Máquina de Turing Quântica

Elloá B. Guedes[1,2,3] Bernardo Lula Jr.[2,3]
{elloa,lula}@dsc.ufcg.edu.br

[1] Integrante do Grupo PET Computação da UFCG
[2] IQuanta - Instituto de Estudos em Computação e Informação Quânticas
[3] DSC - Departamento de Sistemas e Computação - UFCG

Palavras-chave: *Computabilidade, Computação Quântica, Máquina de Turing Quântica*

1 Introdução

Turing, escrevendo antes da existência dos modernos computadores digitais, estava interessado em saber o que de fato significava alguma tarefa ser computável. Intuitivamente, podemos definir uma tarefa como computável se é possível determinar um conjunto finito de passos que levarão à realização da mesma. A partir deste questionamento, foi capaz de propor um dispositivo, a chamada Máquina de Turing, que levou à uma noção formal de computação [Copeland 2004].

As Máquinas de Turing (MT) foram propostas em 1936 e são idealizações de um modelo matemático de computação que pode ser usado para compreender os limites do que os computadores podem fazer [Turing 1936, Hopcroft 1984]. Podem ser descritas como dispositivos que manipulam símbolos e que podem ser adaptados para simular a lógica de qualquer computador que já tenha sido construído ou, de forma equivalente, como sendo "*uma máquina dotada de um conjunto finito de estados associados à uma forma de armazenamento externo ou meio de memória*"[Minsky 1967].

A figura 1 ilustra uma máquina de Turing descrita em alto nível. Podemos ver a ilustração da fita, a qual possui os valores a serem utilizados como entrada, neste caso, um alfabeto binário (0 ou 1) e o registrador do estado atual, indicando que a máquina está no estado Q_a. Observe que a luz verde acesa indica que ela está em um estado de aceitação.

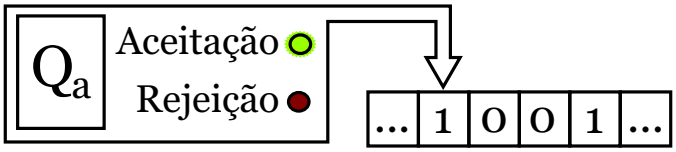

Figura 1. *Exemplo de uma máquina de Turing a partir da sua descrição em alto nível.*

Em 1980, Benioff [Benioff 1980]verificou a possibilidade da existência de uma máquina de Turing reversível, ou seja, conhecendo o estado em que essa MT se encontra em um dado momento, é possível afirmar quais são todos os seus estados futuros e também quais foram os seus estado passados [Willians and Clearwater 1998].

Posteriormente, Feynman [Feynman 1982] provou que nenhuma MT Clássica poderia simular alguns fenômenos quânticos sem que houvesse um aumento muito grande (exponencial) no número de estados clássicos nesta simulação, mas que um "simulador quântico universal" poderia fazê-lo sem sofrer tais conseqüências .

Decorrente das questões e possibilidades levantadas nestes dois trabalhos, Deutsch [Deutsch 1985] concebeu a primeira Máquina de Turing Quântica, a qual levava em consideração propriedades da Mecânica Quântica na realização da computação . A Máquina de Turing Quântica (MTQ) proposta seria capaz de ler, escrever e realizar operações de deslocamento de acordo com interações descritas pela Mecânica Quântica e cuja fita poderia existir em estados não-clássicos [Willians and Clearwater 1998], isto é, na MTQ a computação é, por definição, reversível e esta também consegue simular fenômenos quânticos de forma eficiente.

Este trabalho possui por objetivo analisar de modo mais aprofundado a proposição de Deutsch, através das definições de Máquina de Turing Quântica e da classe de complexidade que abrange os problemas tratáveis neste modelo de computação.

2 Máquina de Turing Quântica

Para que seja possível o entendimento do conceito de Máquina de Turing Quântica, é ideal ilustrar, informalmente, como se caracteriza uma computação neste modelo a fim de facilitar a apresentação da sua definição formal, visando tornar possível a melhor compreensão deste conceito.

2.1 Descrição Informal de uma Árvore de Computação em uma MTQ

Seja uma árvore de computação definida como todas as possibilidades de estados que uma MT qualquer pode assumir em uma computação para uma dada entrada. Cada nodo nesta árvore corresponde a um estado da máquina e cada nível da árvore corresponde a um passo na computação. A raiz dessa árvore representa o estado inicial e todos os demais nodos correspondem a diferentes configurações atingíveis com probabilidade não-nula associada proveniente de seu nodo pai.

