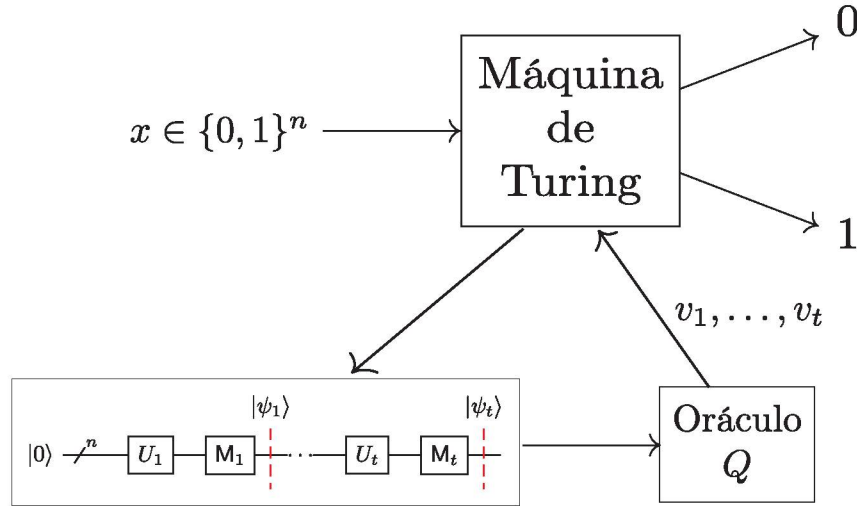


Figura 3.1: Descrição de um algoritmo naCQP: uma máquina de Turing polinomial com saída 0 ou 1 (com probabilidade de erro menor ou igual a $1/3$) que, tendo como entrada uma string x de n bits, escreve a descrição de um circuito quântico $C = (U_1, M_1, U_2, M_2, \dots, U_t, M_t)$, faz uma única consulta sobre C ao oráculo Q e recebe do oráculo as amostras $v_1, \dots, v_t$.

construir o circuito $(U_{t+1}, M_{t+1}, \dots, U_{t'}, M_{t'})$ para a próxima consulta. A classe CQP é definida como a classe dos problemas de promessa que podem ser resolvidos no *modelo* CQP, i.e., em tempo polinomial por uma máquina de Turing com consultas adaptativas a Q com probabilidade de erro menor ou igual a $1/3$.

Para simplificar nossas provas, mostramos um circuito equivalente para naCQP usando um procedimento no qual as medições são postergadas. Esse procedimento também é descrito por Aaronson et al. (2016) (Apêndice C). Como explicado anteriormente, dado um circuito C , os estados nos diferentes estágios do circuito são dados pela sequência $\{|\psi_t\rangle\}_{t=0}^R$. Em seguida, o oráculo Q retorna medições da sequência $\{|\psi_t\rangle\}_{t=1}^R$ denominada $v_1, \dots, v_R$. Note que $v_1, \dots, v_R$ é uma sequência de variáveis aleatórias governada por uma distribuição de Markov. Isto é, para todo $1 \leq t \leq R$, temos que v_t é independente de $v_0, \dots, v_{t-2}$ condicionado a um valor particular de v_{t-1} . Em nosso caso, a variável v_t depende do valor dos qubits de $|\psi_{t-1}\rangle$ que foram medidos por $M_1, \dots, M_{t-1}$ e, portanto, colapsaram. Agora, suponha que o oráculo Q não amostra todos os estados $\{|\psi_t\rangle\}$ no mesmo momento, mas, no lugar, é feito o seguinte procedimento. O oráculo Q amostra cada $|\psi_t\rangle$ de modo sequencial. Em cada amostragem, o oráculo constrói um circuito equivalente em que todas as medições $M_1, \dots, M_{t-1}$ são adiadas de acordo com o Princípio da Medição Postergada (PMP), e uma medição equivalente M'_t ocorre imediatamente antes do estado $|\psi_t\rangle$. Contudo, como a construção desse circuito equivalente é condicionado ao valor de v_{t-1} na amostragem anterior, M'_t depende não somente das medições postergadas de $M_1, \dots, M_{t-1}$, mas também dos resultados em que os qubits colapsaram, com os quais a atual amostragem deve ser consistente. Desse modo, M'_t é construído de tal modo que as medições que ocorreriam até o tempo $t - 1$ colapsam agora para os mesmos valores que na última amostragem. No decorrer desse texto, supomos sem perda de generalidades que cada estado é amostrado conforme o procedimento acima, com todas as medições postergadas, condicionadas ao valor da amostragem anterior. Também, para um t fixo, usamos $|\phi_t\rangle$ para denotar o estado imediatamente antes da única medição M'_t feita por Q para obter $|\psi_t\rangle$, e $|\phi_0\rangle, \dots, |\phi_{t-1}\rangle$ são usados para denotar os estados prévios, sem ter sido feitas medições. Além disso, apesar das portas $U_1, \dots, U_t$ serem modificadas pelo uso de PMP, por abuso mantemos os nomes $U_1, \dots, U_t$, supondo que as portas foram modificadas de modo que as medições sejam postergadas (veja a Figura 3.2).

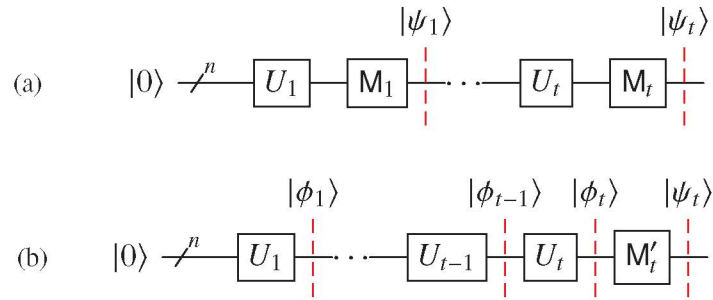


Figura 3.2: (a) O circuito naCQP original usado por $\mathcal{Q}$ para a amostragem de todos $|\psi_1\rangle, \dots, |\psi_t\rangle$. (b) O circuito construído por $\mathcal{Q}$ para amostrar apenas $|\psi_t\rangle$ com as amostragens postergadas.

Sendo A uma linguagem, um *algoritmo naCQP com consulta a um oráculo A* ¹ é um algoritmo naCQP que ao receber a entrada x de tamanho n pode consultar uma função $f : \{0, 1\}^n \rightarrow \{0, 1\}$ tal que $f(x) = 1$ se e somente se $x \in A$. Isto é, para a construção dos operadores unitários $U_1, \dots, U_R$ podemos usar a transformação unitária de n -qubits definida como

$$U_f : |x, b\rangle \rightarrow |x, b \oplus f(x)\rangle,$$

para quaisquer $x \in \{0, 1\}^n$ e $b \in \{0, 1\}$, onde a operação $\oplus$ indica a adição módulo 2, ou, de modo equivalente,

$$U_f |x\rangle = (-1)^{f(x)} |x\rangle, \quad \text{para } x \in \{0, 1\}^n.$$

Durante esse texto, assumimos sem perda de generalidade que cada uma das transformações unitárias $U_1, \dots, U_R$, de um algoritmo naCQP com consulta a um oráculo f , é ou uma cópia de U_f , ou um circuito de n qubits construído usando apenas portas de um conjunto universal fixo. Assim como estabelecemos, na Definição 2.8, a notação $q_y(|\phi_t\rangle)$ no contexto de um algoritmo BQP, estendemos de modo análogo essa notação para um algoritmo naCQP.

Claramente a classe naCQP contém BQP, pois podemos fornecer ao oráculo um circuito quântico polinomial e usar apenas o resultado da medição no final do circuito. Em contrapartida, como dito por Aaronson et al. (2016), “o oráculo $\mathcal{Q}$ nos fornece informação sobre os estágios intermediários da computação quântica sem colapsar o estado; é isso que fornece a naCQP um poder adicional sobre BQP”. Isso proporciona, inclusive, que algoritmos quânticos nesse modelo sejam capazes de resolver problemas SZK-completos (não conhecemos se $\text{SZK} \subseteq \text{BQP}$). Pois podemos resolver o problema de distância estatística SD, completo para SZK, no modelo naCQP.

Teorema 3.6 (Aaronson et al., 2016). $\text{SD} \in \text{naCQP}$.

Demonstração. Seja (m, n, C_0, C_1) uma instância de $\text{SD} = (\epsilon, 1 - \epsilon)\text{-SD}$, com $\epsilon = 2^{-\text{poly}(n)}$. Sejam Y_0 e Y_1 as distribuições de probabilidade correspondentes a C_0 e C_1 , respectivamente.

Primeiro, preparamos o estado

$$|\psi\rangle = \frac{1}{2^{(n+1)/2}} \sum_{b \in \{0,1\}, x \in \{0,1\}^n} |b\rangle |x\rangle |C_b(x)\rangle.$$

¹Não confundir esse oráculo A com o também assim chamado *oráculo $\mathcal{Q}$* na definição de um algoritmo naCQP. Apesar do uso do mesmo termo *oráculo* em ambas as situações, ainda o fazemos para seguir o padrão da literatura.

Nós nos referimos ao qubit de $|b\rangle$ como sendo o primeiro registrador, aos qubits de $|x\rangle$ como sendo o segundo registrador e aos qubits de $|C_b(x)\rangle$ como sendo o terceiro registrador. Rescrevendo ψ da seguinte forma

$$|\psi\rangle = \frac{1}{2^{(n+1)/2}} \sum_{x \in \{0,1\}^n} (|0\rangle |x\rangle |C_0(x)\rangle + |1\rangle |x\rangle |C_1(x)\rangle)$$

e fazendo uma medição, com colapso, do terceiro registrador, este colapsa para $|y\rangle$, onde $y \in \{0, 1\}^m$, com probabilidade

$$\frac{1}{2}(\Pr[Y_0 = y] + \Pr[Y_1 = y]).$$

Medimos sem colapso, então, o primeiro registrador por três vezes, obtendo o bit b_i na i -ésima medição, $1 \leq i \leq 3$. Respondemos que a instância é negativa (i.e. $d_{VT}(Y_0, Y_1) \geq 1 - \epsilon$) se e somente se $b_1 = b_2 = b_3$. Vamos mostrar que a probabilidade de erro é no máximo:

- $\frac{1}{4} + \epsilon$ se $d_{VT}(Y_0, Y_1) \leq \epsilon$;
- 6ϵ se $d_{VT}(Y_0, Y_1) \geq 1 - \epsilon$.

Calculamos a probabilidade para o caso em que $d_{VT}(Y_0, Y_1) \leq \epsilon$. Neste caso o algoritmo erra quando ocorre $b_1 = b_2 = b_3$. A probabilidade disso ocorrer, dado que houve colapso em y , é

$$\left(\frac{\Pr[Y_0 = y]}{\Pr[Y_0 = y] + \Pr[Y_1 = y]} \right)^3 + \left(\frac{\Pr[Y_1 = y]}{\Pr[Y_0 = y] + \Pr[Y_1 = y]} \right)^3.$$

Sendo $\delta(y) = \Pr[Y_1 = y] - \Pr[Y_0 = y]$, temos $\sum_y \delta(y) = 0$ e $\sum_y |\delta(y)| \leq 2\epsilon$. A probabilidade total de um algoritmo errar nesse caso, então, é

$$\begin{aligned} \Pr[\text{erro}] &= \sum_y \Pr[Y_0 = y \text{ ou } Y_1 = y] \cdot \Pr[b_1 = b_2 = b_3 \mid Y_0 = y \text{ ou } Y_1 = y] \\ &= \sum_y \left(\frac{\Pr[Y_0 = y] + \Pr[Y_1 = y]}{2} \right) \frac{\Pr[Y_0 = y]^3 + \Pr[Y_1 = y]^3}{[\Pr[Y_0 = y] + \Pr[Y_1 = y]]^3} \\ &= \frac{1}{2} \sum_y \frac{\Pr[Y_0 = y]^3 + \Pr[Y_1 = y]^3}{[\Pr[Y_0 = y] + \Pr[Y_1 = y]]^2} \\ &= \frac{1}{2} \sum_y \frac{\Pr[Y_0 = y]^3 + [\Pr[Y_0 = y] + \delta(y)]^3}{[2 \Pr[Y_0 = y] + \delta(y)]^2} \\ &= \frac{1}{2} \sum_y \frac{2 \Pr[Y_0 = y]^3 + 3 \Pr[Y_0 = y] \delta(y)^2 + \delta(y)^3}{4 \Pr[Y_0 = y]^2 + 3 \Pr[Y_0 = y] \delta(y) + 4 \Pr[Y_0 = y] \delta(y) + \delta(y)^2} \\ &= \frac{1}{2} \sum_y \left(\frac{\Pr[Y_0 = y]}{2} + \frac{\Pr[Y_0 = y]^2 \delta(y) + \frac{5}{2} \Pr[Y_0 = y] \delta(y)^2 + \delta(y)^3}{4 \Pr[Y_0 = y]^2 + 4 \Pr[Y_0 = y] \delta(y) + \delta(y)^2} \right) \\ &= \frac{1}{4} + \frac{1}{2} \sum_y \delta(y) \frac{\Pr[Y_0 = y]^2 + \frac{5}{2} \Pr[Y_0 = y] \delta(y) + \delta(y)^2}{4 \Pr[Y_0 = y]^2 + 4 \Pr[Y_0 = y] \delta(y) + \delta(y)^2} \\ &\leq \frac{1}{4} + \frac{1}{2} \sum_y |\delta(y)| \frac{\Pr[Y_0 = y]^2 + \frac{5}{2} \Pr[Y_0 = y] |\delta(y)| + |\delta(y)|^2}{4 \Pr[Y_0 = y]^2 + 4 \Pr[Y_0 = y] |\delta(y)| + |\delta(y)|^2} \end{aligned}$$

$$\begin{aligned}
&\leq \frac{1}{4} + \frac{1}{2} \sum_y |\delta(y)| \frac{\Pr[Y_0 = y]^2 + \frac{5}{2} \Pr[Y_0 = y] |\delta(y)| + |\delta(y)|^2}{4 \Pr[Y_0 = y]^2 + 4 \Pr[Y_0 = y] |\delta(y)| + |\delta(y)|^2} \\
&\leq \frac{1}{4} + \frac{1}{2} \sum_y |\delta(y)| \frac{4 \Pr[Y_0 = y]^2 + 4 \Pr[Y_0 = y] |\delta(y)| + |\delta(y)|^2}{4 \Pr[Y_0 = y]^2 + 4 \Pr[Y_0 = y] |\delta(y)| + |\delta(y)|^2} \\
&= \frac{1}{4} + \frac{1}{2} \sum_y |\delta(y)| \leq \frac{1}{4} + \epsilon
\end{aligned}$$

Calculamos a probabilidade para o caso em que $d_{VT}(Y_0, Y_1) \geq 1 - \epsilon$. Neste caso o algoritmo erra quando *não* ocorre $b_1 = b_2 = b_3$. A probabilidade disso ocorrer, dado que houve colapso em y , é

$$\frac{3 \Pr[Y_0 = y] \Pr[Y_0 = y] \Pr[Y_1 = y] + 3 \Pr[Y_0 = y] \Pr[Y_1 = y] \Pr[Y_1 = y]}{[\Pr[Y_0 = y] + \Pr[Y_1 = y]]^3}.$$

A probabilidade total de um algoritmo errar nesse caso, então, é

$$\begin{aligned}
\Pr[\text{erro}] &= \sum_y \Pr[X_0 = y \text{ or } X_1 = y] \cdot \Pr[b_1, b_2, b_3 \text{ não são todas iguais} \mid X_0 = y \text{ or } X_1 = y] \\
&= \sum_y \frac{\Pr[Y_0 = y] + \Pr[Y_1 = y]}{2} \cdot \frac{3 \Pr[Y_0 = y]^2 \Pr[Y_1 = y] + 3 \Pr[Y_0 = y] \Pr[Y_1 = y]^2}{[\Pr[Y_0 = y] + \Pr[Y_1 = y]]^3} \\
&= \frac{3}{2} \left(\sum_y \frac{\Pr[Y_0 = y]^2 \Pr[Y_1 = y]}{[\Pr[Y_0 = y] + \Pr[Y_1 = y]]^2} + \sum_y \frac{\Pr[Y_0 = y] \Pr[Y_1 = y]^2}{[\Pr[Y_0 = y] + \Pr[Y_1 = y]]^2} \right). \tag{3.1}
\end{aligned}$$

Encontramos agora um limite superior para a primeira soma da Equação 3.1, e encontramos de modo similar para a segunda. Como $d_{VT}(Y_0, Y_1) \geq 1 - \epsilon$, então existe um conjunto S de valores y , e um complemento $\bar{S}$, tal que $\sum_{y \in S} \Pr[Y_0 = y] \geq 1 - \epsilon$ e $\sum_{y \in S} \Pr[Y_1 = y] \leq \epsilon$, o que implica $\sum_{y \in \bar{S}} \Pr[Y_0 = y] \leq \epsilon$ e $\sum_{y \in \bar{S}} \Pr[Y_1 = y] \geq 1 - \epsilon$. Desse modo temos

$$\begin{aligned}
&\sum_y \frac{\Pr[Y_0 = y]^2 \Pr[Y_1 = y]}{[\Pr[Y_0 = y] + \Pr[Y_1 = y]]^2} \\
&= \sum_{y \in S} \frac{\Pr[Y_0 = y]^2 \Pr[Y_1 = y]}{[\Pr[Y_0 = y] + \Pr[Y_1 = y]]^2} + \sum_{y \in \bar{S}} \frac{\Pr[Y_0 = y]^2 \Pr[Y_1 = y]}{[\Pr[Y_0 = y] + \Pr[Y_1 = y]]^2} \\
&\leq \sum_{y \in S} \frac{\Pr[Y_0 = y]^2 \Pr[Y_1 = y]}{[\Pr[Y_0 = y]]^2} + \sum_{y \in \bar{S}} \frac{\Pr[Y_0 = y]^2 \Pr[Y_1 = y]}{[\Pr[Y_1 = y]]^2} \\
&\leq \sum_{y \in S} \Pr[Y_1 = y] + \sum_{y \in \bar{S}} \frac{\Pr[Y_0 = y]^2}{\Pr[Y_1 = y]} \\
&\leq \sum_{y \in S} \Pr[Y_1 = y] + \sum_{y \in \bar{S}} \Pr[Y_0 = y] \\
&\leq 2\epsilon.
\end{aligned}$$

Aplicando o mesmo raciocínio na segunda soma, encontramos

$$\Pr[\text{erro}] \leq \frac{3}{2}(2\epsilon + 2\epsilon) = 6\epsilon,$$

como queríamos. $\square$

Corolário 3.7. $\text{SZK} \subseteq \text{naCQP}$. $\square$

Façamos dois exemplos, ambos com 3 bits de entrada e saída, o primeiro em que $d_{VT}(Y_0, Y_1) = 1$ e o segundo em que $d_{VT}(Y_0, Y_1) = 0$.

