UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
CIÊNCIAS DA COMPUTAÇÃO

Gabriel Medeiros Lopes Carneiro

Estudo de Algoritmos Quânticos

Florianópolis, SC
2022

Gabriel Medeiros Lopes Carneiro

Estudo de Algoritmos Quânticos

Universidade Federal de Santa Catarina
Centro Tecnológico
Departamento de Informática e Estatística
Ciências da Computação

Orientador: Eduardo Inácio Duzzioni

Florianópolis, SC
2022

Resumo

A bolsa de iniciação científica teve como foco de estudo computação quântica. Durante o período, duas linguagens de programação quântica foram estudadas, sendo elas Qiskit e Ket. Além disso, os principais algoritmos quânticos foram vistos, como, por exemplo, o algoritmo de busca de Grover, a estimativa de fase, busca de ordem, entre outros. Com os conhecimentos adquiridos também foi possível participar de um projeto de extensão relacionado a um simulador quântico.

Lista de ilustrações

Sumário

1 Introdução

1.1 Motivação

Alguns problemas não possuem solução clássica em tempo polinomial, como a fatoração. Para vários desses, a computação quântica já se mostrou eficiente, inclusive para a fatoração, algo que pode comprometer a criptografia RSA.

1.2 Justificativas

A computação quântica ainda está crescendo, mas já mostra grande potencial para resolver problemas de otimização, logística, finanças, álgebra linear e vários outros. Grandes empresas já começaram a investir fortemente no setor, algo que faz aumentar a busca por profissionais na área. Então, estudar, entender e se adaptar ao novo modo de computação pode trazer grandes retornos num futuro relativamente próximo.

1.3 Objetivos

1.3.1 Objetivo Geral

Estudo de computação e algoritmos quânticos.

1.3.2 Objetivos Específicos

- Entender a base da computação quântica.
- Conhecer linguagens de programação quântica.
- Estudar e implementar algoritmos quânticos.

2 Computação Quântica

2.1 Breve Histórico

No início do século XX os cientistas enfrentavam um grande problema, que era explicar o comportamento da radiação emitida por um corpo negro. A solução desse problema levou ao surgimento da Mecânica Quântica, que segundo a hipótese de Max Planck "a radiação só pode ser emitida ou absorvida por um corpo negro em quantidades múltiplas inteiras de hf", em que $h \approx 6,62 \cdot 10^{-34} J \cdot s$ é a constante de Planck e f é a frequência de radiação. A quantização da energia e de outras grandezas na escala atômica foi importante para explicar uma série de outros fenômenos, como por exemplo o efeito fotoelétrico e o espectro da radiação emitida por átomos e moléculas. O desenvolvimento da Mecânica Quântica nos permitiu compreender melhor o comportamento da matéria na escala microscópica, nesse caso particular, os materiais semicondutores, que permitiram a criação do transistor. Os transistores substituíram as válvulas usadas nos primeiros computadores digitais a partir de 1955. É importante notar que a lógica usada para realizar as operações computacionais nos nossos notebooks, PCs, tablets e smartphones é a lógica booleana, ou seja, uma lógica clássica que envolve operações como AND, OR e NOT sobre os bits 0 e 1. Devido a sua grande capacidade de cálculo e armazenamento, os computadores são fundamentais para o desenvolvimento de qualquer sociedade moderna. Essa é a chamada primeira revolução quântica.

Um dos grandes impulsionadores da computação quântica foi o físico Richard Feynman, que no início da década de 80 sugeriu o uso de computadores quânticos para simular sistemas quânticos. A percepção de Feynman baseia-se no fato de que o número de configurações possíveis nos sistemas quânticos cresce de maneira exponencial com o número de entes (spins, elétrons, átomos, ...) considerados, tornando-se proibitivo para a memória dos computadores atuais guardar tanta informação mesmo para um número pequeno (< 100) de partículas.