Em uma MTQ, consideramos que cada aresta, do nodo-pai em direção ao nodo-filho, é uma amplitude associada de que a computação estando no nodo-pai siga em direção ao nodo-filho. A amplitude de um nodo é simplesmente o produto das amplitudes das arestas de um caminho da raiz até aquele nodo. A amplitude de uma configuração particular a cada passo da computação é simplesmente a soma da amplitude de todos os nodos correspondentes àquela configuração, no nível da árvore correspondente àquele passo. Agora, a probabilidade de uma configuração de um dado passo é a raiz quadrada de sua amplitude.

Na figura 2 está representada uma árvore de computação de uma MTQ, onde Q_0 representa o estado inicial da Máquina e os pares ordenados (x, a) onde $x \in \{0, 1\}$ é uma letra do alfabeto e a é a amplitude da transição indicada. A amplitude do estado Q_7, por exemplo, é o produto da amplitude da transição de Q_0 para Q_3 e a amplitude de Q_3 para Q_7.

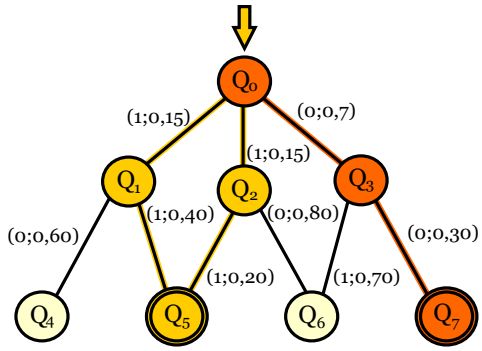

Figura 2. *Exemplo de uma árvore de computação para uma Máquina de Turing Quântica*

Apesar da MTQ utilizar o conceito de amplitude em cada aresta da árvore de computação, é fundamental compreender a diferença a cerca da MTQ em relação à MT Probabilística, que reside no fato de que dois caminhos diferentes podem interferir tanto construtivamente (se as amplitudes são iguais), quanto destrutivamente, se suas amplitudes são opostas [Bernstein and Vazirani 1993], o que não ocorre na MT Probabilística nem nas demais MTs Clássicas. Vale citar também que a fita da MTQ pode conter estados em superposição.

2.2 Definição Formal da Máquina de Turing Quântica

A definição formal de uma MTQ não é um conceito trivial de ser compreendido mas, somente com a definição formal é que é possível obter uma especificação precisa do que um computador quântico pode ou não computar.

Uma Máquina de Turing Quântica é definida como uma 6-tupla ordenada $(\Sigma, \Lambda, Q, q_i, q_f, \delta)$, onde:

1. Σ é um conjunto finito denominado "alfabeto" de todos os possíveis símbolos da fita;
2. $\Lambda \in \Sigma$ é o símbolo denominado "branco";
3. Q é um conjunto finito, denominado "conjunto de estados";
4. q_i é o estado inicial;
5. q_f é o estado final;
6. $\delta : \Sigma \times Q \rightarrow H$ é a função de transição e H é um espaço de Hilbert estendido por vetores da base correspondentes à tripla de $\Sigma \times Q \times \{L, R\}$, e as correspondentes transições finitas da matriz unitária para todos os comprimentos de entrada [Deutsch 1985].

Para estudos formais de complexidade, é comum utilizar outras versões da QTM. Algumas dessas definições são encontradas na literatura: [Paajanen 1993] e [Hirvensalo 2001], por exemplo, definem da QTM de forma análoga à MT Probabilística, substituindo probabilidades por transições de amplitudes. Por outro lado, [Pavicic 2006] [Yamakami 1999] definem tais máquinas como matrizes de transição. Tais modelos diferentes, porém equivalentes, são aceitos pois possuem características particulares que auxiliam principalmente no estudo da Complexidade [Knill and Nielsen 2002].

3 Classe BQP

Problemas que podem ser resolvidos em tempo polinomial são denominados "tratáveis" e aqueles que não podem ser resolvidos em tempo polinomial são denominados "intratáveis" [Toscani and Veloso 2001]. Vale salientar que o tempo polinomial é definido em termos da quantidade de passos que um modelo de computação utiliza para resolver o problema.

Considerando então a MTQ, é possível definir a classe BQP como uma classe de complexidade quântica que consiste dos problemas de decisão que podem ser resolvidos com uma dada probabilidade de erro usando uma Máquina de Turing Quântica de tempo polinomial [Nielsen and Chuang 2005]. Mais precisamente, se uma linguagem $L \in BQP$ existe uma família de circuitos de tamanho polinomial que decidem a linguagem, aceitando palavras da linguagem com probabilidade de pelo menos $\frac{3}{4}$, e rejeitando palavras que não pertencem a linguagem com probabilidade de pelo menos $\frac{3}{4}$.