Primeiro exemplo $d_{VT}(Y_0, Y_1) = 1$.

x	$Y_0(x)$	$Y_1(x)$
000	000	011
001	000	011
010	000	011
011	000	011
100	001	100
101	001	100
110	010	101
111	010	101

As probabilidades codificadas são:

$$\begin{array}{ll}
 \Pr[Y_0 = 000] = 1/2 & \Pr[Y_1 = 000] = 0 \\
 \Pr[Y_0 = 001] = 1/4 & \Pr[Y_1 = 001] = 0 \\
 \Pr[Y_0 = 010] = 1/4 & \Pr[Y_1 = 010] = 0 \\
 \Pr[Y_0 = 011] = 0 & \Pr[Y_1 = 011] = 1/2 \\
 \Pr[Y_0 = 100] = 0 & \Pr[Y_1 = 100] = 1/4 \\
 \Pr[Y_0 = 101] = 0 & \Pr[Y_1 = 101] = 1/4 \\
 \Pr[Y_0 = 110] = 0 & \Pr[Y_1 = 110] = 0 \\
 \Pr[Y_0 = 111] = 0 & \Pr[Y_1 = 111] = 0
 \end{array}
 .$$

O estado inicial que preparamos é

$$\begin{aligned}
 & \frac{1}{4}(|0\rangle|000\rangle|000\rangle + |0\rangle|001\rangle|000\rangle + |0\rangle|010\rangle|000\rangle + |0\rangle|011\rangle|000\rangle + |0\rangle|100\rangle|001\rangle + \\
 & |0\rangle|101\rangle|001\rangle + |0\rangle|110\rangle|010\rangle + |0\rangle|111\rangle|010\rangle + |1\rangle|000\rangle|011\rangle + |1\rangle|001\rangle|011\rangle + \\
 & |1\rangle|010\rangle|011\rangle + |1\rangle|011\rangle|011\rangle + |1\rangle|100\rangle|100\rangle + |1\rangle|101\rangle|100\rangle + |1\rangle|110\rangle|101\rangle + \\
 & |1\rangle|111\rangle|101\rangle) .
 \end{aligned}$$

Medimos agora o terceiro registrador, obtemos, para cada y ,

- com probabilidade $\frac{1}{2} \Pr[Y_0 = 000] + \frac{1}{2} \Pr[Y_1 = 000] = \frac{1}{4}$, o estado

$$\frac{1}{2}(|0\rangle|000\rangle|000\rangle + |0\rangle|001\rangle|000\rangle + |0\rangle|010\rangle|000\rangle + |0\rangle|011\rangle|000\rangle) ;$$

- com probabilidade $\frac{1}{2} \Pr[Y_0 = 001] + \frac{1}{2} \Pr[Y_1 = 001] = \frac{1}{8}$, o estado

$$\frac{1}{\sqrt{2}}(|0\rangle |100\rangle |001\rangle + |0\rangle |101\rangle |001\rangle);$$

- com probabilidade $\frac{1}{2} \Pr[Y_0 = 010] + \frac{1}{2} \Pr[Y_1 = 010] = \frac{1}{8}$, o estado

$$\frac{1}{\sqrt{2}}(|0\rangle |110\rangle |010\rangle + |0\rangle |111\rangle |010\rangle);$$

- com probabilidade $\frac{1}{2} \Pr[Y_0 = 011] + \frac{1}{2} \Pr[Y_1 = 011] = \frac{1}{4}$, o estado

$$\frac{1}{2}(|1\rangle |0\rangle |011\rangle + |1\rangle |001\rangle |011\rangle + |1\rangle |010\rangle |011\rangle + |1\rangle |011\rangle |011\rangle);$$

- com probabilidade $\frac{1}{2} \Pr[Y_0 = 4] + \frac{1}{2} \Pr[Y_1 = 4] = \frac{1}{8}$, o estado

$$\frac{1}{\sqrt{2}}(|1\rangle |100\rangle |100\rangle + |1\rangle |101\rangle |100\rangle);$$

- com probabilidade $\frac{1}{2} \Pr[Y_0 = 5] + \frac{1}{2} \Pr[Y_1 = 5] = \frac{1}{8}$, o estado

$$\frac{1}{\sqrt{2}}(|1\rangle |110\rangle |101\rangle + |1\rangle |111\rangle |101\rangle).$$

Agora medimos três vezes sem colapso o primeiro registrador e obtemos:

- com probabilidade $\frac{1}{2}$ os valores $b_1 = b_2 = b_3 = 0$;
- com probabilidade $\frac{1}{2}$ os valores $b_1 = b_2 = b_3 = 1$.

Segundo exemplo $d_{VT}(Y_0, Y_1) = 0$.

x	$Y_0(x)$	$Y_1(x)$
000	000	001
001	000	001
010	000	000
011	000	000
100	001	000
101	001	000
110	010	010
111	010	010

As probabilidades codificadas são:

$$\begin{array}{ll}
\Pr[Y_0 = 000] = 1/2 & \Pr[Y_1 = 000] = 1/2 \\
\Pr[Y_0 = 001] = 1/4 & \Pr[Y_1 = 001] = 1/4 \\
\Pr[Y_0 = 010] = 1/4 & \Pr[Y_1 = 010] = 1/4 \\
\Pr[Y_0 = 011] = 0 & \Pr[Y_1 = 011] = 0 \\
\Pr[Y_0 = 100] = 0 & \Pr[Y_1 = 100] = 0 \\
\Pr[Y_0 = 101] = 0 & \Pr[Y_1 = 101] = 0 \\
\Pr[Y_0 = 110] = 0 & \Pr[Y_1 = 110] = 0 \\
\Pr[Y_0 = 111] = 0 & \Pr[Y_1 = 111] = 0
\end{array}$$

O estado inicial que preparamos é

$$\begin{aligned}
& \frac{1}{4}(|0\rangle|000\rangle|000\rangle + |0\rangle|001\rangle|000\rangle + |0\rangle|010\rangle|000\rangle + |0\rangle|011\rangle|000\rangle + |0\rangle|100\rangle|001\rangle + \\
& |0\rangle|101\rangle|001\rangle + |0\rangle|110\rangle|010\rangle + |0\rangle|111\rangle|010\rangle + |1\rangle|000\rangle|001\rangle + |1\rangle|001\rangle|001\rangle + \\
& |1\rangle|010\rangle|000\rangle + |1\rangle|011\rangle|000\rangle + |1\rangle|100\rangle|000\rangle + |1\rangle|101\rangle|000\rangle + |1\rangle|110\rangle|010\rangle + \\
& |1\rangle|111\rangle|010\rangle).
\end{aligned}$$

Medimos agora o terceiro registrador, obtemos para cada y

- com probabilidade $\frac{1}{2} \Pr[Y_0 = 000] + \frac{1}{2} \Pr[Y_1 = 000] = \frac{1}{2}$, o estado

$$\begin{aligned}
& \frac{1}{\sqrt{8}}(|0\rangle|000\rangle|000\rangle + |0\rangle|001\rangle|000\rangle + |0\rangle|010\rangle|000\rangle + |0\rangle|011\rangle|000\rangle + \\
& |1\rangle|010\rangle|000\rangle + |1\rangle|011\rangle|000\rangle + |1\rangle|100\rangle|000\rangle + |1\rangle|101\rangle|000\rangle);
\end{aligned}$$

- com probabilidade $\frac{1}{2} \Pr[Y_0 = 001] + \frac{1}{2} \Pr[Y_1 = 001] = \frac{1}{4}$, o estado

$$\frac{1}{\sqrt{4}}(|0\rangle|100\rangle|001\rangle + |0\rangle|101\rangle|001\rangle + |1\rangle|000\rangle|001\rangle + |1\rangle|001\rangle|001\rangle);$$

- com probabilidade $\frac{1}{2} \Pr[Y_0 = 010] + \frac{1}{2} \Pr[Y_1 = 010] = \frac{1}{4}$, o estado

$$\frac{1}{\sqrt{4}}(|0\rangle|110\rangle|010\rangle + |0\rangle|111\rangle|010\rangle + |1\rangle|110\rangle|010\rangle + |1\rangle|111\rangle|010\rangle);$$

Agora medimos três vezes sem colapso o primeiro registrador e obtemos

- com probabilidade $\frac{1}{4}$ os valores $b_1 = b_2 = b_3 = 0$ ou $b_1 = b_2 = b_3 = 1$;
- com probabilidade $\frac{3}{4}$ os demais valores.

3.3 SEPARAÇÃO ORACULAR ENTRE $UP \cap \text{coUP}$ E BQP

Apresentamos nesta seção resultados obtidos por Bennett et al. (1997) sobre a separação oracular entre $UP \cap \text{coUP}$ e BQP. Abordamos os demais resultados obtidos por Fortnow e Rogers (1999) e Tamon e Yamakami (2001) no Capítulo 4, juntamente com as demonstrações mais gerais para a separação oracular entre naCQP e $UP \cap \text{coUP}$.

Um resultado importante de Bennett et al. (1997), usado para a demonstração de separações oraculares entre $UP \cap coUP$ e BQP, é o Teorema 3.8. Anunciamos o teorema conforme anunciado por Fortnow e Rogers (1999), também o referimos por BBBV (iniciais dos autores), assim como referido por Fortnow e Rogers. Usaremos posteriormente, no Capítulo 4, uma versão adaptada para naCQP desse teorema para demonstração de nossos resultados.

Teorema 3.8 (BBBV, Bennett et al. (1997)). *Seja M^A um algoritmo BQP com oráculo A e seja $p(n)$ o tempo de execução de M^A . Seja x uma string de n bits dada como entrada para M . Para todo $\varepsilon > 0$ existe um conjunto de strings S com $|S| \leq 4(p(n))^2/\varepsilon^2$ tal que, para qualquer oráculo A' , se A' difere de A apenas em uma string y e $y \notin S$, então*

$$\left| \Pr[M^{A'} \text{ aceita } x] - \Pr[M^A \text{ aceita } x] \right| \leq \varepsilon.$$

Podemos interpretar o enunciado do Teorema 3.8 da seguinte maneira. Considere alterar o oráculo A , que a máquina consulta, numa única palavra y (i.e. se A respondia “sim” para y , A passa a responder “não”, e vice-versa). O teorema nos garante que é polinomialmente pequeno o conjunto das palavras y que poderiam fazer a probabilidade da máquina aceitar a entrada divergir mais de ε .

Para a demonstração do Teorema 3.8 necessitamos do Teorema 3.9.

Teorema 3.9 (Bennett et al., 1997). *Seja M^A um algoritmo BQP com oráculo A como definido na Seção 2.4. Para um t fixo, seja T o número de cópias de U_f entre as portas $U_1, \dots, U_t$, e, para $\varepsilon > 0$, seja $F \subseteq [1, t] \times \Sigma^*$*

1. *para cada $(i, y) \in F$, U_i é uma cópia de U_f ;*
2. *$\sum_{(i,y) \in F} q_y(|\phi_i\rangle) \leq \varepsilon^2/2T$ (veja Definição 2.8).*

Agora suponha que a resposta para cada consulta $(i, y) \in F$ é modificada para um bit fixo arbitrário $a_{i,y}$, sendo estas respostas, portanto, não necessariamente consistentes com o oráculo. Definimos $|\phi'_i\rangle$ do mesmo modo que $|\phi_i\rangle$, mas com respeito ao oráculo A modificado pela condição acima. Então,

$$\| |\phi_i\rangle - |\phi'_i\rangle \| \leq \varepsilon.$$

Demonstração. Seja $|\phi_i\rangle$ o estado quântico de M^A no tempo i dada a entrada x e sejam $U_1, \dots, U_t$ as portas de M^A . Seja A_i um oráculo tal que se $(i, y) \in F$ então $A_i(y) = a_{i,y}$ e se $(i, y) \notin F$ então $A_i(y) = A(y)$. Seja U'_i a porta de M^{A_i} , definimos $|E_i\rangle$ como o erro no i -ésimo passo causado pela troca do oráculo A pelo oráculo A_i , desse modo,

$$|E_i\rangle = U'_{i+1} |\phi_i\rangle - U_{i+1} |\phi_i\rangle \quad (3.2)$$

ou

$$U_{i+1} |\phi_i\rangle = U'_{i+1} |\phi_i\rangle - |E_i\rangle. \quad (3.3)$$

Considerando que $|\phi_t\rangle = U_t |\phi_{t-1}\rangle$, temos, usando (3.3),

$$\begin{aligned}
|\phi_t\rangle &= U'_t |\phi_{t-1}\rangle - |E_{t-1}\rangle \\
&= U'_t (U_{t-1} |\phi_{t-2}\rangle) - |E_{t-1}\rangle \\
&= U'_t (U'_{t-1} |\phi_{t-2}\rangle - |E_{t-2}\rangle) - |E_{t-1}\rangle \\
&= U'_t U'_{t-1} |\phi_{t-2}\rangle - U'_t |E_{t-2}\rangle - |E_{t-1}\rangle \\
&= U'_t U'_{t-1} U_{t-2} |\phi_{t-3}\rangle - U'_t |E_{t-2}\rangle - |E_{t-1}\rangle \\
&= U'_t U'_{t-1} (U'_{t-2} |\phi_{t-3}\rangle - |E_{t-3}\rangle) - U'_t |E_{t-2}\rangle - |E_{t-1}\rangle \\
&= U'_t U'_{t-1} U'_{t-2} |\phi_{t-3}\rangle - U'_t U'_{t-1} |E_{t-3}\rangle - U'_t |E_{t-2}\rangle - |E_{t-1}\rangle \\
&= U'_t U'_{t-1} \cdots U'_1 |\phi_0\rangle - (U'_t U'_{t-1} \cdots U'_1 |E_0\rangle + \cdots + U'_t |E_{t-2}\rangle + |E_{t-1}\rangle) \\
&= |\phi'_t\rangle - (U'_t U'_{t-1} \cdots U'_1 |E_0\rangle + \cdots + U'_t |E_{t-2}\rangle + |E_{t-1}\rangle);
\end{aligned}$$

então

$$\begin{aligned}
\| |\phi_t\rangle - |\phi'_t\rangle \| &= \| U'_t U'_{t-1} \cdots U'_1 |E_0\rangle + \cdots + U'_t |E_{t-2}\rangle + |E_{t-1}\rangle \| \\
&\leq \| U'_t U'_{t-1} \cdots U'_1 |E_0\rangle \| + \cdots + \| U'_t |E_{t-2}\rangle \| + \| |E_{t-1}\rangle \|.
\end{aligned}$$

Como todos operadores U'_i são unitários, $\| U'_{t-1} \cdots U'_i |E_i\rangle \| = \| |E_i\rangle \|$, temos

$$\| |\phi_t\rangle - |\phi'_t\rangle \| \leq \sum_{i=0}^{t-1} \| |E_i\rangle \|. \quad (3.4)$$

De (3.2), usando a desigualdade triangular temos:

$$\begin{aligned}
\| |E_i\rangle \|^2 &\leq \| U'_i |\phi_i\rangle \|^2 + \| U_i |\phi_i\rangle \|^2 \\
&\leq \| |\phi_i\rangle \|^2 + \| |\phi_i\rangle \|^2 = 2 \| |\phi_i\rangle \|^2;
\end{aligned} \quad (3.5)$$

Como $\| |E_i\rangle \|^2$ é o resultado do erro causado pela alteração do resultado da query $(i, y) \in F$ podemos reescrever (3.5) como:

$$\| |E_i\rangle \|^2 \leq 2q_y(\phi_i). \quad (3.6)$$

Desse modo, usando a Condição 2 no enunciado, temos

$$\sum_{i=0}^{t-1} \| |E_i\rangle \|^2 \leq 2 \sum_{(i,y) \in F} q_y(|\phi_i\rangle) \leq \frac{\epsilon^2}{T}. \quad (3.7)$$

Usando (3.4) e (3.7) e a desigualdade de Cauchy-Schwarz² obtemos

$$\| |\phi_t\rangle - |\phi'_t\rangle \|^2 \leq \left\| \sum_{i=0}^{t-1} \| |E_i\rangle \| \right\|^2 \leq T \sum_{(i,y) \in F} \| |E_i\rangle \|^2 \leq T \frac{\epsilon^2}{T} = \epsilon^2. \quad (3.8)$$

Portanto,

$$\| |\phi_t\rangle - |\phi'_t\rangle \| \leq \epsilon.$$

□

Corolário 3.10. *Seja M^A um algoritmo BQP com oráculo A como definido na Seção 2.4. Seja T o número de cópias de U_f entre as portas $U_0, \dots, U_{t-1}$. Seja $M^{A^{(y)}}$ um algoritmo cujo oráculo $A^{(y)}$ difere do oráculo A apenas na palavra y . Considere $|y| = n$ e $N = 2^n$. Então*

$$\sum_{y=0}^{N-1} \|\phi_t\rangle - |\phi_t^{(y)}\rangle\|^2 \leq 4T^2.$$

A demonstração se baseia também nas notas de aula de Vazirani (2004).