Na década seguinte, os primeiros algoritmos quânticos começaram a surgir, dentre eles, os que mais se destacaram foram o algoritmo de busca de Grover e o de fatoração de Shor. Este último algoritmo foi provavelmente um dos grandes responsáveis pelo desenvolvimento da computação quântica, já que é capaz de encontrar os fatores primos p e q que multiplicados resultavam em um número inteiro $N = p \cdot q$ em uma escala de tempo que cresce polinomialmente com o tamanho do número N. Ou seja, a base do sistema de segurança RSA, amplamente usado para realizar transações bancárias no mundo todo, pode estar comprometida a partir da existência de computadores quânticos de larga escala.

A partir da segunda década do século XXI, não apenas a computação quântica, mas outras áreas como criptografia quântica, sensores quânticos e simulação quântica tem recebido forte atenção não apenas do setor acadêmico, mas também do setor industrial. Dessa forma, tem-se observado o rápido desenvolvimento das chamadas tecnologias quânticas, o que configura a segunda revolução quântica, uma vez que a lógica por trás dos processos é de natureza quântica.

2.2 Desenvolvimento Atual

A Computação Quântica ainda está em fase de amadurecimento, mas já mostra o seu grande potencial para resolver problemas práticos, além do algoritmo de fatoração de Shor e de busca de Grover, tais como problemas de otimização, machine learning, logística, química quântica, finanças, álgebra linear, entre outros. Nos últimos cinco anos, grandes empresas como Google, IBM, Amazon e Microsoft intensificaram ainda mais os seus investimentos no setor, além de vários governos de diversos países da América do Norte, Europa, Oceania e Ásia. Um dos grande temores em relação ao computadores quânticos está relacionado à segurança da informação, que afeta não apenas as transações bancárias, mas toda e qualquer transmissão de informação sigilosa, incluindo a militar, naturalmente. Tentativas de barrar possíveis ataques por computadores quânticos incluem a criptografia pós-quântica, que apesar do nome, se baseia em métodos criptográficos clássicos.

Para quem estiver interessado em aprender computação quântica, vale lembrar que algumas empresas disponibilizam computadores quânticos reais, simuladores, kits e linguagens para desenvolvimento de algoritmos ao público geral. Por exemplo, é possível acessar gratuitamente o kit de desenvolvimento quântico criado pela IBM (Qiskit) e rodar algoritmos em alguns dos seus computadores quânticos com poucos qubits. Através deste projeto será possível programar em um simular quântico de até 30 qubits de maneira gratuita, através da linguagem Ket e do simulador QuBOX.

2.3 O que é um qubit?

O qubit (**qu**antum + **bit**) é um bit quântico. O bit clássico sempre está em um dos possíveis estados 0 ou 1, já um qubit pode estar em ambas configurações simultaneamente. Chamamos esse fenômeno de superposição. Para representar um qubit utilizamos a notação Dirac ou "braket":

$$|\psi\rangle = \begin{bmatrix} a \\ b \end{bmatrix} = a\,|0\rangle + b\,|1\rangle$$

em que a e b são amplitudes de probabilidade (números complexos), de modo que

$|a|^2$ representa a probabilidade de após uma medida encontrar o sistema no estado $|0\rangle$

$|b|^2$ representa a probabilidade de após uma medida encontrar o sistema no estado $|1\rangle$

Como a probabilidade total deve somar 100%, temos que a condição de normalização para o estado $|\psi\rangle$ é $|a|^2 + |b|^2 = 1$.

2.3.1 Esfera de Bloch

Os estados de um qubit podem ser representados por meio de pontos em uma superfície esférica de raio unitário, utilizando o sistema de coordenadas esféricas. Para isso, é preciso parametrizar o estado do qubit $|\psi\rangle = a\,|0\rangle + b\,|1\rangle$ da seguinte forma

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle \text{ tal que } \theta \in [0,\pi], \phi \in [0,2\pi)$$

Agora, utilizando θ e ϕ no sistemas de coordenadas esféricas, tem-se a Esfera de Bloch. Todos os estados acessíveis a um qubit podem ser representados utilizando-se Figura 1.