3.1 Relações da classe BQP com outras classes de complexidade

Embora algumas questões relacionadas às relações entre estas classes de complexidade ainda não estejam solucionadas, atualmente a relação da classe BQP com as classes anteriormente citadas se dá segundo a figura 3. É fácil visualizar que P $\subseteq$ BPP $\subseteq$ BQP $\subseteq$ PSPACE e que P $\subseteq$ NP $\subseteq$ PSPACE. Atualmente ainda não se provou que se estes subconjuntos são próprios mas acredita-se que P $\neq$ NP, NP $\neq$ PSPACE e ainda que BPP $\neq$ BQP [Kaye et al. 2007].

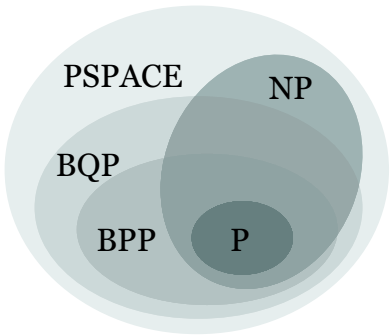

Figura 3. *Relação entre a classe BQP e as classes de complexidade P, NP, BPP e PSPACE*

4 Considerações Finais

O presente trabalho visou apresentar os conceitos de Máquina de Turing Quântica e a classe de complexidade que abrange os problemas tratáveis para este tipo de modelo de computação de uma forma didática, voltada para alunos de graduação em Ciência da Computação. Portanto, a relevância do mesmo consiste em introduzir este conteúdo de forma esclarecedora e minimalista, propiciando não só o aprendizado dos mesmos, mas também a divulgação e o estímulo à pesquisa em Computação Quântica.

Este trabalho foi desenvolvido em parceria com o IQuanta - Instituto de Estudos em Computação e Informação Quânticas, localizado na Universidade Federal de Campina Grande.

Referências

[Benioff 1980] Benioff, P. (1980). The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computer as represented by turing machines. *Journal of Statistical Physics*, 22:495–507.

[Bernstein and Vazirani 1993] Bernstein, E. and Vazirani, U. (1993). Quantum complexity theory. *Proc. 25th Annual ACM Symposium on the Theory of Computing*, ACM Press, New York:11–20.

[Copeland 2004] Copeland, B. J. (2004). *The Essential Turing: Seminal Writings in Computing, Logic, Philosophy, Artificial Intelligence, and Artificial Life: Plus The Secrets of Enigma*. Oxford University Press.

[Deutsch 1985] Deutsch, D. (1985). Quantum theory, the church-turing principle and the universal quantum computer. *Proceedings of the Royal Society of London A*, 400:pp. 97–117.

[Feynman 1982] Feynman, R. (1982). Simulating physics with computers. *International Journal of Theoretical Physics*, 21:467–788.

[Hirvensalo 2001] Hirvensalo, M. (2001). *Quantum Computing*. Berlin.

[Hopcroft 1984] Hopcroft, J. (1984). Turing machines. *Scientific American*, 1:86–98.

[Kaye et al. 2007] Kaye, P., Laflamme, R., and Mosca, M. (2007). *An Introduction to Quantum Computing*. Oxford University.

[Knill and Nielsen 2002] Knill, E. and Nielsen, M. A. (2002). Theory of quantum computation. Supplement III, Encyclopaedia of Mathematics.

[Minsky 1967] Minsky, M. (1967). *Computation: Finite and Infinite Machines*. Prentice-Hall.

[Nielsen and Chuang 2005] Nielsen, M. A. and Chuang, I. L. (2005). *Computação Quântica e Informação Quântica*. Bookman.

[Paajanen 1993] Paajanen, S.-L. (1993). Quantum turing machines. Seminar on Quantum Computing, Department of Computer Science, University of Helsinki.

[Pavicic 2006] Pavicic, M. (2006). *Quantum Computation and Quantum Communication: Theory and Experiments*. New York.

[Toscani and Veloso 2001] Toscani, L. V. and Veloso, P. A. S. (2001). *Complexidade de Algoritmos*. Sagra-Luzzato.

[Turing 1936] Turing, A. (1936). On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, Series 2, 42:230–265.

[Willians and Clearwater 1998] Willians, C. P. and Clearwater, S. H. (1998). *Explorations in Quantum Computing*. The Eletronic Library of Science.

[Yamakami 1999] Yamakami, T. (1999). A foundation of programming a multi-tape quantum turing machine. Department of Computer Science, Princeton University (1999).Disponível em http://www.arxiv.org/quant-ph/9906084.