Demonstração. De (3.7) e (3.8), temos

$$\|\phi_t\rangle - |\phi_t^{(y)}\rangle\|^2 \leq 2T \sum_{(i,y) \in F} q_y(|\phi_i\rangle);$$

$$\sum_{y=0}^{N-1} \|\phi_t\rangle - |\phi_t^{(y)}\rangle\|^2 \leq 2T \sum_{y=0}^{N-1} \sum_t q_y(|\phi_i\rangle).$$

O valor esperado de $\sum_t q_y(|\phi_i\rangle)$ é

$$E_y \left(\sum_t q_y(|\phi_i\rangle) \right) = \frac{T}{N}.$$

Portanto,

$$\sum_{y=0}^{N-1} \|\phi_t\rangle - |\phi_t^{(y)}\rangle\|^2 \leq 2T \sum_{y=0}^{N-1} \frac{T}{N} \leq 4T^2. \quad \square$$

Corolário 3.11 (Bennett et al., 1997). *Seja M^A um algoritmo BQP com oráculo A como definido na Seção 2.4. Seja T o número de cópias de U_f entre as portas $U_0, \dots, U_{t-1}$. Para todo $\epsilon > 0$, existe um conjunto de strings S com*

$$|S| \leq \frac{2T^2}{\epsilon^2}$$

tal que, para qualquer oráculo A' , se A' difere de A apenas em uma string y e $y \notin S$, então,

$$\|\phi_t\rangle - |\phi_t^{(y)}\rangle\| \leq \epsilon.$$

²A desigualdade de Cauchy-Schwarz diz que, para dois vetores $\mathbf{a}$ e $\mathbf{b}$ de tamanho T ,

$$\left(\sum_t a_t b_t \right)^2 \leq \left(\sum_t a_t^2 \right) \left(\sum_t b_t^2 \right).$$

Se $b_t = 1$ para todo t , então

$$\left(\sum_t a_t \right)^2 \leq T \sum_t a_t^2.$$

Demonstração. Como cada $|\phi_i\rangle$ tem tamanho unitário,

$$\sum_{i=0}^{T-1} \sum_y q_y(|\phi_i\rangle) \leq T.$$

Seja S o conjunto de strings y tal que

$$\sum_{i=0}^{T-1} q_y(|\phi_i\rangle) \geq \frac{\varepsilon^2}{2T}.$$

Como para qualquer y em S temos

$$|S| \sum_{i=1}^T q_y(|\phi_i\rangle) \leq T,$$

podemos concluir que

$$|S| \leq \frac{2T^2}{\varepsilon^2}.$$

Se $y \notin S$ então

$$\sum_{i=0}^{T-1} q_y(|\phi_i\rangle) < \frac{\varepsilon^2}{2T}. \quad (3.9)$$

Portanto, pelo Teorema 3.9, para todo $y \notin S$,

$$\left\| |\phi_t\rangle - |\phi_t\rangle^{(y)} \right\| \leq \varepsilon. \quad \square$$

Demonstração do Teorema 3.8 (BBBV). Segue diretamente do Corolário 3.11. $\square$

Teorema 3.12 (Bennett et al., 1997). *Relativo a um oráculo aleatório A , temos $\text{UP}^A \cap \text{coUP}^A \not\subseteq \text{BQP}^A$ com probabilidade 1.*

Ressaltamos que, na verdade, Bennett et al. provaram esse enunciado para $\text{NP} \cap \text{coNP}$. Contudo, a mesma prova vale para $\text{UP} \cap \text{coUP}$. Considere convenientemente um oráculo A como uma função $f : \Sigma^* \rightarrow \Sigma^*$ *preservadora do tamanho*, o que pode ser obtido ao interpretar a resposta do oráculo para o par (x, i) como o i -ésimo bit de $f(x)$. Definimos um *oráculo de permutação* como uma função preservadora de tamanho que, para cada $n \geq 0$, retorna uma permutação em Σ^n .

A demonstração de Bennett et al. (1997) é referente a um oráculo aleatório de permutação. Devido à lei zero-um de Kolmogorov, um algoritmo com um oráculo aleatório aceita (ou rejeita) uma linguagem com probabilidade 1 ou 0 (Bennett e Gill, 1981; Dasgupta, 2011). Conforme Bennett et al. argumentam, existe um número contável de algoritmos com oráculo e a intersecção de um número contável de eventos com probabilidade 1 ainda tem probabilidade 1, portanto, uma afirmação que é verdadeira com probabilidade 1, para um oráculo aleatório de permutação, também é verdadeira para qualquer outro oráculo aleatório.

Demonstração do Teorema 3.12. Para qualquer oráculo de permutação A , seja

$$L_A = \{y : \text{primeiro bit de } A^{-1}(y) \text{ é } 1\},$$

uma linguagem que claramente está em $(UP \cap coUP)^A$ — o certificado para y , independentemente de $y \in L_A$ ou $y \notin L_A$, é o único x tal que $A(x) = y$, já que A é um oráculo de permutação.

Seja $M^?$ um algoritmo $BQTime(T(n))$ que pode consultar um oráculo. Seja $T(n) = o(2^{n/3})$. Ao fazer uma amostragem aleatória uniforme de um oráculo de permutação A , vamos mostrar que temos com probabilidade 1 que M^A retorna a resposta errada para a entrada 1^n , para algum n suficientemente grande, o que implica que M^A não aceita L_A , o que é suficiente para mostrar que $(UP \cap coUP)^A \not\subseteq BQP^A$. No que segue, descrevemos a prova que M^A retorna a resposta errada para a entrada 1^n .

Podemos amostrar permutações aleatórias sobre $\{0, 1\}^n$ de acordo com o procedimento a seguir. Sejam $x_0, x_1, \dots, x_{T(n)+1}$ strings escolhidas uniformemente de modo aleatório em $\{0, 1\}^n$. Seja π_0 uma permutação amostrada uniformemente entre permutações π satisfazendo $\pi(x_0) = 1^n$. Agora, sendo τ a transposição (x_{i-1}, x_i) , defina $\pi_i = \pi_{i-1} \cdot \tau$, i.e. $\pi_i(x_i) = \pi_{i-1}(x_{i-1})$ e $\pi_i(x_{i-1}) = \pi_{i-1}(x_i)$. Claramente, cada π_i é uma permutação aleatória de $\{0, 1\}^n$. Seja $\{A_i\}$ uma sequência de oráculos de permutação tal que $A_i(y) = \pi_i(y)$ se $y \in \{0, 1\}^n$ e $A_i(y) = A_j(y)$ se $y \notin \{0, 1\}^n$, para todo j (lembre-se de que Σ pode ter outros símbolos além de 0 e 1, portanto, tudo o que precisamos é definir que as respostas de todos os oráculos A_i sejam consistentes sempre que x não for uma string binária).

Afirmção 3.12.1. A probabilidade de 1^n pertencer a exatamente uma das duas linguagens $L_{A_{T(n)}}$ e $L_{A_{T(n)-1}}$ é $1/2$ (portanto, também é $1/2$ a probabilidade de 1^n não pertencer a nenhuma das linguagens ou a ambas).

Demonstração da afirmação. Tanto $A_{T(n)}^{-1}(1^n) = x_{T(n)}$ quanto $A_{T(n)-1}^{-1}(1^n) = x_{T(n)-1}$ foram amostradas uniformemente de forma aleatória, de modo que a probabilidade de que exatamente uma delas tenha o primeiro bit igual a 1 é $1/2$. $\diamond$

Sejam $|\phi_i\rangle$ e $|\phi'_i\rangle$ os estados quânticos no tempo i de $M^{A_{T(n)}}$ e $M^{A_{T(n)-1}}$, respectivamente, dada a entrada 1^n .

Afirmção 3.12.2. $E[||\phi_{T(n)}\rangle - |\phi'_{T(n)}\rangle||] \leq 1/50$.

Demonstração da afirmação. Para encontrarmos o limitante de $E[||\phi_{T(n)}\rangle - |\phi'_{T(n)}\rangle||]$, mostramos que $|\phi_{T(n)}\rangle$ e $|\phi'_{T(n)}\rangle$ estão próximos à superposição $|\psi'_{T(n)}\rangle$. Para definirmos essa superposição rodamos M para a entrada 1^n com um oráculo diferente em cada passo; no passo i , usamos A_i para responder às consultas para o oráculo. Denotamos por $|\psi'_i\rangle$, a superposição resultante no tempo i . Considere o conjunto de pares $S = \{(i, x_j) : j \geq i, 0 \leq i \leq T\}$. É fácil ver que as consultas ao oráculo na computação descrita acima e as consultas de $M_{T(n)}^A$ e $M_{T(n)+1}^A$ diferem apenas no conjunto S . Podemos afirmar que a magnitude esperada da consulta de qualquer par no conjunto é menor que $1/2^n$, pois para $j \geq i$ podemos considerar x_j como tendo sido escolhido de modo aleatório durante o passo j após as consultas ao oráculo terem sido feitas. Seja α a soma das magnitudes das consultas para os pares em S . Então

$$E[\alpha] \leq \frac{|S|}{2^n} = \frac{\binom{T(n)+1}{2}}{2^n} \leq \frac{(T(n))^2}{2^n}$$

para $T(n) \geq 4$. Seja ε uma variável aleatória tal que $\alpha = \varepsilon^2/2T(n)$. Então, pelo Teorema 3.9, $||\phi\rangle - |\phi_{T(n)}\rangle|| \leq \varepsilon$ e $||\phi\rangle - |\phi'_{T(n)}\rangle|| \leq \varepsilon$. Mostramos acima que

$$E[\varepsilon^2/2T(n)] = E[\alpha] \leq \frac{(T(n))^2}{2^n}$$

Mas $E \left[\varepsilon / \sqrt{2T(n)} \right]^2 \leq E \left[\varepsilon^2 / 2T(n) \right]$. Portanto,

$$E[\varepsilon] = \sqrt{2T(n)} E \left[\frac{\varepsilon}{\sqrt{2T(n)}} \right] \leq \sqrt{2T(n) E \left[\frac{\varepsilon^2}{2T(n)} \right]} \leq \sqrt{2T(n) \frac{T(n)^2}{2^n}}.$$

Como $T(n) \leq \frac{2^{n/3}}{100}$, temos

$$E[\varepsilon] \leq \sqrt{2T(n) \frac{T(n)^2}{2^n}} \leq \sqrt{\frac{2}{100^3}} < \frac{1}{100}.$$

Portanto, $E \left[\left| |\phi\rangle - |\phi_{T(n)}\rangle \right| \right] \leq E[\varepsilon] < 1/100$ e $E \left[\left| |\phi\rangle - |\phi'_{T(n)}\rangle \right| \right] \leq E[\varepsilon] < 1/100$.

Segue que $E \left[\left| |\phi\rangle - |\phi'_{T(n)}\rangle \right| \right] < 1/50$.

$$E \left[\left| |\phi\rangle - |\phi_{T(n)}\rangle \right| \right] + E \left[\left| |\phi\rangle - |\phi'_{T(n)}\rangle \right| \right] < 1/100 + 1/100$$

$$E \left[\left| |\phi\rangle - |\phi_{T(n)}\rangle - |\phi\rangle + |\phi'_{T(n)}\rangle \right| \right] \leq E \left[\left| |\phi\rangle - |\phi_{T(n)}\rangle \right| + \left| |\phi\rangle - |\phi'_{T(n)}\rangle \right| \right] < 1/50$$

$$E \left[\left| |\phi\rangle - |\phi'_{T(n)}\rangle \right| \right] < 1/50$$

◇

Pelo limitante de Markov, $\Pr \left[\left| |\phi\rangle - |\phi'_{T(n)}\rangle \right| \leq 2/25 \right] \geq 3/4$. Usando o Teorema 2.4 concluímos que se $\left| |\phi_{T(n)}\rangle - |\phi'_{T(n)}\rangle \right| \leq 2/25$, então a probabilidade de aceitação de $M_{T(n)}^A$ e $M_{T(n)-1}^A$ diferem no máximo em $8/25 < 1/3$, o que significa que ambos $M^{A_{T(n)}}$ e $M^{A_{T(n)-1}}$ ou aceitam, ou rejeitam 1^n com probabilidade maior que $3/4$.

Como mostrado na Afirmação 3.12.1, a probabilidade de 1^n pertencer exatamente a uma das linguagens $L_{A_{T(n)}}$ e $L_{A_{T(n)-1}}$ é $1/2$. Portanto, com probabilidade de pelo menos $3/4 - 1/2 = 1/4$, $M^{A_{T(n)}}$ ou $M^{A_{T(n)-1}}$ dá a resposta errada na entrada 1^n . Como $A_{T(n)}$ e $A_{T(n)-1}$ são escolhidos da mesma distribuição, $M^{A_{T(n)}}$ retorna a resposta errada com probabilidade de pelo menos $1/8$. Isso leva à conclusão de que M decide L_A com probabilidade de 0 para um oráculo uniformemente aleatório de permutação A . □

3.4 SEPARAÇÃO ORACULAR ENTRE naCQP E NP

Apresentamos nessa seção os resultados obtidos por Aaronson et al. (2016) sobre a separação oracular entre NP e naCQP.

Teorema 3.13 (Aaronson et al., 2016). *Seja M^A um algoritmo naCQP com um oráculo A como definido na Seção 3.2. Seja $N = 2^n$, sendo n o tamanho da entrada x . Seja Q o número de consultas feitas ao oráculo A e T o número de medições. Então, qualquer M^A que faz uma busca não estruturada, no modelo caixa preta, de um valor x obedece $Q + T = \Omega(N^{1/4})$, e, portanto, a busca requer tempo $\Omega(N^{1/4})$.*

Demonstração. Seja M^A um algoritmo naCQP com um oráculo A como definido na Seção 3.2. Seja Q o número de consultas feitas ao oráculo A e T o número de medições. Seja $v(x) = (v_0(x), v_1(x), \dots, v_T(x))$ o resultado das amostragens feitas quando o item marcado é x , i.e. existe um $x \in \{0, 1\}^n$ tal que o oráculo A retorna 1, e seja $v = (v_0, \dots, v_T)$ o resultado

das amostragens sem um item marcado. Ambos $v(x)$ e v são variáveis aleatórias. Como o algoritmo pode distinguir entre as distribuições v e $v(x)$ com probabilidade de sucesso $2/3$, temos $d_{TV}(v, v(x)) \geq 1/3$ para todo x . Notamos que v e $v(x)$ são governadas por uma distribuição de Markov. Portanto, pelo Lema 2.5,

$$d_{TV}(v, v(x)) \leq 2 \sum_{i=1}^T d_{TV}((v_{i-1}, v_i), (v_{i-1}(x), v_i(x))) .$$

Limitemos agora o termo

$$d_{x,i} \equiv d_{TV}((v_{i-1}, v_i), (v_{i-1}(x), v_i(x))) .$$

Assuma que as medições com colapso antes de U_i , que devem ser postergadas conforme descrito na Seção 3.2, foram aplicadas aos k primeiros qubits. Sejam $|\phi\rangle$ e $|\phi(x)\rangle$ os estados anteriores às amostragens. Então, decompomos

$$|\phi\rangle = \sum_{s \in \{0,1\}^k} \alpha_s |s\rangle |\phi_s\rangle$$

e

$$|\phi(x)\rangle = \sum_{s \in \{0,1\}^k} \beta_s |s\rangle |\phi_s(x)\rangle .$$

Possíveis valores para (v_{i-1}, v_i) e $(v_{i-1}(x), v_i(x))$ podem ser escritos na forma (st_i, st_2) , onde s é uma string com k bits e t_1, t_2 são strings com $(l-k)$ bits.