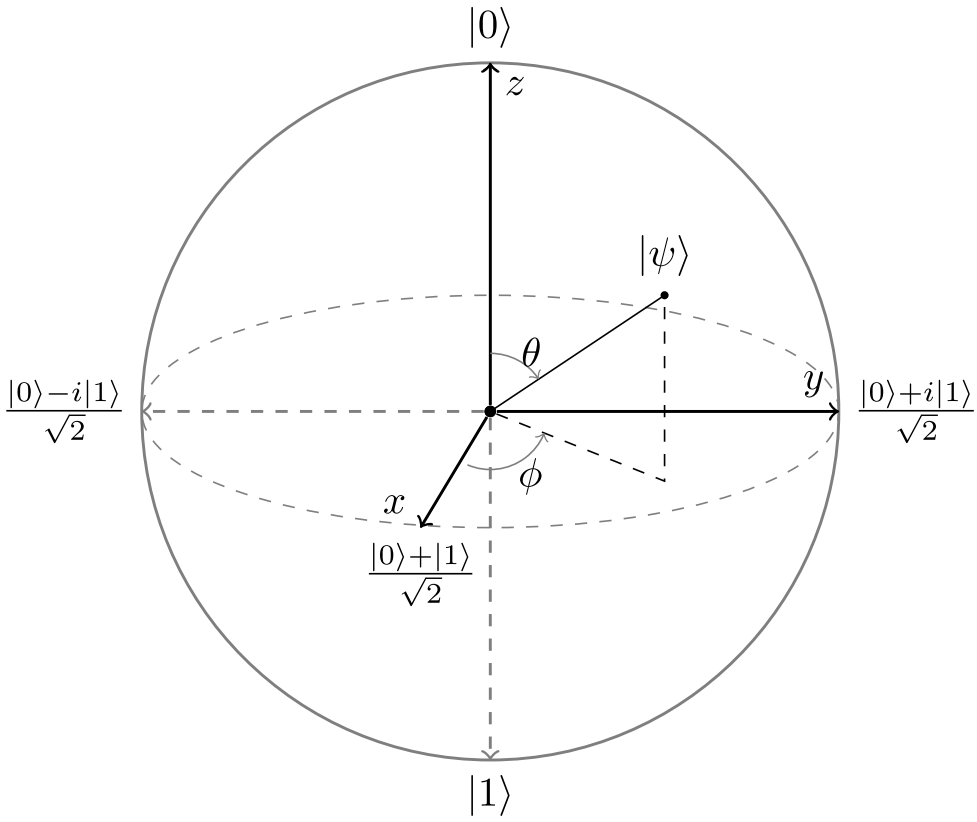

Figura 1 – Representação de um qubit na Esfera de Bloch.

2.3.2 Representação de 2 ou mais qubits

Existem diversas formas de se representar um sistema de 2 qubits, seguem algumas equivalências:

$$|\psi_0\rangle \otimes |\psi_1\rangle = |\psi_0\rangle\,|\psi_1\rangle = |\psi_0\psi_1\rangle$$

em que $\otimes$ é produto tensorial de ψ_0 com ψ_1. Seja

$$|\psi_0\rangle \otimes |\psi_1\rangle = \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} \otimes \begin{bmatrix} b_0 \\ b_1 \end{bmatrix} = \begin{bmatrix} a_0 b_0 \\ a_0 b_1 \\ a_1 b_0 \\ a_1 b_1 \end{bmatrix}$$

De forma análoga, é possível representar sistemas de n qubits como

$$|\psi_0\rangle \otimes |\psi_1\rangle \otimes \cdots \otimes |\psi_n\rangle = |\psi_0\rangle\,|\psi_1\rangle \ldots |\psi_n\rangle = |\psi_0\psi_1 \ldots \psi_n\rangle$$

Como será mostrado na seção 2.7, a superposição de estados desse tipo pode levar ao emaranhamento.

2.4 Etapas de um Algoritmo Quântico

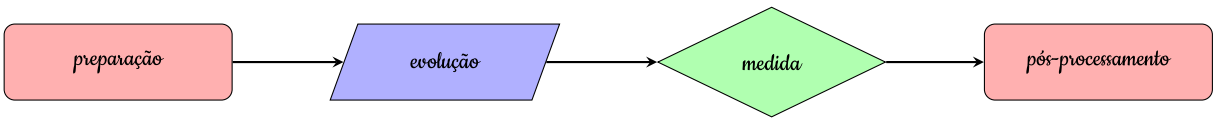

Figura 2 – Etapas básicas de um algoritmo quântico

De forma geral, é possível separar um algoritmo quântico em quatro etapas, como mostra a Figura 2.