Assuma por enquanto que U_i não contém a consulta para f . Então, como U_i não afeta os primeiros k qubits, pode ser decomposta na soma

$$\sum_{s \in \{0,1\}^k} |s\rangle \langle s| \otimes V_s$$

para determinados operadores unitários V_s . A transformação U_i é equivalente a aplicar V_s aos últimos $l-k$ qubits se a medição dos primeiros k qubits for igual a s . Então, a probabilidade de $(v_{i-1}, v_i) = (st_1, st_2)$ é igual a

$$|\alpha_s|^2 |\langle t_1 | \phi_s \rangle|^2 |\langle t_2 | V_s | \phi_s \rangle|^2 ,$$

e a probabilidade de $(v_{i-1}(x), v_i(x)) = (st_1, st_2)$ é igual a

$$|\beta_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s(x) \rangle|^2 ,$$

Portanto, usando a desigualdade triangular

$$\begin{aligned} |a^2 b^2 c^2 - d^2 e^2 f^2| &= |a^2 e^2 f^2 - d^2 e^2 f^2 + a^2 b^2 c^2 - a^2 e^2 c^2 + a^2 e^2 c^2 - a^2 e^2 f^2| \\ &\leq |a^2 e^2 f^2 - d^2 e^2 f^2| + |a^2 b^2 c^2 - a^2 e^2 c^2| + |a^2 e^2 c^2 - a^2 e^2 f^2| \end{aligned}$$

a distância de variação total $d_{x,i}$ é

$$\begin{aligned}
d_{x,i} &= \frac{1}{2} \sum_{s,t_1,t_2} \left| |\alpha_s|^2 |\langle t_1 | \phi_s \rangle|^2 |\langle t_2 | V_s | \phi_s \rangle|^2 - |\beta_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s(x) \rangle|^2 \right| \\
&\leq \frac{1}{2} \sum_{s,t_1,t_2} \left| |\alpha_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s(x) \rangle|^2 - |\beta_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s(x) \rangle|^2 \right| \\
&\quad + \frac{1}{2} \sum_{s,t_1,t_2} \left| |\alpha_s|^2 |\langle t_1 | \phi_s \rangle|^2 |\langle t_2 | V_s | \phi_s \rangle|^2 - |\alpha_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s \rangle|^2 \right| \\
&\quad + \frac{1}{2} \sum_{s,t_1,t_2} \left| |\alpha_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s \rangle|^2 - |\alpha_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s(x) \rangle|^2 \right| \\
&=: \frac{1}{2} (S_1 + S_2 + S_3)
\end{aligned}$$

onde S_i, S_2, S_3 são as três somas escritas acima que variam entre $s \in \{O, 1\}^k$ e $t_1, t_2 \in \{0, 1\}^{l-k}$. Temos agora,

$$\begin{aligned}
S_1 &:= \sum_{s,t_1,t_2} \left| |\alpha_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s(x) \rangle|^2 - |\beta_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s(x) \rangle|^2 \right| \\
&= \sum_{s,t_1,t_2} \left| \left(|\alpha_s|^2 - |\beta_s|^2 \right) \left(|\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s(x) \rangle|^2 \right) \right| \\
&= \sum_s \left| |\alpha_s|^2 - |\beta_s|^2 \right| \sum_{t_1,t_2} |\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s(x) \rangle|^2 \\
&= \sum_s \left| |\alpha_s|^2 - |\beta_s|^2 \right| \\
&\leq \| |\phi\rangle \langle \phi| - |\phi(x)\rangle \langle \phi(x)| \|_{\text{tr}} \\
&\leq 2 \| |\phi(x)\rangle - |\phi\rangle \|_2.
\end{aligned}$$

Adicionalmente,

$$\begin{aligned}
S_2 &:= \sum_{s,t_1,t_2} \left| |\alpha_s|^2 |\langle t_1 | \phi_s \rangle|^2 |\langle t_2 | V_s | \phi_s \rangle|^2 - |\alpha_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s \rangle|^2 \right| \\
&= \sum_{s,t_1,t_2} \left| |\langle t_2 | V_s | \phi_s \rangle|^2 \left(|\alpha_s|^2 |\langle t_1 | \phi_s \rangle|^2 - |\alpha_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 \right) \right| \\
&= \sum_{s,t_1} \left| |\alpha_s|^2 |\langle t_1 | \phi_s \rangle|^2 - |\alpha_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 \right| \\
&\leq \sum_{s,t_1} \left| |\alpha_s|^2 |\langle t_1 | \phi_s \rangle|^2 + (|\alpha_s|^2 - 2|\beta_s|^2) |\langle t_1 | \phi_s(x) \rangle|^2 \right| \\
&\leq \sum_{s,t_1} \left| |\alpha_s|^2 |\langle t_1 | \phi_s \rangle|^2 - |\beta_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 \right| + \sum_{s,t_1} \left| |\alpha_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 - |\beta_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 \right| \\
&= \sum_{s,t_1} \left| |\alpha_s|^2 |\langle t_1 | \phi_s \rangle|^2 - |\beta_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 \right| + \sum_{s,t_1} \left| |\alpha_s|^2 - |\beta_s|^2 \right| \\
&\leq 2 \| |\phi\rangle \langle \phi| - |\phi(x)\rangle \langle \phi(x)| \|_{\text{tr}} \\
&\leq 4 \| |\phi(x)\rangle - |\phi\rangle \|_2.
\end{aligned}$$

Finalmente,

$$\begin{aligned}
S_3 &:= \sum_{s,t_1,t_2} \left| |\alpha_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s \rangle|^2 - |\alpha_s|^2 |\langle t_1 | \phi_s(x) \rangle|^2 |\langle t_2 | V_s | \phi_s(x) \rangle|^2 \right| \\
&= \sum_{s,t_1,t_2} \left| |\alpha_s|^2 \left(|\langle t_2 | V_s | \phi_s \rangle|^2 - |\langle t_2 | V_s | \phi_s(x) \rangle|^2 \right) \right| \\
&\leq 2 \left| \langle \phi | \langle \phi | - |\phi(x)\rangle \langle \phi(x)| \right|_{\text{tr}} \\
&\leq 4 \left| \langle \phi(x) | - |\phi\rangle \right|_2.
\end{aligned}$$

Portanto,

$$d_{x,i} \leq \frac{1}{2} (S_1 + S_2 + S_3) \leq 5 \left| \langle \phi(x) | - |\phi\rangle \right|_2.$$

Por outro lado, se U_i é uma consulta a f , então é aplicada apenas a mudança de sinal para algumas das amplitudes de probabilidade de $|\phi\rangle$ e $|\phi(x)\rangle$. Portanto, o mesmo argumento ainda mostra que $d_{x,i} \leq 5 \left| \langle \phi(x) | - |\phi\rangle \right|_2$.

Pela desigualdade de Cauchy-Schwarz e Corolário 3.10,

$$\begin{aligned}
\frac{1}{N} \sum_{x=0}^{N-1} d_{x,i} &\leq 5 \frac{1}{N} \sum_{x=0}^{N-1} \left| \langle \phi(x) | - |\phi\rangle \right|_2 \\
&\leq 5 \sqrt{\frac{1}{N} \sum_{x=0}^{N-1} \left| \langle \phi(x) | - |\phi\rangle \right|_2^2} \\
&\leq \frac{10Q}{\sqrt{N}}
\end{aligned}$$

para todo i . Portanto, existe algum x tal que

$$d_{TV}(v, v(x)) \leq 2 \sum_{i=1}^T d_{x,i} \leq 2 \sum_{i=1}^T \sum_{x=0}^{N-1} d_{x,i} \leq \frac{20TQ}{\sqrt{N}}.$$

Por outro lado, $d_{VT}(V, v(x)) \geq 1/3$ para todo x , portanto $\frac{20TQ}{\sqrt{N}} \geq \frac{1}{3}$ e o tempo de execução do algoritmo é de pelo menos $T + Q = \Omega(N^{1/4})$. $\square$

Usando o método de diagonalização do Teorema de Baker-Gill-Solovay (Teorema 2.7, Baker et al. (1975)) obtemos o Corolário 3.14.

Corolário 3.14 (Aaronson et al., 2016). *Existe um oráculo A tal que $\text{NP}^A \not\subseteq \text{naCQP}^A$.*

O principal argumento da demonstração do Teorema 3.13 pode ser resumido pelo Lema 3.15.

Lema 3.15. *Seja M^A um algoritmo naCQP com um oráculo A como definido na Seção 3.2. Seja A' um oráculo com respostas arbitrárias para qualquer consulta, sejam v e w os resultados de medições sem colapso para M^A e $M^{A'}$ respectivamente. Seja $d_i = d_{TV}((v_{i-1}, v_i), (w_{i-1}, w_i))$, então $d_i \leq 5 \left| \langle \phi_{i-1} | - |\phi'_{i-1}\rangle \right|$. $\square$*

4 SEPARAÇÃO ORACULAR PARA naCQP

Neste capítulo apresentamos as contribuições da tese no assunto de separação oracular para a classe naCQP. Especificamente, demonstramos os Teoremas 1.1, 1.2 e 1.3 já enunciados na introdução do trabalho (Capítulo 1).

Teorema 1.1 (Hepp et al., 2025). *Relativo a um oráculo aleatório A , temos $UP^A \cap coUP^A \not\subseteq naCQP^A$ com probabilidade 1.*

Teorema 1.2. *Existe um oráculo genérico A tal que $P^A = BQP^A = SZK^A = naCQP^A \neq UP^A \cap coUP^A$.*

Teorema 1.3 (Hepp et al., 2025). *Existe um oráculo A para o qual $P^A = naCQP^A$ e $UP^A \cap coUP^A = EXP^A$.*

Algumas das propriedades para naCQP que usamos ao estender as provas para BQP de Fortnow e Rogers (1999), de Bennett et al. (1997) e de Tamon e Yamakami (2001) para provar nossos resultados são enunciados no Teorema 4.1 abaixo, que é uma versão (para naCQP) do Teorema BBBV para BQP de Bennett et al. (1997), que anunciamos e discutimos brevemente no Capítulo 3 (Teorema 3.8).

Teorema 4.1 (Hepp et al., 2025). *Seja M^A um algoritmo naCQP com oráculo A e seja $p(n)$ o tempo de execução de M^A . Seja x uma string de n bits dada como entrada para M . Para todo $\varepsilon > 0$ existe um conjunto de strings S com $|S| \leq 200(p(n))^4/\varepsilon^2$ tal que, para qualquer oráculo A' , se A' difere de A apenas em uma única string y e $y \notin S$, então*

$$\left| \Pr[M^{A'} \text{ aceita } x] - \Pr[M^A \text{ aceita } x] \right| \leq \varepsilon.$$

Apresentamos na Seção 4.1 a demonstração do Teorema 4.1; na Seção 4.2 a demonstração do Teorema 1.1; na Seção 4.3 a demonstração do Teorema 1.2; e na Seção 4.4 a demonstração do Teorema 1.3.

4.1 PROVA DO TEOREMA BBBV PARA naCQP

Queremos nesta seção demonstrar o Teorema 4.1, mas antes precisamos mostrar alguns resultados prévios.

Lema 4.2 (Hepp et al., 2025; adaptado de Bennett et al., 1997). *Seja M^A um algoritmo naCQP com oráculo A como definido na Seção 3.2. Para um t fixo, seja T o número de cópias de U_f entre as portas $U_1, \dots, U_t$, e, para $\varepsilon > 0$, seja $F \subseteq [1, t] \times \Sigma^*$*

1. *para cada $(i, y) \in F$, U_i é uma cópia de U_f ;*
2. $\sum_{(i,y) \in F} q_y(|\phi_i\rangle) \leq \varepsilon^2/2T$.

Agora suponha que a resposta para cada consulta $(i, y) \in F$ é modificada para um bit fixo arbitrário $a_{i,y}$, sendo estas respostas, portanto, não necessariamente consistentes com o oráculo. Definimos $|\phi'_i\rangle$ do mesmo modo que $|\phi_i\rangle$, mas com respeito ao oráculo A modificado pela condição acima. Então,

$$\| |\phi_i\rangle - |\phi'_i\rangle \| \leq \varepsilon.$$

□

A prova do Lema 4.2 segue por inspeção da prova mostrada no Teorema 3.9, mas adaptando os argumentos para naCQP. Uma observação importante para tal é que assumimos, na definição de $|\phi_i\rangle$ na Seção 3.2, que até o tempo t não ocorreu nenhuma medição com colapso, o que significa que até esse momento o algoritmo naCQP tem o mesmo comportamento que um algoritmo BQP.

Apresentamos agora o Teorema 4.3, do qual seguem o Corolário 4.4 e o Teorema 4.1.

Teorema 4.3 (Hepp et al., 2025). *Seja M^A um algoritmo naCQP com um oráculo A como definido na Seção 3.2. Seja T o número de cópias de U_f entre as portas $U_1, \dots, U_R$, e, para $\varepsilon > 0$, seja $F \subseteq [1, R] \times \Sigma^*$ o conjunto de pares tal que:*

1. *para cada $(i, y) \in F$, U_i é uma cópia de U_f ;*
2. $\sum_{(i,y) \in F} q_y(|\phi_i\rangle) \leq \varepsilon^2/2T$.

Agora suponha que a resposta para cada consulta $(i, y) \in F$ é modificada para um bit fixo arbitrário $a_{i,y}$, sendo estas respostas, portanto, não necessariamente consistentes com o oráculo. Definimos $|\phi'_i\rangle$ do mesmo modo que $|\phi_i\rangle$, mas com respeito ao oráculo A modificado pela condição acima. Então, sendo $v = (v_1, \dots, v_R)$ e $w = (w_1, \dots, w_R)$ as variáveis aleatórias retornadas pela amostragem de $\{|\psi_t\rangle\}$ e $\{|\psi'_t\rangle\}$, respectivamente, a distância da variação total entre v e w é

$$d_{TV}(v, w) \leq 2 \sum_{i=1}^R 5\varepsilon \leq 10R\varepsilon.$$

Demonstração. Primeiro, fixemos um i . Como podemos postergar todas as medições com colapso de modo que ocorram imediatamente antes da amostragem de v_i e w_i , temos, pelo Lema 3.15,

$$d_i = d_{VT}((v_{i-1}, v_i), (w_{i-1}, w_i)) \leq 5\| |\phi_i\rangle - |\phi'_i\rangle \|.$$

Também, pelo Lema 4.2, como $\sum_{(i,y) \in F} q_y(|\phi_i\rangle) \leq \varepsilon^2/2T$, temos

$$\| |\phi_i\rangle - |\phi'_i\rangle \| \leq \varepsilon.$$

Portanto, temos, pelo Lema 2.5,

$$d_{TV}(v, w) \leq 2 \sum_{i=1}^R d_{TV}((v_{i-1}, v_i), (w_{i-1}, w_i)) \leq 2 \sum_{i=1}^R 5\varepsilon = 10R\varepsilon. \quad \square$$

Corolário 4.4 (Hepp et al., 2025). *Seja M^A um algoritmo naCQP com oráculo A como definido na Seção 3.2. Seja T o número de cópias de U_f entre as portas $U_1, \dots, U_R$. Para todo $\varepsilon > 0$, existe um conjunto de strings S com*

$$|S| \leq 200T^2R^2/\varepsilon^2$$

tal que, para qualquer oráculo A' , se A' difere de A apenas em uma string y e $y \notin S$, então, sendo $v = (v_1, \dots, v_R)$ e $w = (w_1, \dots, w_R)$ as variáveis aleatórias retornadas por M^A e $M^{A'}$, temos

$$d_{TV}(v, w) \leq \varepsilon.$$

Demonstração. Primeiro, seguimos a prova do Corolário 3.11 para o resultado análogo referente a BQP. Como cada $|\phi_i\rangle$ tem tamanho unitário,

$$\sum_{i=1}^R \sum_y q_y(|\phi_i\rangle) \leq T.$$

Seja S o conjunto de strings y tal que

$$\sum_{i=1}^R q_y(|\phi_i\rangle) \geq \frac{\varepsilon^2}{2T}.$$

Como para qualquer y em S temos

$$|S| \sum_{i=1}^R q_y(|\phi_i\rangle) \leq T,$$

podemos concluir que

$$|S| \leq \frac{2T^2}{\varepsilon^2}.$$

Se $y \notin S$ então

$$\sum_{i=1}^R q_y(|\phi_i\rangle) < \frac{\varepsilon^2}{2T}. \quad (4.1)$$

Agora, de (4.1), aplicamos o Teorema 4.3, obtendo que para todo $y \notin S$,

$$d_{TV}(v, w) \leq 10R\varepsilon.$$

Considerando $\varepsilon = \frac{\varepsilon'}{10R}$, temos

$$|S| \leq \frac{2T^2}{(\varepsilon'/10R)^2} = \frac{200T^2R^2}{\varepsilon'^2}.$$

Desse modo, para todo $y \notin S$, temos, como desejado,

$$d_{TV}(v, w) \leq \varepsilon'. \quad \square$$

Demonstração do Teorema 4.1. Segue diretamente do Corolário 4.4. $\square$

4.2 SEPARAÇÃO ORACULAR ALEATÓRIA ENTRE $UP \cap \text{coUP}$ E naCQP

Bennett et al. (1997) usaram a versão original do Lema 4.2 para BQP para provar o Teorema 3.12: em relação a um oráculo aleatório A , temos $(UP \cap \text{coUP})^A \not\subseteq \text{BQP}^A$ com probabilidade 1. Estendemos a prova de Bennett et al. usando o Teorema 4.3 em vez do Lema 4.2 para BQP e mostramos que $(UP \cap \text{coUP})^A \not\subseteq \text{naCQP}^A$ com probabilidade de 1.

Demonstração do Teorema 1.1. Para qualquer oráculo de permutação A , seja

$$L_A = \{y : \text{primeiro bit de } A^{-1}(y) \text{ é } 1\},$$

uma linguagem que claramente está em $(UP \cap coUP)^A$. Seja $M^?$ um algoritmo $naCQTime(T(n))$ que pode consultar um oráculo. Seja $T(n) = o(2^{n/5})$. Ao fazermos uma amostragem aleatória uniforme de um oráculo de permutação A , vamos mostrar que temos com probabilidade 1 que M^A retorna a resposta errada para a entrada 1^n , para algum n suficientemente grande, o que implica que M^A não aceita L_A , o que é o suficiente para mostrarmos que $(UP \cap coUP)^A \not\subseteq naCQP^A$. Esta probabilidade 1, quanto à seleção aleatória de um oráculo de *permutação* (i.e. uma função que preserva o tamanho e devolve uma permutação em Σ^*) permanece sendo a mesma quanto à seleção de um oráculo aleatório. No que segue, argumentamos que M^A retorna a resposta errada para a entrada 1^n .

Podemos amostrar permutações aleatórias de $\{0, 1\}^n$ de acordo com o procedimento a seguir. Sejam $x_0, x_1, \dots, x_{T(n)+1}$ strings escolhidas uniformemente de modo aleatório em $\{0, 1\}^n$. Seja π_0 uma permutação amostrada uniformemente entre permutações π satisfazendo $\pi(x_0) = 1^n$. Agora, sendo τ a transposição (x_{i-1}, x_i) , defina $\pi_i = \pi_{i-1} \cdot \tau$, i.e. $\pi_i(x_i) = \pi_{i-1}(x_{i-1})$ e $\pi_i(x_{i-1}) = \pi_{i-1}(x_i)$. Claramente, cada π_i é uma permutação aleatória de $\{0, 1\}^n$. Seja $\{A_i\}$ uma sequência de oráculos de permutação tal que $A_i(y) = \pi_i(y)$ se $y \in \{0, 1\}^n$ e $A_i(y) = A_j(y)$ se $y \notin \{0, 1\}^n$, para todo j .

Para $naCQTime(T(n))$, sejam $\{|\psi_t\rangle\}$ e $\{|\psi'_t\rangle\}$ os estados amostrados por $M^{A_{T(n)}}$ e $M^{A_{T(n)-1}}$, respectivamente, dada a entrada 1^n , sendo $v = (v_1, \dots, v_R)$ e $w = (w_1, \dots, w_R)$ as variáveis aleatórias correspondentes, como no Teorema 4.3.

Afirmção 1.1.1. $d_{VT}(v, w) \leq 1/50$

Demonstração da afirmação. Considere que o algoritmo $naCQTime(T(n))$ $M^?$ consulta um oráculo A_t diferente para cada $t = 1, \dots, R$, sendo $\{|\bar{\psi}_t\rangle\}$ os estados correspondentes e $q = (q_1, \dots, q_R)$ as variáveis aleatórias correspondentes. Considere o conjunto de pares $S = \{(i, x_j) : j \geq i, 0 \leq i \leq T(n)\}$. Por construção, as consultas dos oráculos descritos acima e de $M_{T(n)}^A$ e $M_{T(n)+1}^A$ diferem apenas no conjunto S . Como cada x_j , para $j \geq i$, pode ser visto como tendo sido escolhido de modo aleatório no passo j , após a superposição das consultas do oráculo ter sido escrita na resposta do oráculo, a magnitude esperada da consulta de cada par em S é no máximo $1/2^n$. Portanto, sendo α a soma das magnitudes das consultas após o passo $T(n) \geq 4$ para os elementos de S ,

$$E[\alpha] \leq \frac{|S|}{2^n} = \frac{\binom{T(n)+1}{2}}{2^n} \leq \frac{(T(n))^2}{2^n}.$$

Seja ε uma variável aleatória tal que $\alpha = \frac{\varepsilon^2}{200R^2T(n)}$. Então pelo Teorema 4.3, e ao fazer uma substituição de variáveis $\varepsilon = \frac{\varepsilon'}{10R}$, temos $d_{VT}(q, v) \leq \varepsilon$ e $d_{VT}(q, w) \leq \varepsilon$. Mostramos acima que $E[\varepsilon^2/200R^2T(n)] = E[\alpha] \leq (T(n))^2/2^n$. Mas $E\left[\varepsilon/R\sqrt{200T(n)}\right]^2 \leq E[\varepsilon^2/200R^2T(n)]$. Portanto,

$$\begin{aligned} E[\varepsilon] &= \sqrt{200R^2T(n)} E\left[\frac{\varepsilon}{\sqrt{200R^2T(n)}}\right] \\ &\leq \sqrt{200R^2T(n)} E\left[\frac{\varepsilon^2}{200R^2T(n)}\right] \leq \sqrt{200R^2T(n)} \frac{(T(n))^2}{2^n}. \end{aligned}$$

Como $T(n) \leq \frac{2^{n/5}}{200}$ e $R \leq \frac{2^{n/5}}{200}$, temos

$$E[\varepsilon] \leq \sqrt{200R^2T(n)} \frac{(T(n))^2}{2^n} \leq \sqrt{\frac{200}{200^5}} < \frac{1}{100}.$$

Portanto $d_{VT}(q, v) \leq E[\varepsilon] < 1/100$ e $d_{VT}(q, w) \leq E[\varepsilon] < 1/100$. Segue que $d_{VT}(v, w) < 1/50$.

◇

Pelo limitante de Markov, se $d_{VT}(v, w) \leq 1/50$, então $\Pr[d_{VT}(v, w) \leq 2/25] \geq 3/4$, i.e. com probabilidade ao menos $3/4$ temos $d_{VT}(v, w) \leq 2/25 < 1/3$, o que significa que ambos $M^{A_{T(n)}}$ e $M^{A_{T(n)-1}}$ ou aceitam, ou rejeitam 1^n .

Como mostrado na Afirmação 3.12.1, a probabilidade de 1^n pertencer exatamente a uma das linguagens $L_{A_{T(n)}}$ e $L_{A_{T(n)-1}}$ é $1/2$. Portanto, com probabilidade de pelo menos $3/4 - 1/2 = 1/4$, $M^{A_{T(n)}}$ ou $M^{A_{T(n)-1}}$ dá a resposta errada na entrada 1^n . Como $A_{T(n)}$ e $A_{T(n)-1}$ são escolhidos da mesma distribuição, $M^{A_{T(n)}}$ retorna a resposta errada com probabilidade de pelo menos $1/8$. Isso leva à conclusão de que M decide L_A com probabilidade de 0 para um oráculo uniformemente aleatório de permutação A , concluindo assim a prova. □

4.3 SEPARAÇÃO ORACULAR GENÉRICA ENTRE $UP \cap coUP$ E $naCQP$

Discutimos agora como usar o Teorema 4.1 para provar o Teorema 1.2 (existe um oráculo genérico A tal que $P^A = naCQP^A \neq UP^A \cap coUP^A$) do mesmo modo que Fortnow e Rogers (1999) usaram o Teorema BBBV (Teorema 3.8) para provar que existe um oráculo genérico A para o qual $P^A = BQP^A \neq (UP \cap coUP)^A$.

Uma *condição de Cohen* é uma função parcial $\{0, 1\}^* \rightarrow \{0, 1\}$ cujo domínio é finito. Uma condição σ *estende* outra condição τ se para todo $x \in \text{dom}(\tau)$ temos $\sigma(x) = \tau(x)$. Um oráculo A estende a condição τ se para todo $x \in \text{dom}(\tau)$ temos $A(x) = \tau(x)$.

Um *oráculo genérico* é um oráculo que estende ao menos uma condição em todo conjunto S . Sendo S :

- Um conjunto de condições de Cohen.
- Definível. Uma condição $\sigma \in S$ pode ser expressa por uma fórmula finita em termos de predicados recursivos, operações booleanas e quantificadores universais.
- Denso. Para toda condição Cohen τ existe uma condição $\sigma \in S$ que estende τ .

Um *tamanho aceitável* é um inteiro no intervalo de uma função *torre*, que tem a seguinte definição recursiva:

$$\begin{aligned} \text{torre}(0) &= 2, \\ \text{torre}(n+1) &= 2^{\text{torre}(n)}. \end{aligned}$$

Uma *condição UP* é uma condição de Cohen que pode ter o valor 1 para no máximo uma string para todo tamanho aceitável e 0 para qualquer string que não tem um tamanho aceitável. Um *oráculo genérico UP* é um oráculo que atende a condição UP.

Demonstração do Teorema 1.2. Sendo B uma linguagem PSPACE completa, e sendo C um oráculo genérico $UP \cap coUP$, Fortnow e Rogers definem

$$A = B \oplus C = \{0x : x \in B\} \cup \{1y : y \in C\}.$$

Fortnow e Rogers afirmam primeiro que, pelo argumento de diagonalização do Teorema de Baker-Gill-Solovay (Teorema 2.7), relativo ao oráculo A ,

$$P^A \neq (UP \cap coUP)^A.$$

Depois, Fortnow e Rogers usam o Teorema BBBV para provar que $P^A = BQP^A$ relativo ao oráculo A como construído. Apontamos os detalhes que devem ser adaptados para que o mesmo seja válido para $P^A = naCQP^A$, desse modo, concluindo a prova para o Teorema 1.2.

Seja M um algoritmo BQP^A (em nosso caso, um algoritmo $naCQP^A$) que roda em tempo $O(p(n))$, e seja W um algoritmo P^A .

Os autores provam que, dado M , é possível construir W tal que, com alta probabilidade, M^A aceita uma entrada x se e somente se W^A também aceita x . Como B é PSPACE-completo, claramente, $M^B(x) = W^B(x)$.

Concernente ao oráculo C , note que M pode consultar C apenas para um número polinomial de strings, também com tamanho polinomial. Pelas condições impostas pela definição do oráculo, C contém apenas um número polinomial de strings que pode alterar a computação de M , com respeito ao caso em que M consulta apenas B .

Como uma das condições impostas em C é que os tamanhos de suas strings são exponencialmente distantes, existe no máximo uma string y , de tamanho ℓ , limitado polinomialmente, tal que W pode encontrar em tempo polinomial todas as strings menores que y em C apenas consultando C .

Se $y \notin C$, então W ao consultar B pode decidir se M^A aceita ou rejeita x . Desse modo, a única possibilidade da saída de M^A para x divergir da saída de W^A é se $y \in C$. Contudo, pelo Teorema BBBV (em nosso caso, Teorema 4.1), conhecemos que o conjunto S de tais strings y que podem alterar com probabilidade maior que ε a decisão de M de aceitar para rejeitar (ou vice-versa) é polinomialmente pequena: $4(p(n))^2/\varepsilon$ pelo Teorema BBBV para BQP; $200(p(n))^4/\varepsilon$ pelo Teorema 4.1 para $naCQP$.

Ora, o problema de encontrar todas as strings em S com tamanho ℓ está em PSPACE. Portanto, W pode encontrar essas strings ao consultar B e, ao consultar C , quanto a cada uma delas, pode encontrar com alta probabilidade se a saída de M é alterada ou não. $\square$

4.4 SEPARAÇÃO ORACULAR EXPONENCIAL ENTRE $UP \cap coUP$ E $naCQP$

Tamon e Yamakami (2001) afirmaram que existe um oráculo A para o qual $P^A = BQP^A$ e $UP^A = EXP^A$, apresentando um esboço de prova para tal. Como $P^B \neq EXP^B$, para todo oráculo B , temos que $P^A = BQP^A \neq UP^A = EXP^A$. Utilizamos as ideias principais desse esboço de prova, para provarmos que existe um oráculo A para o qual $P^A = naCQP^A$ e $UP^A \cap coUP^A = EXP^A$. Considerando que o esboço de Tamon e Yamakami não é fácil de entender e contém detalhes técnicos que não estamos certos se estão corretos, a demonstração para o Teorema 1.3 não somente prova o resultado mais forte para as classe $naCQP$ e $UP \cap coUP$, como também fornece uma prova completa para o resultado mais restrito de Tamon e Yamakami.

Demonstração do Teorema 1.3. Seja $M^?$ um algoritmo $naCQP$ que pode consultar um oráculo tal que dada a entrada $(0^j, x)$, $j \leq |x|$: roda em tempo linear em $j + |x|$ para qualquer oráculo; faz uma consulta com tamanho menor que $|x|$; e tem probabilidade de erro menor que $2^{-j-|x|}$. Seja $\{M_k^?\}_{k \in \mathbb{N}}$ uma enumeração de tais algoritmos.

AFIRMAÇÃO 1.3.1. Seja $\{M_k^A\}_{k \in \mathbb{N}}$ a enumeração dos algoritmos $naCQP^A$ como definidos e restritos acima. Então $naCQP^A$ é o conjunto das linguagens *decididas* por um algoritmo no conjunto $\{M_k^A\}_{k \in \mathbb{N}}$.

Demonstração da afirmação. A prova usa um argumento de *padding*. Seja $W^?$ um algoritmo $naCQP$ que consulta um oráculo e roda em tempo no máximo $|x|^c$ para algum $c \in \mathbb{N}$. Então, $M_k^?$ é o algoritmo $naCQP$ que consulta um oráculo e que, para a entrada $(0^{|x|}, x0^{|x|^c-|x|})$, emula

$W^?(x)$. Assumindo que $M_k^?$ rejeita se a entrada não condiz com esse formato, claramente temos que $M_k^?$, para a entrada $(0^j, y)$, roda em tempo linear no tamanho de entrada (i.e. $j + |y|$) e faz consultas para o oráculo apenas com tamanho menor que $|y|$. Assumimos também sem perda de generalidade que se $x \in L$, então $\Pr[M_k(0^{|x|}, x0^{|x|^c-|x|}) = 1] \geq 1 - 2^{-|x|-|x|^c}$, se $x \notin L$, então $\Pr[M_k(0^{|x|}, x0^{|x|^c-|x|}) = 1] \leq 2^{-|x|-|x|^c}$. $\diamond$

Para o oráculo A que queremos construir, seja L^A uma linguagem completa EXP^A e E^A uma máquina de Turing determinística com consulta a A que decide L^A em tempo exponencial. Podemos também assumir sem perda de generalidade que E^A roda em tempo $2^{|x|}$ dada a entrada $(0^{j \leq |x|}, x)$. Seja $\langle \cdot, \cdot, \cdot, \cdot, \cdot \rangle$ uma função de Σ^5 para Σ tal que $|\langle a, b, c, d, e \rangle| > \max\{|a|, |b|, |c|, |d|, |e|\}$ para todos $a, b, c, d, e \in \Sigma$. Dados $k > 0$ e $s \in \{0, 1\}^*$, denotamos s_k^n como a $(k + 1)$ -ésima string em $\{0, 1\}^n$. Se $k = 0$ temos $s_0^n = 0^n$. Construímos A tal que para qualquer n suficientemente grande e todas strings x de tamanho n e todo k com $0 \leq k < n$,

- (a) $\Pr[M_k^A(0^{j \leq n}, x) = 1] \geq 1 - 2^{-2n} \Rightarrow \langle 00, s_k^{\lceil \log(n) \rceil}, x, 1^{n^2}, 1^{n^2} \rangle \in A$;
- (b) $\Pr[M_k^A(0^{j \leq n}, x) = 0] \geq 1 - 2^{-2n} \Rightarrow \langle 00, s_k^{\lceil \log(n) \rceil}, x, 1^{n^2}, 1^{n^2} \rangle \notin A$;
- (c) $x \in L^A \Rightarrow |\{y \in \Sigma^{n^2} : \langle 01, 1^{\lceil \log(n) \rceil}, x, y, 1^{n^2} \rangle \in A\}| = 1$ e $|\{z \in \Sigma^{n^2} : \langle 10, 1^{\lceil \log(n) \rceil}, x, 1^{n^2}, z \rangle \in A\}| = 0$;
- (d) $x \notin L^A \Rightarrow |\{y \in \Sigma^{n^2} : \langle 01, 1^{\lceil \log(n) \rceil}, x, y, 1^{n^2} \rangle \in A\}| = 0$ e $|\{z \in \Sigma^{n^2} : \langle 10, 1^{\lceil \log(n) \rceil}, x, 1^{n^2}, z \rangle \in A\}| = 1$.

AFIRMAÇÃO 1.3.2. Se condições (a) e (b) são satisfeitas, então $\text{P}^A = \text{naCQP}^A$.

Demonstração da afirmação. Uma máquina de Turing determinística pode decidir a mesma linguagem que M_k^A ao consultar A sobre $\langle 00, s_k^{\lceil \log(n) \rceil}, x, 1^{n^2}, 1^{n^2} \rangle$. $\diamond$

AFIRMAÇÃO 1.3.3. Se condições (c) e (d) são satisfeitas, então $\text{UP}^A \cap \text{coUP}^A = \text{EXP}^A$.

Demonstração da afirmação. Uma máquina de Turing não determinística pode decidir L^A ao encontrar um único $y \in \Sigma^{n^2}$ e consultar o oráculo sobre $\langle 01, 1^{\lceil \log(n) \rceil}, x, y, 1^{n^2} \rangle$, essa máquina também pode decidir $\text{co}L^A$ ao encontrar um único $z \in \Sigma^{n^2}$ e consultar o oráculo sobre $\langle 10, 1^{\lceil \log(n) \rceil}, x, 1^{n^2}, z \rangle$. $\diamond$

Dizemos que:

- w é uma 00-string se w tem a forma $\langle 00, s_k^{\lceil \log(n) \rceil}, x, 1^{n^2}, 1^{n^2} \rangle$ com $0 \leq k < n$;
- w é uma 01-string se w se tem a forma $\langle 01, 1^{\lceil \log(n) \rceil}, x, y, 1^{n^2} \rangle$ com $|y| = n^2$;
- w é uma 10-string se w tem a forma $\langle 10, 1^{\lceil \log(n) \rceil}, x, 1^{n^2}, z \rangle$ com $|z| = n^2$.

Construímos A por estágios, para cada entrada x seguindo a ordem lexicográfica. Em cada estágio o oráculo é descrito como uma função parcial σ_A . O estágio inicial é definido como $\sigma_A(\lambda) = 0$ para uma string vazia λ . Para cada estágio x , consideramos todas as possíveis strings w de $w = \langle 00, 0, x, 0, 0 \rangle$ até $w = \langle 10, 1^{\lceil \log(n) \rceil}, x, 1^{n^2}, 1^{n^2} \rangle$.