1. **Preparação**: aqui cada qubit é inicializado em algum estado, geralmente em $|0\rangle$.
2. **Evolução**: nessa parte o algoritmo é de fato aplicado, através das portas lógicas quânticas.
3. **Medida**: após a aplicação das portas, é necessário medir os qubits, para se ter o resultado do circuito.
4. **Pós-processamento**: finalmente, nessa etapa o resultado obtido deve ser interpretado de acordo com o contexto.

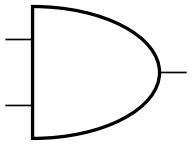

Figura 3 – Representação da porta AND.

2.5 Comparação com Computação Clássica

2.5.1 Entradas e Saídas

- **Clássica**: portas podem ter diferentes números de bits entrando e saindo.

Exemplo
A porta AND possui dois ou mais bits de entrada e apenas um de saída.

- **Quântica**: portas possuem mesmo número de qubits na entrada e na saída.

2.5.2 Reversibilidade

- **Clássica**: a maioria das portas clássicas não são reversíveis, isto é, dado uma saída não conseguimos identificar quais foram as entradas.

Exemplo
Na porta OR de dois bits podemos obter 1 como saída em três casos.

X	Y	X OR Y
0	0	0
0	1	1
1	0	1
1	1	1

Sabendo que a saída foi 1 não é possível identificar qual/quais bits eram 1.

- **Quântica**: seus circuitos são reversíveis, isso ocorre, pois, seus operadores são unitários.

Observação
Embora a evolução temporal seja reversível durante o processamento da informação no circuito quântico, a medição dos qubits é um processo irreversível.

2.6 Portas Lógicas Quânticas

As portas lógicas quânticas são operações unitárias que ao atuar em um estado inicial levam para outro estado final, ou seja, funcionam como rotações na esfera de Bloch. A seguir, alguns exemplos de portas lógicas quânticas que atuam sobre um qubit.

2.6.1 Porta X

Essa porta é o equivalente a porta NOT da computação clássica.
Matriz

$$X = \sigma_x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Comportamento

$$\begin{aligned} X|0\rangle &= |1\rangle \\ X|1\rangle &= |0\rangle \end{aligned}$$

Símbolo

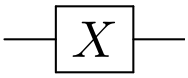

Figura 4 – Representação da porta X.

ou ainda

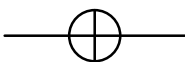

Figura 5 – Outra representação da porta X.

2.6.2 Porta Y

Matriz

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

Comportamento

$$\begin{aligned} Y|0\rangle &= i|1\rangle \\ Y|1\rangle &= -i|0\rangle \end{aligned}$$

Símbolo

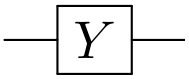

Figura 6 – Representação da porta Y.

2.6.3 Porta Z

A porta Z introduz uma fase relativa de π entre os estados da base computacional.
Matriz

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

Comportamento

$$\begin{aligned} Z\left|0\right\rangle &= \left|0\right\rangle \\ Z\left|1\right\rangle &= -\left|1\right\rangle \end{aligned}$$

Símbolo

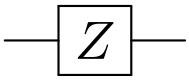

Figura 7 – Representação da porta Z.

2.6.4 Porta Hadamard

Essa porta gera uma superposição dos estados da base computacional.
Matriz

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

Comportamento

$$\begin{aligned} H\left|0\right\rangle &= \frac{1}{\sqrt{2}}\left(\left|0\right\rangle + \left|1\right\rangle\right) &= \left|+\right\rangle \\ H\left|1\right\rangle &= \frac{1}{\sqrt{2}}\left(\left|0\right\rangle - \left|1\right\rangle\right) &= \left|-\right\rangle \end{aligned}$$

Símbolo

Figura 8 – Representação da porta H.