Seja $p(w)$ um qubit definido abaixo de modo recursivo. Queremos que $p(w)$ represente a resposta do oráculo, de modo que, se a resposta for 0 (1), então $p(w)$ é $|0\rangle$ ($|1\rangle$) com alta probabilidade, i.e. $\geq 1 - 2^{-2n}$.

Passo 1. Para toda string w que não é 00-string, 01-string, ou 10-string: $p(w) = |0\rangle$.

Passo 2. Atribuímos para toda 01-string w e 10-string w a variável v_w que indica se w pertence ao oráculo A , e fazemos $p(w) = |v_w\rangle$. A variável v_w será valorada mais tarde. Se $w \in \text{dom}(\sigma_A)$, então $p(w) = |\sigma_A(w)\rangle$.

Passo 3. Se w é uma 00-string, definimos $p(w)$ como o estado quântico correspondente à saída, antes da medição final, de $M_k^A(0^{j \leq n}, x)$. Note que, como o tamanho de cada consulta é menor que $|x|$, e, portanto, menor que $\sqrt{|w|}$, podemos considerar as respostas para essas consultas como já definidas por recursão. Então, adicionamos w para A se e somente se a saída majoritária de $p(w)$ é 1.

Para completar a construção do oráculo A no estágio x , precisamos valorar v_w quando w é ou uma 01-string ou uma 10-string. Denotamos a string $\mathbf{r}$ de tamanho $2 \cdot 2^{n^2}$ contendo em sua primeira metade a indexação de todas as strings $y \in \Sigma^{n^2}$ e em sua segunda metade a indexação de todas as strings $z \in \Sigma^{n^2}$. Isto é, cada bit da primeira metade de $\mathbf{r}$ corresponde a uma 01-string na forma $\langle 01, 1^{\lceil \log(n) \rceil}, x, y, 1^{n^2} \rangle$ e cada bit da segunda metade de $\mathbf{r}$ corresponde a uma 10-string na forma $\langle 10, 1^{\lceil \log(n) \rceil}, x, 1^{n^2}, z \rangle$. Definimos $C_x(\mathbf{r})$ como a saída da emulação de $E^A(0^{j \leq n}, x)$ com a condição que quando E fizer uma consulta w , computamos $p(w)$ e usamos a saída majoritária.

Afirmção 1.3.4. Para todo x existe uma string $\mathbf{r}$ dada por $\mathbf{r} = \mathbf{d}_i = 0^i 10^{2^{n^2+1}-i-1}$ ($0 \leq i < 2^{n^2+1}$) que pode estar em um dos dois casos:

1. $C_x(\mathbf{d}_i) = 1$ se $(i < 2^{n^2})$;
2. $C_x(\mathbf{d}_i) = 0$ se $(i \geq 2^{n^2})$.

Interpretamos a string $\mathbf{r}$ do seguinte modo. Atribuímos $v_w = 1$ para a string w indexada pelo $(i+1)$ -ésimo bit de $\mathbf{r}$, e $v_w = 0$ para todo w indexado pelos demais bits de $\mathbf{r}$.

Demonstração da afirmação. Devido a recursão usada para obter $p(w)$ para as 00-strings, o número de níveis necessários para computar $C_x(\mathbf{r})$ é no máximo $\lfloor \log n \rfloor - 1$. Portanto, o número de consultas que $C_x(\mathbf{r})$ faz é no máximo $\prod_{i=0}^{\lfloor \log n \rfloor} (2^n)^{1/2^i}$, que é menor que 2^{2n-1} . Portanto, a probabilidade de $C_x(\mathbf{r})$ obter a resposta correta é maior que $(1 - 2^{-2n})^{2^{2n-1}} > \sqrt{1/e} > 2/3$, para n suficientemente grande.

Precisamos, finalmente, considerar que uma subcomputação $p(w)$ pode retornar um resultado errado com probabilidade maior que 2^{-2n} . Para tais casos podemos considerar que a subcomputação $p(w)$ retorna $|1\rangle$ ou $|0\rangle$ de modo arbitrário. De acordo com o Teorema 4.1, temos um conjunto polinomialmente pequeno S de respostas do oráculo que pode alterar com probabilidade maior que ε a saída de um algoritmo naCQP. Concluimos, portanto, que sempre existe um modo de atribuir os valores de v_w para esses valores w “ruins” de modo a não ocorrerem os casos:

1. $(i < 2^{n^2})$ e $C_x(\mathbf{d}_i) = 0$; e
2. $(i > 2^{n^2})$ e $C_x(\mathbf{d}_i) = 1$.

◇

O Passo 3, correspondente à construção das 00-strings, juntamente com a Afirmção 1.3.4, correspondente à construção das 01-strings e 10-strings, cumprem as Condições (a), (b), (c) e (d), assim, concluindo a prova. □

5 PROBLEMAS RELACIONADOS A CLASSES DE CONHECIMENTO ZERO

Parte dos resultados apresentados nesse capítulo correspondem a um trabalho feito em conjunto com Abner F. B. Costa, cujo foco de pesquisa era o problema do subgrupo oculto (Costa, 2024).

O Problema do Subgrupo Oculto, referido por HSP (do inglês “Hidden Subgroup Problem”) generaliza muitos problemas que se suspeita serem NP-intermediários, tais como Fatoração (Shor, 1994) e Isomorfismo de Grafos (Jozsa, 2001), que são problemas de bastante interesse no contexto de computação quântica. A definição comumente encontrada na literatura para HSP é a introduzida por Babai e Szemerédi (1984) num contexto de problemas de caixa preta. Determinar se um subgrupo oculto é ou não o subgrupo trivial é uma definição bem estudada para uma versão de decisão de HSP (Ettinger et al., 2004; Hayashi et al., 2008; Sdroievski et al., 2019) e é a utilizada neste trabalho, embora não seja a única versão de decisão (detalhes técnicos são discutidos na sequência). Esta versão de decisão, referida por dHSP, foi demonstrada estar na classe de conhecimento zero HVPZK e, se a dimensão do grupo for conhecida, também pertence à classe NISZK (Sdroievski et al., 2019).

Investigamos restrições sob as quais dHSP tem um protocolo de conhecimento zero perfeito não-interativo com abrangência perfeita (i.e. pertence à classe NIPZK₁). Mostramos que isto é válido sempre que podemos amostrar uniformemente ao acaso elementos do grupo e de um conjunto, com a mesma dimensão do grupo, que contenha a imagem da função que esconde o subgrupo (Teorema 1.4). Outras restrições são também apresentadas (Corolários 5.2 e 5.3).

A classe NIPZK foi definida por Malka (2008), que mostrou um problema completo para a classe. Ele também insinuou que, com uma certa restrição, este problema poderia tornar-se completo para NIPZK₁, mas não provou explicitamente esta afirmação, uma vez que não era esse o objetivo do trabalho. Chamamos esse problema de Uniforme-ou-Pequeno (US, de *Uniform-or-Small*) e mostramos sua prova na Seção 5.2.

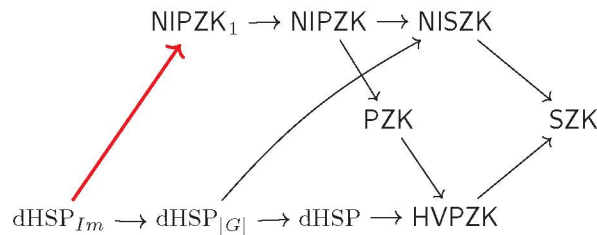


Figura 5.1: Uma seta $A \rightarrow B$ representa: que $A \subseteq B$ se A e B são classes; que $A \in B$, se A é um problema e B é uma classe; que A é uma restrição de B , se A e B são problemas. Nosso resultado é destacado por uma seta vermelha em negrito.

A Figura 5.1 mostra as relações conhecidas entre as classes de complexidade de conhecimento zero e as restrições de HSP. Para distinguir as diferentes versões de decisão mencionadas, usamos dHSP para o caso geral, dHSP_{|G|} para o caso restrito quando a dimensão do grupo é conhecida, e dHSP_{Im} para o nosso caso restrito.

Assumimos que o leitor está familiarizado com os tópicos básicos da teoria dos grupos, para os quais remetemos para Herstein (1991) e Costa (2024).

Considere uma família de grupos $\mathcal{B} = \{B_n\}_{n \geq 1}$ tal que: os elementos de B_n são representados unicamente por palavras de comprimento $\text{poly}(n)$; e as operações, de cada grupo B_n , inverso, produto e teste de identidade são computadas em tempo $\text{poly}(n)$, denotamos por e

o elemento identidade de B_n . A definição formal de HSP que consideramos é a proposta por Sdroievski et al. (2019), em que nos é *dado* um inteiro positivo n (em unário) e um circuito booleano C_f que recebe codificações de elementos de um grupo $G \subseteq B_n$ como entrada e devolve uma saída de m bits, sendo m um inteiro positivo. Assumimos que C_f calcula uma função f que *oculta* um subgrupo H em G , i.e. $f(a) = f(b)$ se e somente se $aH = bH$ para todos $a, b \in G$. O *objetivo* do problema HSP é produzir um conjunto gerador de H . Agora, para dHSP, como definido abaixo, o *objetivo* é decidir se H é trivial, i.e. $f(a) = f(b)$ se e somente se $a = b$.

dHSP (para uma família de grupos $\mathcal{B} = \{B_n\}_{n \geq 1}$ como acima)

Dados: $(0^n, T, C_f)$, onde T é o conjunto gerador de um grupo $G \subseteq B_n$, sendo $|T| = \text{poly}(n)$, e C_f é um circuito de tamanho $\text{poly}(n)$ que recebe codificações de elementos de B_n e devolve strings com m bits, para algum $m \in \mathbb{Z}_{\geq 0}$, de modo que C_f computa uma função f que oculta um subgrupo H em G ;

decidir entre: *instância positiva:* $\text{dHSP}_Y = \{(0^n, T, C_f) : |H| = 1\}$;
instância negativa: $\text{dHSP}_N = \{(0^n, T, C_f) : |H| \geq 2\}$;

prometido que f oculta um subgrupo $H \subseteq G$.

Além da relação do problema HSP com classes de conhecimento zero, uma questão de interesse nesse trabalho é a relação de classes de conhecimento zero com computação quântica. Especificamente, exploramos a seguinte pergunta:

Pergunta 5.1. $\text{NIPZK}_1 \subseteq \text{coQMA}$?

Sabe-se que $\text{SZK} \subseteq \text{AM} \cap \text{coAM}$. Portanto, sob hipóteses de desaleatorização, suspeita-se que $\text{SZK} \subseteq \text{NP} \cap \text{coNP}$, porém essa não parece ser uma pergunta fácil de responder. Observe que $\text{SZK} \subseteq \text{MA} \cap \text{coMA}$ implica $\text{NIPZK} \subseteq \text{coQMA}$ uma vez que $\text{NIPZK} \subseteq \text{SZK}$ e $\text{coQMA} \supseteq \text{coMA}$; ressalte-se que se conjectura que $\text{MA} = \text{AM}$ o que é mais uma evidência de $\text{NIPZK} \subseteq \text{coQMA}$.

Com relação a separações oraculares, sabemos que existe um oráculo para o qual NISZK não está contido em PP , uma superclasse de QMA e coQMA (Bouland et al., 2019). Também existe um oráculo para o qual NIPZK_1 não está contido em QMA (Aaronson, 2012; Sherstov e Thaler, 2022).

Uma maneira de demonstrarmos $\text{NIPZK}_1 \subseteq \text{coQMA}$ é provando que um problema completo para NIPZK_1 pertence a coQMA . Uma estratégia possível é verificar primeiro que tipo de instâncias não necessitam de um provador, estando a restrição correspondente do problema em BQP . Um resultado foi a identificação de uma dessas restrições, a qual chamamos de Distância Estatística entre Distribuição Biparticionada e Distribuição Uniforme (BSDU). Provamos que $\text{BSDU} \in \text{BQP}$, um resultado que também é interessante por si só, pois não parece razoável que $\text{BSDU} \in \text{BPP}$.

Neste capítulo mostraremos algumas restrições para as quais dHSP pertence à classe NIPZK_1 , o problema completo para NIPZK_1 , e que $\text{BSDU} \in \text{BQP}$. Depois, discutimos a conjectura $\text{NISZK} \subseteq \text{coQMA}$.

5.1 VERSÃO DE DECISÃO DE HSP E CLASSES DE COMPLEXIDADE DE CONHECIMENTO ZERO

Embora não saibamos se dHSP pertence à classe NP , o problema pertence à classe coNP , pois para $|H| \geq 2$, podemos usar um elemento $h \neq e \in H$ como um certificado para uma

instância negativa. Para verificar isso, verificamos se $f(h) = f(e)$. É mostrado em Sdroievski et al. (2019) um protocolo perfeito de conhecimento zero com verificador honesto para dHSP, estabelecendo que $\text{dHSP} \in \text{HVPZK}$. Além disso, Sdroievski et al. (2019) demonstrou que, se soubermos o tamanho do grupo G , como é o caso dos grupos de permutação Seress (2003), então há uma redução polinomial de Karp de dHSP para o Problema de Aproximação de Entropia (EA), um problema de promessa completo para NISZK.

Observamos que $\text{Im}(f)$, a imagem da função f , tem no máximo $|G|$ elementos, portanto C_f tem no máximo $|G|$ possíveis saídas diferentes. Definimos dHSP_{Im} como uma restrição de dHSP em que podemos fazer uma amostragem uniforme de elementos aleatórios de G e de um conjunto $A_f \subseteq \{0, 1\}^m$ de modo que $\text{Im}(f) \subseteq A_f$ e $|A_f| = |G|$. Embora essa restrição possa parecer artificial, observe que ela é válida para casos importantes, como quando $G = \mathbb{Z}_n$ com adição ou multiplicação modular. O Corolário 5.2 expande ainda mais quando essa restrição pode ser atendida.

Demonstramos agora o Teorema 1.4 já enunciado na introdução do trabalho (Capítulo 1).

Teorema 1.4 (Costa et al., 2023). $\text{dHSP}_{\text{Im}} \in \text{NIPZK}_1$.

Demonstração. Seja $r \in A_f$, escolhido uniformemente de forma aleatória, a string compartilhada e seja $B_r = \{g \in G : f(g) = r\}$. O provador P faz uma amostragem de $g \in B_r$ uniformemente aleatória e a envia para o verificador V , que aceita se $f(g) = r$ e rejeita caso contrário.

Como C_f é um circuito de tamanho $\text{poly}(n)$, o verificador V pode calcular $f(g)$ de modo eficiente. Se $|H| = 1$, então V sempre aceita, alcançando a propriedade de abrangência. Agora, mostramos a propriedade de credibilidade. Se $|H| \geq 2$, então $|\text{Im}(f)|/|G| \leq 1/2$. Como $r \in A_f$ é escolhido uniformemente de modo aleatório, a probabilidade de $r \in \text{Im}(f)$ é de no máximo $1/2$. Portanto, a probabilidade de que exista $g \in G$ que possa ser enviado de P para V de modo que $f(g) = r$ é de no máximo $1/2$.

Para obtermos a propriedade de conhecimento zero, considere S como o simulador que escolhe $g' \in G$ uniformemente de modo aleatório e calcula $r' = f(g')$. Como $|H| = 1$, a função f é uma bijeção, obtendo $r' \in A_f$ também uniformemente de modo aleatório. Portanto, as transcrições $\langle C_f, r', g' \rangle$ do simulador S e as transcrições $\langle C_f, r, g \rangle$ do protocolo são distribuídas de forma idêntica sempre que $|H| = 1$. $\square$

Os Corolários 5.2 e 5.3 decorrem imediatamente do Teorema 1.4.

Corolário 5.2 (Costa et al., 2023). *Se existe uma amostragem uniforme de elementos aleatórios de G e se A_f é o conjunto das primeiras $|G|$ strings em ordem lexicográfica, então $\text{dHSP} \in \text{NIPZK}_1$.*

Corolário 5.3 (Costa et al., 2023). *Se existe uma amostragem uniforme de elementos aleatórios de G , e o circuito C_f tem como saída strings de tamanho $(\log |G|)$, então $\text{dHSP} \in \text{NIPZK}_1$.*

5.2 UM PROBLEMA COMPLETO PARA NISZK₁

A classe NIPZK foi definida por Malka (2008), que também apresentou um problema de promessa completo para a classe. Ele também deu a entender que uma versão restrita desse problema é completa para a classe NIPZK₁, mas não provou explicitamente essa afirmação. Mostramos, seguindo uma estrutura semelhante da prova de Malka, que o resultado é realmente válido.

A seguir, definimos o problema *Uniforme-ou-Pequeno* (US, de *Uniform-or-Small*), em que U_m denota a distribuição uniforme em todas as strings de m bits e $\text{suporte}(X) = \{y \in$

$\{0, 1\}^m : \Pr[X = y] \neq 0\}$ é o suporte de X . Esse problema já foi estudado por Dixon et al. (2020), que mostrou que existe um oráculo A em relação ao qual US não está na classe probabilística SBP.

US

Dado: um circuito $C : \{0, 1\}^n \rightarrow \{0, 1\}^m$ de tamanho $\text{poly}(n)$ que codifica uma distribuição X ;

decidir entre: instância positiva: $\text{US}_Y = \{X : d_{VT}(X, U_m) = 0\}$;

instância negativa: $\text{US}_N = \{X : |\text{suporte}(X)| \leq 2^m/3\}$.

prometido que vale um dos casos.

Demonstramos agora o Teorema 1.5 já enunciado na introdução do trabalho (Capítulo 1).

Teorema 1.5 (Costa et al., 2023). *O problema US é completo para NIPZK_1 .*

Demonstração que $\text{US} \in \text{NIPZK}_1$. Seja $r \in \{0, 1\}^m$, escolhido uniformemente de forma aleatória, a string compartilhada e seja $B_r = \{\pi \in \{0, 1\}^n : X(\pi) = r\}$. O provador P faz uma amostragem uniforme e aleatória de $\pi \in B_r$ e a envia para o verificador V , que aceita se $X(\pi) = r$ e rejeita caso contrário.

Como C é um circuito de tamanho $\text{poly}(n)$, o verificador V pode calcular $X(\pi)$ eficientemente. Se $X \in \text{US}_Y$, então V sempre aceita, obtendo a propriedade de abrangência. Se $X \in \text{US}_N$, então, como $|\text{suporte}(X)| \leq 2^m/3$ e r é uniformemente aleatório, a probabilidade de $r \in \text{suporte}(X)$ é de no máximo $1/3$. Portanto, a probabilidade de que haja algum $\pi \in \{0, 1\}^n$ com $X(\pi) = r$ que possa ser enviado por P para V é de no máximo $1/3$, obtendo a propriedade de credibilidade.

Para obtermos a propriedade de conhecimento zero, consideremos $X \in \text{US}_Y$ e S como o simulador que escolhe $\pi' \in \{0, 1\}^n$ uniformemente de forma aleatória e computa $r' = X(\pi')$. Quando $X \in \text{US}_Y$, obtemos $r' \in \{0, 1\}^m$ uniformemente de forma aleatória. Portanto, as transcrições $\langle X, r', \pi' \rangle$ do simulador S e as transcrições $\langle X, r, \pi \rangle$ do protocolo são distribuídas de forma idêntica sempre que $X \in \text{US}_Y$. $\diamond$

Demonstração que US é NIPZK_1 -difícil. Seja $\Pi = \langle \Pi_Y, \Pi_N \rangle$ um problema de NIPZK_1 e seja (P, V) um protocolo não interativo para Π com abrangência perfeita e erro de credibilidade de no máximo $1/3$. Denotamos r como a string compartilhada, escolhida uniformemente de forma aleatória, com tamanho $|r| = |x|^c$ para algum $c \in \mathbb{N}$ e todo $x \in \Pi_Y \cup \Pi_N$. Seja S um simulador para $\langle P, V \rangle$ e d uma constante tal que S não use mais do que $|x|^d$ bits aleatórios para toda entrada x .

Mostramos que Π tem uma redução polinomial de Karp para US. Definimos uma máquina de Turing polinomial que, dada a entrada $x \in \Pi_Y \cup \Pi_N$, tem como saída um circuito $C : \{0, 1\}^{|x|^d} \rightarrow \{0, 1\}^{|x|^c}$ que codifica uma distribuição X , de modo que se $x \in \Pi_Y$, então $X \in \text{US}_Y$ e se $x \in \Pi_N$, então $X \in \text{US}_N$. Dado x , o circuito C é construído de forma que, dada a entrada r_S com $|r_S| \leq |x|^d$, emula a computação de S em x sob a aleatoriedade de r_S , produzindo a visualização $\langle x, r, \pi \rangle$, em que r é a string compartilhada em (P, V) e π é a mensagem enviada de P para V . Então, C tem como saída r se $V(x, r, \pi) = \text{aceita}$ e $0^{|x|^c}$ caso contrário.

Agora, analisamos a redução. Se $x \in \Pi_Y$, então $V(x, r, \pi) = \text{aceita}$. Por construção, C tem como saída r . Como r é uniformemente distribuído, X é a distribuição uniforme. Se $x \in \Pi_N$, mostramos que $|\text{suporte}(X)| \leq 2^{|x|^c}/3$. Pela condição de credibilidade, a probabilidade do verificador $V(x)$ aceitar em $\langle P, V \rangle$ é menor que $1/3$. Assim, por construção, em mais

de $2/3$ casos para todos os $2^{|x|^c|}$ possíveis resultados de C temos a string $0^{|x|^c|}$. Portanto, $|\text{suporte}(X)| \leq 2^{|x|^c|}/3$. $\diamond$

Claramente, $|\text{suporte}(X)| \leq 2^m/3$ na definição de US poderia ser substituído por $|\text{suporte}(X)| \leq \beta \cdot 2^m$ para qualquer constante positiva $\beta < 1$, e o Teorema 1.5 também seria válido. De fato, US foi declarado de forma equivalente por Dixon et al. (2020) com $\beta = 1/2$.

5.3 BSDU $\in$ BQP

Recordemos, da definição do problema SDU, problema completo para NISZK, que recebemos um circuito booleano clássico C de tamanho polinomial em n que codifica uma distribuição de probabilidade X sobre todas as strings com n bits. Podemos, a partir de C , obter em tempo polinomial clássico um circuito quântico Q com n qubits de saída e $m > n$ de entrada tal que, recebendo $|0\rangle^{\otimes m}$, devolve o estado misto cuja matriz de densidade ρ é a matriz diagonal das probabilidades das 2^n strings. Desse modo, rescrevemos a definição de SDU.

SDU

Dados: dois inteiros não-negativos m e n tais que $m \geq n$ e um circuito quântico Q de tamanho polinomial em m com m qubits de entrada e n qubits de saída;

sendo: ρ uma matriz diagonal e a saída de Q ao receber $|0\rangle^{\otimes m}$;

decidir entre: *instância positiva:* $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} < 1/n$;

instância negativa: $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} > 1 - 1/n$;

prometido que vale um dos casos.

Definimos o problema BSDU a seguir.

BSDU

Dados: dois inteiros não-negativos m e n tais que $m \geq n$ e um circuito quântico Q de tamanho polinomial em m com m qubits de entrada e n qubits de saída;

sendo: ρ a saída de Q ao receber $|0\rangle^{\otimes m}$, de modo que ρ é uma matriz diagonal, e para todo $i \leq 2^n/2$ e todo $j > 2^n/2$, temos $\rho_{ii} \geq \rho_{jj}$;

decidir entre: *instância positiva:* $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} < 1/n$;

instância negativa: $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} > 1 - 1/n$;

prometido que vale um dos casos.

Demonstramos agora o Teorema 1.6 já enunciado na introdução do trabalho (Capítulo 1). Quando os primeiros $2^n/2$ da diagonal de ρ não são maiores ou iguais aos outros $2^n/2$ elementos, podemos resolver BSDU em tempo polinomial quântico ao se considerar apenas o primeiro qubit.

Teorema 1.6 (Hepp et al., 2021). *BSDU está em BQP.*

Demonstração. Supomos sem perda de generalidade que $n > 3$, pois, caso contrário, poderíamos decidir entre $d_{VT}(X, U_n) < 1/n$ e $d_{VT}(X, U_n) > 1 - 1/n$ em tempo $O(1)$ clássico. Seja ρ a matriz de densidade devolvida pelo circuito quântico Q , conforme discutimos. Sejam $\alpha = 1/n$, $\beta = 1 - 1/n$, $N = 2^n$, e σ a matriz de densidade reduzida do primeiro qubit obtida ao se fazer o *traço parcial* dos demais qubits de ρ . Vamos mostrar que:

- (i) se $\|\rho - \mathbb{1}/N\|_{\text{tr}} < \alpha$, então $\|\sigma - \mathbb{1}/2\|_{\text{tr}} < \alpha$;

(ii) se $\|\rho - \mathbb{1}/N\|_{\text{tr}} > \beta$, então $\|\sigma - \mathbb{1}/2\|_{\text{tr}} > \beta/2$.

Observe que

$$\sigma = \left(\sum_{i=1}^{N/2} \rho_{ii} \right) |0\rangle \langle 0| + \left(\sum_{i=N/2+1}^N \rho_{ii} \right) |1\rangle \langle 1|.$$

Considerando a definição do problema BSDU e que ρ e σ são matrizes diagonais, fazendo $\delta_k := \rho_{kk} - 1/N$ para todo k , temos $\delta_i \geq \delta_j$ para todo $i \leq N/2$ e todo $j > N/2$ e:

$$\begin{aligned} \|\rho - \mathbb{1}/N\|_{\text{tr}} &= \frac{1}{2} (|\rho_{11} - 1/N| + \cdots + |\rho_{NN} - 1/N|) \\ &= \frac{1}{2} (|\delta_1| + \cdots + |\delta_N|); \\ \|\sigma - \mathbb{1}/2\|_{\text{tr}} &= \frac{1}{2} (|\rho_{11} + \rho_{22} + \cdots + \rho_{N/2, N/2} - 1/2| + |\rho_{N/2+1, N/2+1} + \cdots + \rho_{N, N} - 1/2|) \\ &= \frac{1}{2} (|\delta_1 + \delta_2 + \cdots + \delta_{N/2}| + |\delta_{N/2+1} + \cdots + \delta_N|). \end{aligned}$$

(i) Por desigualdade triangular, $\|\sigma - \mathbb{1}/2\|_{\text{tr}} < \alpha$ segue imediatamente da promessa $\|\rho - \mathbb{1}/N\|_{\text{tr}} < \alpha$.

(ii) Como o traço de ρ é igual a 1 e todos os elementos de ρ são reais não-negativos, temos

$$\|\rho - \mathbb{1}/N\|_{\text{tr}} = \sum_{\delta_i > 0} \delta_i = \sum_{\delta_i < 0} |\delta_i| > \beta;$$

$$-1/N \leq \delta_i \leq 1 - 1/N \text{ para todo } i.$$

Observe que, de $\sum_{\delta_i < 0} |\delta_i| > \beta$, segue que

$$x := |\{\delta_i : \delta_i > 0\}| > 0;$$

$$|\{\delta_i : \delta_i < 0\}| = N - x > N/2.$$

Portanto, como $\delta_i \geq \delta_j$ para todo $i \leq N/2$ e todo $j > N/2$, a soma $|\delta_1 + \delta_2 + \cdots + \delta_{N/2}| + |\delta_{N/2+1} + \cdots + \delta_N|$ é minimizada quando todos os $N - x$ valores em $\{|\delta_i| : \delta_i < 0\}$ são iguais a

$$\frac{\sum_{\delta_i < 0} |\delta_i|}{N - x} > \frac{\beta}{N - x}.$$

Assim,

$$\begin{aligned} \|\sigma - \mathbb{1}/2\|_{\text{tr}} &= \frac{1}{2} (|\delta_1 + \delta_2 + \cdots + \delta_{N/2}| + |\delta_{N/2+1} + \cdots + \delta_N|) \\ &> \frac{1}{2} \left(\beta - \left(\frac{N}{2} - x \right) \frac{\beta}{N - x} + \frac{N}{2} \left(\frac{\beta}{N - x} \right) \right) \\ &\geq \frac{\beta}{2} \left(\frac{N}{N - x} \right) \geq \frac{\beta}{2} \left(\frac{N}{N - 1} \right) > \frac{\beta}{2} \end{aligned}$$

Como consequência, o problema BSDU reduz-se a $(1/n, 1/2 - 1/(2n))$ -1QSC; com isto, o algoritmo polinomial quântico para BSDU segue da Proposição 3.5. $\square$

Podemos generalizar o resultado do Teorema 1.6 para resolver em BQP o problema que chamamos de BSDU_i , para $1 \leq i \leq n$ fixo:

BSDU_i

Dados: dois inteiros não-negativos m e n tais que $m \geq n$ e um circuito quântico Q de tamanho polinomial em m com m qubits de entrada e n qubits de saída;

sendo: ρ a saída de Q ao receber $|0\rangle^{\otimes m}$, de modo que ρ é uma matriz diagonal e σ_i , a matriz de densidade reduzida do i -ésimo qubit;

decidir entre: *instância positiva:* $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} < 1/n$;

instância negativa: $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} > 1 - 1/n$ e $\|\sigma_i - \mathbb{1}/2\|_{\text{tr}} > 1/2 - 1/(2n)$;

prometido que vale um dos casos.

Note que $\text{BSDU} = \text{BSDU}_1$.

5.4 SOBRE A PERGUNTA $\text{US} \in \text{coQMA}$

Recordemos, da definição do problema US , problema completo para NIPZK_1 , que recebemos um circuito booleano clássico C de tamanho polinomial em n que codifica uma distribuição de probabilidade X sobre todas as strings com n bits. Conforme a Seção 3.1, é possível, a partir de C , obter em tempo polinomial clássico um circuito quântico Q com n qubits de saída e $m > n$ de entrada tal que, recebendo $|0\rangle^{\otimes m}$, devolve o estado misto cuja matriz de densidade ρ é a matriz diagonal das probabilidades das 2^n strings. Desse modo, rescrevemos a definição de US .

US

Dados: dois inteiros não-negativos m e n tais que $m \geq n$ e um circuito quântico Q de tamanho polinomial em m com m qubits de entrada e n qubits de saída;

sendo: ρ uma matriz diagonal e a saída de Q ao receber $|0\rangle^{\otimes m}$;

decidir entre: *instância positiva:* $\rho = \mathbb{1}/2^n$;

instância negativa: $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} \geq 2/3$;

prometido que vale um dos casos.

Seja (P, V) um protocolo interativo quântico de $q(m)$ qubits com uma única mensagem (do provador para o verificador). Temos que $\text{US} \in \text{coQMA}$ se:

1. (Eficiência) (P, V) é polinomialmente limitado e V é computável em tempo polinomial;
2. (Abrangência) se $x \in \text{US}_N$, então V aceita x com probabilidade pelo menos $2/3$ na ação do circuito (P, V) com uma mensagem sobre $|0\rangle^{\otimes q(m)}$;
3. (Credibilidade) se $x \in \text{US}_S$, então para qualquer P^* , o verificador V rejeita x com probabilidade pelo menos $2/3$ na ação do circuito (P^*, V) com uma mensagem sobre $|0\rangle^{\otimes q(m)}$.

Seja σ_i a matriz de densidade reduzida do i -ésimo qubit da instância. Observe que $x \in \text{US}_S$ implica $\sigma_i = \mathbb{1}/2$, para qualquer i . Assim, o verificador não precisa do provador para aceitar as instâncias positivas de US e tampouco precisa ser quântico, pois bastaria, para uma

amostragem da saída verificar se segue a distribuição uniforme. No entanto, isso ainda não é suficiente para a existência de um protocolo dotado de abrangência.

Se $x \in \text{US}_N$, não é possível usar a estratégia usada para resolver BSDU_i , pois podemos ter o *contraexemplo* de ρ representar uma distribuição suficientemente distante da uniforme, i.e. $\|\rho - \mathbb{1}/2^n\|_{\text{tr}} > 2/3$, e ainda assim a cada qubit individualmente corresponder, de modo exato ou aproximado, a matriz de densidade maximalmente mista. Um menor contraexemplo é a matriz de densidade de dimensão $2^3 \times 2^3$ dada por

$$\rho = \begin{bmatrix} \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} \end{bmatrix},$$

ou, na notação de Dirac,

$$\rho = \frac{1}{2} |000\rangle \langle 000| + \frac{1}{2} |111\rangle \langle 111|.$$

A distância de traço entre ρ e $\mathbb{1}/8$ é

$$\begin{aligned} \|\rho - \mathbb{1}/8\|_{\text{tr}} &= \frac{1}{2} \left(\left| \frac{1}{2} - \frac{1}{8} \right| + \left| 0 - \frac{1}{8} \right| + \left| 0 - \frac{1}{8} \right| + \left| 0 - \frac{1}{8} \right| + \left| 0 - \frac{1}{8} \right| + \left| 0 - \frac{1}{8} \right| + \left| 0 - \frac{1}{8} \right| + \left| \frac{1}{2} - \frac{1}{8} \right| \right) \\ &= \frac{3}{4} > \frac{2}{3}. \end{aligned}$$

A matriz de densidade reduzida para o primeiro qubit é

$$(\rho_{00} + \rho_{11} + \rho_{22} + \rho_{33}) |0\rangle \langle 0| + (\rho_{44} + \rho_{55} + \rho_{66} + \rho_{77}) |1\rangle \langle 1| = \frac{1}{2} |0\rangle \langle 0| + \frac{1}{2} |1\rangle \langle 1|;$$

a matriz de densidade reduzida para o segundo qubit é

$$(\rho_{00} + \rho_{11} + \rho_{44} + \rho_{55}) |0\rangle \langle 0| + (\rho_{22} + \rho_{33} + \rho_{66} + \rho_{77}) |1\rangle \langle 1| = \frac{1}{2} |0\rangle \langle 0| + \frac{1}{2} |1\rangle \langle 1|;$$

e a matriz de densidade reduzida para o terceiro qubit é

$$(\rho_{00} + \rho_{22} + \rho_{44} + \rho_{66}) |0\rangle \langle 0| + (\rho_{11} + \rho_{33} + \rho_{55} + \rho_{77}) |1\rangle \langle 1| = \frac{1}{2} |0\rangle \langle 0| + \frac{1}{2} |1\rangle \langle 1|.$$

Uma opção seria o Provedor enviar para o Verificador o procedimento necessário para reduzir o problema US a uma instância do problema BSDU, que pertence à classe BQP. Esse procedimento poderia ser a ordenação da diagonal principal da matriz de densidade, mas para a ordenação de uma lista não ordenada com N itens, até então, são necessárias $\Omega(N \log N)$ comparações quânticas binárias (Høyer et al., 2002).