2.6.5 Portas Controladas

Para se fazer computação quântica universal, ou seja, realizar todas as transformações unitárias desejadas entre os qubits de entrada e saída em um algoritmo, é necessário realizar operações que façam dois ou mais qubits interagirem entre si. Tais portas podem envolver um qubit de controle e o outro como alvo, sendo possível generalizá-la para múltiplos qubits de controle e de alvo. Segue o exemplo para porta controlada X, ou CNOT, com um controle e um alvo.

Matriz

$$\text{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Comportamento

$$\begin{aligned} \text{CNOT}\,|00\rangle &= |00\rangle \\ \text{CNOT}\,|01\rangle &= |01\rangle \\ \text{CNOT}\,|10\rangle &= |11\rangle \\ \text{CNOT}\,|11\rangle &= |10\rangle \end{aligned}$$

Símbolo

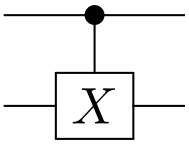

Figura 9 – Representação da porta X-controlada.

ou ainda

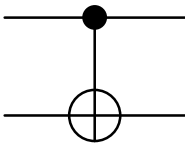

Figura 10 – Outra representação da porta X-controlada.

2.7 Emaranhamento

Estados emaranhados são aqueles que não podem ser escritos como produto tensorial de estados de 1 qubit, ou seja, não é possível separá-los. Os mais conhecidos são os estados de Bell, os quais envolvem apenas 2 qubits, sendo dados por:

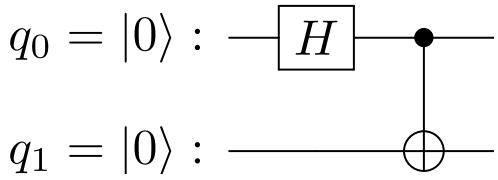

Figura 11 – Circuito para criar um estado de Bell.

$$
\begin{aligned}
|\beta_{00}\rangle &= |\Phi^+\rangle &&= \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right) \\
|\beta_{01}\rangle &= |\Phi^-\rangle &&= \frac{1}{\sqrt{2}}\left(|00\rangle - |11\rangle\right) \\
|\beta_{10}\rangle &= |\Psi^+\rangle &&= \frac{1}{\sqrt{2}}\left(|01\rangle + |10\rangle\right) \\
|\beta_{11}\rangle &= |\Psi^-\rangle &&= \frac{1}{\sqrt{2}}\left(|01\rangle - |10\rangle\right)
\end{aligned}
$$

Os estados emaranhados são apontados como sendo os responsáveis por fazer não apenas a computação quântica mais veloz do que a computação clássica, mas também permitem aumentar a precisão de medidas de observáveis físicos e realizar comunicação de forma segura.

2.7.1 Criando um Estado de Bell

Já vimos o que é o emaranhamento, agora vamos criá-lo. Como exemplo, criaremos o estado $|\beta_{00}\rangle$, na Figura 11 temos o circuito para isso e em 1 o código para o mesmo usando Ket.

```
q0, q1 = quant(2)    # cria dois qubits
H(q0)                # aplica a porta de Hadamard no qubit 0
ctrl(q0, X, q1)      # aplica a porta X no qubit 1, com o qubit 0 como controle
```

Listing 1: Criando um estado de Bell em Ket.

Seja $|\psi\rangle = q_0 \otimes q_1$. Após a aplicação da porta de Hadamard, teremos $q_0 = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right)$, conforme visto anteriormente. Logo,

$$
\begin{aligned}
|\psi\rangle &= \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) \otimes |0\rangle \\
&= \frac{1}{\sqrt{2}}\left(|00\rangle + |10\rangle\right)
\end{aligned}
$$

Na sequência, temos uma porta CNOT, com o qubit 0 como controle e o qubit 1 como alvo. Gerando a seguinte situação

$$
\begin{aligned}
|\psi\rangle &= \mathrm{CNOT}\left[\frac{1}{\sqrt{2}}\left(|00\rangle + |10\rangle\right)\right] \\
&= \frac{1}{\sqrt{2}}\left(\mathrm{CNOT}\,|00\rangle + \mathrm{CNOT}\,|10\rangle\right) \\
&= \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle\right) \\
&= |\beta_{00}\rangle
\end{aligned}
$$

Portanto, com apenas duas portas é possível gerar uma situação de emaranhamento.

Referências