Um outro caminho seria investigarmos a polarização do problema US e se este problema estaria em coQMA. Existem técnicas de polarização disponíveis para os problemas SD e SDU (problemas completos para SZK e NISZK), como podemos ver pelos Lemas 3.1 e 3.3. Contudo,

uma característica dessas técnicas é que, a medida que se aumenta a distância estatística entre as distribuições, também se aumenta o tamanho das distribuições, podendo com isso trazer restrições similares ao Teorema 3.2.

O mesmo não se pode afirmar quanto à classe CQP, visto que não se conhece uma representação de um circuito CQP por meio de uma sequência de variáveis aleatórias governada por uma distribuição de Markov. Isso leva às perguntas em aberto:

- existe um oráculo para o qual $UP \cap coUP$ não está contido em CQP? Ou,
- $UP \cap coUP$ está contido em CQP para todo oráculo?

Quanto aos limites superiores de naCQP e CQP sabe-se, de acordo com Aaronson et al. (2016) e Hiromasa et al. (2025), que

$$naCQP \subseteq CQP \subseteq BPP^{PP} \subseteq BQP^{PP} \subseteq CBQP \subseteq PSPACE .$$

Sendo a classe CBQP a classe de problemas resolvidos polinomialmente por algoritmos quânticos com a capacidade de clonar estados quânticos (Hiromasa et al., 2025). Aaronson et al. (2016) questionam se

- $CQP \subseteq P^{PP}$ ou ainda $CQP \subseteq PP$?

Para uma eventual demonstração de $CQP \subseteq PP$ seria necessária uma prova não relativizante, visto que existe um oráculo para o qual $SZK \not\subseteq PP$ (Bouland et al., 2019).

Considerando que a classe naCQP contém a classe SZK, podemos perguntar se QSZK está contida em superclasses (abaixo de PSPACE) de naCQP. Até onde sabemos isso não é conhecido, portanto, está em aberto a seguinte relação:

- $QSZK \subseteq CBQP$?

Estudamos a relação da versão de decisão para o problema HSP (dHSP) com classes de provas interativas de conhecimento zero, apresentando casos particulares de dHSP que estão na classe NIPZK₁. Para a classe NIPZK₁, também mostramos que esta admite um problema completo, que é uma versão restrita do problema de promessa completo para a classe NIPZK. As relações concernentes a NIPZK e demais classes podem ser vistas na Figura 6.2.

Um problema completo para a classe NISZK é o problema de distância estatística uniforme (SDU), em que precisamos comparar uma distribuição de probabilidade X com a distribuição uniforme U_n , ambas sobre todas as strings binárias com n bits, perguntando-nos o quão próximo a distância estatística $d_{VT}(X, U_n)$ está de 0 ou de 1. Um problema completo para a classe NIPZK₁ é o problema Uniforme-ou-Pequeno (US, de *Uniform-or-Small*), similar ao problema SDU, com a diferença que perguntamos se a distância estatística $d_{VT}(X, U_n)$ é igual a 0 ou se é próxima de 1. Apresentamos uma restrição de SDU em que as probabilidades das $2^n/2$ primeiras strings (sob a ordenação lexicográfica usual) nunca são menores que as das $2^n/2$ últimas. Chamamos esse problema de Distância Estatística entre Distribuição Biparticionada e Distribuição Uniforme (BSDU). Provamos que BSDU $\in$ BQP.

Conforme já visto no Capítulo 5, existe um oráculo para o qual $NIPZK_1^A \not\subseteq QMA^A$, contudo, não sabemos se existe um oráculo tal que $NIPZK_1^A \not\subseteq coQMA^A$. Podemos perguntar:

- $NIPZK_1 \subseteq coQMA$?
- $NIPZK_1 \subseteq coQMA$ para todo oráculo?
- Existe um oráculo para o qual $NIPZK_1 \subseteq coQMA$?

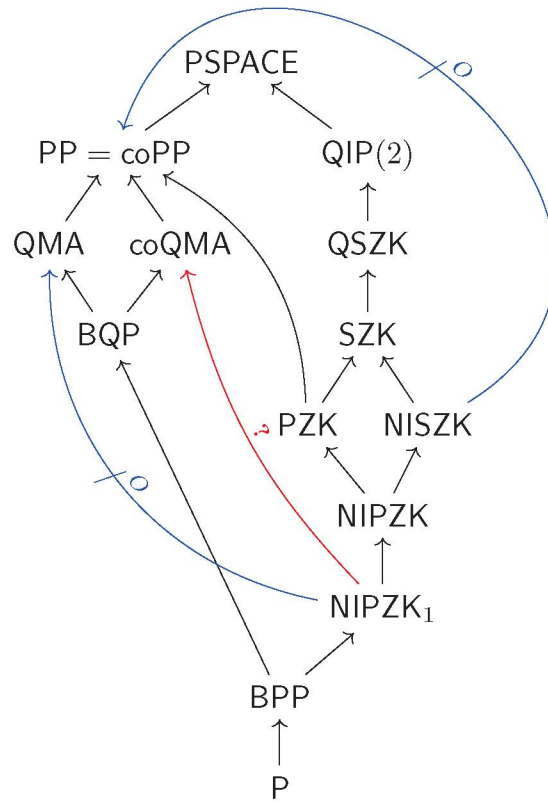


Figura 6.2: Relação entre as classes de conhecimento zero e as classes PP e QMA. As flechas pretas indicam inclusão entre as classes. As flechas com um risco indicam que existe um oráculo para o qual essa classe não está incluída na outra. A flecha vermelha com ponto de interrogação indica uma relação desconhecida.

Observamos no Capítulo 5 que provar $\text{NIPZK}_1 \subseteq \text{coQMA}$ não parece ser trivial. Os métodos até então obtidos para resolver um problema completo de NIPZK_1 necessitam de uma mensagem de tamanho exponencial por parte de um provador quântico.

Finalmente, gostaríamos de salientar a importância dos resultados obtidos nessa tese. Concernente a **naCQP** estudamos os limites da computação quântica na linha dos papers de Bennett et al. (1997) e Aaronson et al. (2016). Quando se mostra que BQP não contém NP relativo a um oráculo aleatório sugere-se que computadores quânticos não são capazes de resolver problemas NP -completos. Mostramos que mesmo que um computador quântico pudesse fazer medições sem colapso e mesmo com uma subclasse de NP as limitações no modelo quântico continuam válidas. Quanto aos resultados referentes a classes de conhecimento zero, mostramos casos particulares de dHSP que estão na classe NIPZK_1 ; apresentamos um problema, BSDU , que pode ser resolvido por um computador quântico; e mostramos que parece difícil resolver o problema US , completo para NIPZK_1 , usando o modelo coQMA , superclasse de BQP .

REFERÊNCIAS

- Aaronson, S. (2012). Impossibility of succinct quantum proofs for collision-freeness. *Quantum Information & Computation*, 12(1-2):21–28.
- Aaronson, S. (2016). $P \stackrel{?}{=} NP$, páginas 1–122. Springer International Publishing, Cham.
- Aaronson, S. (2018). $P_{DQP}/qpoly = ALL$. *Quantum Information & Computation*, 18(11-12):901–909.
- Aaronson, S., Bouland, A., Fitzsimons, J. e Lee, M. (2014). The space “just above” BQP. *Preprint*.
- Aaronson, S., Bouland, A., Fitzsimons, J. e Lee, M. (2016). The space “just above” BQP. Em *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, páginas 271–280.
- Aaronson, S., Grewal, S., Iyer, V., Marshall, S. C. e Ramachandran, R. (2024). $P_{DQMA} = DQMA = NEXP$: QMA with hidden variables and non-collapsing measurements.
- Aharonov, D. (2003). A simple proof that Toffoli and Hadamard are quantum universal. *arXiv preprint quant-ph/0301040*.
- Aiello, W. e Hastad, J. (1991). Statistical zero-knowledge languages can be recognized in two rounds. *Journal of Computer and System Sciences*, 42(3):327–345.
- Arora, S. e Barak, B. (2009). *Computational Complexity: a Modern Approach*. Cambridge University Press.
- 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.
- Baker, T., Gill, J. e Solovay, R. (1975). Relativizations of the $P \stackrel{?}{=} NP$ question. *SIAM Journal on Computing*, 4(4):431–442.
- Beigel, R., Buhrman, H. e Fortnow, L. (1998). NP might not be as easy as detecting unique solutions. Em *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, páginas 203–208.
- Bennett, C. H., Bernstein, E., Brassard, G. e Vazirani, U. (1997). Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523.
- Bennett, C. H. e Gill, J. (1981). Relative to a random oracle A , $P^A \neq NP^A \neq coNP^A$ with probability 1. *SIAM Journal on Computing*, 10(1):96–113.
- Bernstein, E. e Vazirani, U. (1997). Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473.
- Böhler, E., Glaßer, C. e Meister, D. (2006). Error-bounded probabilistic computations between MA and AM. *Journal of Computer and System Sciences*, 72(6):1043–1076.

- Boulund, A., Chen, L., Holden, D., Thaler, J. e Vasudevan, P. N. (2019). On the power of statistical zero knowledge. *SIAM Journal on Computing*, 49(4):FOCS17–1.
- Cobham, A. (1964). The intrinsic computational difficulty of functions. Em *Proc. 1964 Congress for Logic, Methodology, and the Philosophy of Science*, páginas 24–30. North-Holland.
- Cook, S. A. (1971). The complexity of theorem-proving procedures. Em *Proceedings of the third annual ACM symposium on Theory of computing*, páginas 151–158.
- 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.
- Costa, A. F. B. (2024). Um algoritmo quântico para casos não abelianos de HSP. Tese de mestrado, Universidade Federal do Paraná.
- Dasgupta, A. (2011). Mathematical foundations of randomness. Em Bandyopadhyay, P. e Forster, M., editores, *Philosophy of Statistics*, volume 7 de *Handbook of the Philosophy of Science*, páginas 641–710. Elsevier.
- Dixon, P., Gayen, S., Pavan, A. e Vinodchandran, N. (2020). Perfect zero knowledge: New upperbounds and relativized separations. Em *Theory of Cryptography Conference*, páginas 684–704. Springer.
- Edmonds, J. (1965). Paths, trees, and flowers. *Canadian Journal of mathematics*, 17:449–467.
- 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.
- Fortnow, L. (1987). The complexity of perfect zero-knowledge. Em *Proceedings of the nineteenth annual ACM symposium on Theory of computing*, páginas 204–209.
- Fortnow, L. e Rogers, J. (1999). Complexity limitations on quantum computation. *J. Comput. Syst. Sci.*, 59(2):240–252.
- Gill, J. (1977). Computational complexity of probabilistic turing machines. *SIAM J. Comput.*, 6(4):675–695.
- Goldreich, O. (2005). On promise problems: A survey. Em *Theoretical Computer Science: Essays in Memory of Shimon Even*, páginas 254–290. Springer.
- Goldreich, O. (2007). *Foundations of cryptography: volume 1, basic tools*. Cambridge university press.
- 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.
- Goldreich, O., Sahai, A. e Vadhan, S. (1999). Can statistical zero knowledge be made non-interactive? or On the relationship of SZK and NISZK. Em *Annual International Cryptology Conference*, páginas 467–484.
- Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. Em *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, páginas 212–219.

- Gurevich, Y. e Blass, A. (2022). Quantum circuits with classical channels and the principle of deferred measurements. *Theoretical Computer Science*, 920:21–32.
- 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.
- Hemaspaandra, L. e Ogihara, M. (2001). *The complexity theory companion*. Springer Science & Business Media.
- Hepp, H., Silva, M. V. G. e Zatesko, L. M. (2019). Uma revisão sobre a relação de BQP com outras classes de complexidade computacional. Em *Anais do III Workshop de Pesquisa em Computação dos Campos Gerais (WPCCG'19)*, volume 3, páginas 74–79.
- Hepp, H., Silva, M. V. G. e Zatesko, L. M. (2021). Um algoritmo quântico para um problema de distância estatística. Em *Anais do VI Encontro de Teoria da Computação*, páginas 1–4, Porto Alegre, RS, Brasil. SBC.
- Hepp, H., Silva, M. V. G. e Zatesko, L. M. (2025). Oracle separations for non-adaptive collapse-free quantum computing. *Theoretical Computer Science*, 1030:115078.
- Herstein, I. N. (1991). *Topics in algebra*. John Wiley & Sons.
- Hiromasa, R., Mizutani, A., Takeuchi, Y. e Tani, S. (2025). Rewindable quantum computation and its equivalence to cloning and adaptive postselection. *Theory of Computing Systems*, 69(1):1–37.
- Høyer, P., Neerbek, J. e Shi, Y. (2002). Quantum complexities of ordered searching, sorting, and element distinctness. *Algorithmica*, 34(4):429–448.
- Janzing, D. e Wocjan, P. (2007). A simple PromiseBQP-complete matrix problem. *Theory of computing*, 3(1):61–79.
- Jozsa, R. (2001). Quantum factoring, discrete logarithms, and the hidden subgroup problem. *Computing in science & engineering*, 3(2):34–43.
- Karp, R. M. (1972). Reducibility among combinatorial problems. Em Miller, R. E. e Thatcher, J. W., editores, *Complexity of Computer Computations*, The IBM Research Symposia Series, páginas 85–103. Plenum Press, New York.
- Kaye, P., Laflamme, R. e Mosca, M. (2007). *An Introduction to Quantum Computing*. Oxford University Press.
- Kobayashi, H. (2003). Non-interactive quantum perfect and statistical zero-knowledge. Em *International Symposium on Algorithms and Computation*, páginas 178–188.
- Ladner, R. E. (1975). On the structure of polynomial time reducibility. *J. ACM*, 22(1):155–171.
- Levin, L. A. (1973). Universal sequential search problems. *Problemy peredachi informatsii*, 9(3):115–116.
- 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.

- Malka, L. (2015). How to achieve perfect simulation and a complete problem for non-interactive perfect zero-knowledge. *Journal of Cryptology*, 28(3):533–550.
- Menda, S. e Watrous, J. (2018). Oracle separations for quantum statistical zero-knowledge. *Preprint*.
- Mitzenmacher, M. e Upfal, E. (2017). *Probability and computing: Randomization and probabilistic techniques in algorithms and data analysis*. Cambridge university press.
- Montanaro, A. e de Wolf, R. (2016). A survey of quantum property testing. *Theory of Computing*, (7):1–81.
- Nielsen, M. A. e Chuang, I. L. (2011). *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 10th edition.
- Okamoto, T. (1996). On relationships between statistical zero-knowledge proofs. Em *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, páginas 649–658.
- Papadimitriou, C. (1994). *Computational Complexity*. Addison-Wesley.
- Rabin, M. O. (1960). Degree of difficulty of computing a function and a partial ordering of recursive sets. *Technical Report 2*.
- Raz, R. e Tal, A. (2022). Oracle separation of BQP and PH. *Journal of the ACM*, 69(4).
- Sahai, A. e Vadhan, S. (2003). A complete problem for statistical zero knowledge. *Journal of the ACM (JACM)*, 50(2):196–249.
- Savage, J. E. (1997). *Models of Computation: Exploring the Power of Computing*. Addison-Wesley Longman Publishing Co., Inc.
- Savitch, W. J. (1970). Relationships between nondeterministic and deterministic tape complexities. *Journal of computer and system sciences*, 4(2):177–192.
- Sdroievski, N. M. (2018). Conhecimento zero estatístico e reduções eficientes para o problema MKTP. Tese de mestrado, Universidade Federal do Paraná.
- Sdroievski, N. M., da Silva, M. V. e Vignatti, A. L. (2019). The hidden subgroup problem and MKTP. *Theoretical Computer Science*, 795:204–212.
- Seress, Á. (2003). *Permutation group algorithms*. Número 152. Cambridge University Press.
- Shamir, A. (1992). $IP = PSPACE$. *Journal of the ACM (JACM)*, 39(4):869–877.
- Sherstov, A. A. e Thaler, J. (2022). Vanishing-error approximate degree and qma complexity. *Chicago journal of theoretical computer science*.
- 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.
- Stearns, R. E., Hartmanis, J. e Lewis, P. M. (1965). Hierarchies of memory limited computations. Em *6th annual symposium on switching circuit theory and logical design (SWCT 1965)*, páginas 179–190. IEEE.

- Tamon, C. e Yamakami, T. (2001). Quantum computation relative to oracles. Em *Unconventional Models of Computation, UMC'2K*, páginas 273–288. Springer.
- Vadhan, S. P. (1999). *A study of statistical zero-knowledge proofs*. Tese de doutorado, Massachusetts Institute of Technology.
- Valiant, L. G. (1976). Relative complexity of checking and evaluating. *Information Processing Letters*, 5(1):20–23.
- Vazirani, U. (2004). Lecture 10: NP-complete problems, lower bounds and Grover's algorithm. <https://people.eecs.berkeley.edu/~vazirani/f04quantum/notes/lec10/lec10.pdf>.
- Watrous, J. (2002). Limits on the power of quantum statistical zero-knowledge. Em *The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. Proceedings*, páginas 459–468. IEEE.
- Watrous, J. (2009). Quantum computational complexity. Em *Encyclopedia of complexity and systems science*, páginas 7174–7201. Springer.
- Yan, J. (2012). Complete problem for perfect zero-knowledge quantum proof. Em *International Conference on Current Trends in Theory and Practice of Computer Science*, páginas 419–430. Springer.