Simulação do Algoritmo de Grover

Rafael Ferreira Lago
PESC, COPPE-UFRJ
21941-972, Rio de Janeiro, RJ
E-mail: rafael@cos.ufrj.br

Luiz Mariano de Carvalho
Departamento de Matemática Aplicada, IME-UERJ
20559-900, Rio de Janeiro, RJ
E-mail: luizmc@gmail.com

Carlile Lavor
Departamento de Matemática Aplicada, IMECC-UNICAMP
13081-970, Campinas, SP
E-mail: clavor@ime.unicamp.br

Nelson Maculan
PESC, COPPE-UFRJ
21941-972, Rio de Janeiro, RJ
E-mail: maculan@cos.ufrj.br

1 Introdução

Uma das principais características dos transistores é a sua capacidade de aumento de velocidade à medida que reduz-se o seu custo e seu tamanho. Os transistores de hoje em dia ocupam menos de 1% da área que os transistores de 20 anos atrás ocupavam. Embora alguns estudiosos tenham previsto um limite para a redução do tamanho dos transistores, estes limites foram quebrados, e os transistores continuam diminuindo de tamanho hoje em dia.

Com a redução do tamanho dos circuitos de um microprocessador, possivelmente chegaremos à um circuito onde as leis da física clássica falhariam, e onde entraria em vigor as leis da física quântica. Tais leis criam certos fenômenos que parecem inconcebíveis no mundo clássico, mas que, entretanto, se bem estudados e trabalhados podem trazer benefícios e avanços à computação.

2 Conceitos Básicos

Para fazer uso das propriedades quânticas na computação, fez-se necessária uma redefinição da forma de representação da informação. Ao invés de utilizarmos bits, utilizamos os bits quânticos - q-bits - que são representados por $|0\rangle$ e o $|1\rangle$. Estes q-bits também podem ser representados da seguinte forma:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad e \qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Os q-bits $|0\rangle$ e $|1\rangle$ podem ser interpretados como uma base de um espaço vetorial, mais precisamente, uma base ortogonal de dimensão dois dos números complexos, que é $\mathbb{C}^2$. Chamamos esta base de *base computacional*. Enunciaremos agora o primeiro postulado da mecânica quântica:

Postulado 1 *Existe, para cada sistema físico isolado, um espaço vetorial complexo com produto interno (ou seja, um espaço de Hilbert), conhecido como espaço de estados do sistema. O sistema é completamente descrito por seu vetor estado, que é um vetor unitário no espaço de estados do sistema.[2]*

Tendo em mente o postulado acima, seja $|\psi\rangle \in \mathbb{C}^2$ o estado atual do sistema. Como

$|0\rangle$ e $|1\rangle$ formam uma base ortonormal de $\mathbb{C}^2$, podemos escrever um estado $|\psi\rangle$ genérico como uma combinação linear destes dois q-bits:

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

onde

$$|\alpha|^2 + |\beta|^2 = 1, \qquad \alpha, \beta \in \mathbb{C} \tag{2}$$

Disto podemos concluir, por exemplo, que $|0\rangle$ pode ser compreendido como:

$$|0\rangle = 1 \times |0\rangle + 0 \times |1\rangle.$$

Os valores α e β são chamados de *amplitudes*, e estão diretamente relacionados à *medição*, que veremos mais a frente. Enunciemos agora outro postulado:

Postulado 2 *A evolução de $|\psi\rangle$ no tempo é dada pela equação de Schrödinger*

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

com $\hbar = h/2\pi$, onde h é a constante de Plank.

Em [4], podemos ver que a equação de Schrödinger é linear para t, portanto, dado um valor inicial $|\psi(t_0)\rangle$, é possível calcular qualquer $|\psi(t)\rangle$, que está unicamente definido.

Em [2], temos que a equação de Schrödinger pode ser escrita como:

$$|\psi(t)\rangle = U(t, t_0)|\psi(t_0)\rangle, \tag{3}$$

sendo $U(t, t_0)$ um operador unitário. Assim, temos que a evolução de um sistema quântico no tempo é dado por uma aplicação de um operador unitário sobre um vetor estado.

Para representar um estado com mais de um q-bit, temos o seguinte postulado:

Postulado 3 *O espaço de estados de um sistema físico composto é o produto tensorial dos espaços de estado dos sistemas que o compõem.*

Vamos exemplificar um espaço de dois q-bits. Seja E_2 o espaço de estados de um sistema de dois q-bits e E_1 o espaço de estados de um sistema de um q-bit. Podemos entender um sistema de dois q-bits como um sistema composto de dois sistemas de um q-bit. Então, segundo o postulado, o espaço de estados de E_2 é o produto tensorial de E_1 por ele mesmo.

3 Propriedades da Mecânica Quântica

Dizemos que um estado quântico está emaranhado quando é impossível representá-lo como produto tensorial de outros dois estados de um q-bit.

Para uma melhor exemplificação do problema, tomemos os seguintes estados, $|\varphi_1\rangle$ e $|\varphi_2\rangle$:

$$|\varphi_1\rangle = \begin{bmatrix} \frac{1}{2} \\ 0 \\ 0 \\ \frac{\sqrt{3}}{2} \end{bmatrix} \quad e \quad |\varphi_2\rangle = \begin{bmatrix} 0 \\ 0 \\ \frac{\sqrt{3}}{3} \\ \frac{\sqrt{6}}{3} \end{bmatrix} \tag{4}$$

Ambos tratam-se de estados válidos de um sistema quântico de dois q-bits. Suponha que precisamos conhecer o valor do segundo q-bit apenas. Tendo em mente o postulado 3, podemos imaginar tais estados como o produto tensorial de outros dois estados. Para $|\varphi_2\rangle$, temos:

$$|\varphi_2\rangle = \begin{bmatrix} \frac{\sqrt{3}}{3} \\ \frac{\sqrt{6}}{3} \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Assim, temos que o valor do segundo q-bit é $|1\rangle$; entretanto é impossível separarmos o estado $|\varphi_1\rangle$ em dois estados de um q-bit. Neste caso, chamamos $|\varphi_1\rangle$ de um *estado emaranhado*. A única forma de obtermos informação do estado $|\varphi_1\rangle$, definido no exemplo 4, é o submetendo a uma medição. Ao tentarmos medir um estado quântico dizemos que há um *colapso*, forçando o estado do sistema quântico para um dos estados da base de estados do sistema. Por exemplo, o estado $|\psi\rangle$ definido em (1), ao ser medido, seria colapsado para $|0\rangle$ ou $|1\rangle$, com probabilidade igual a $|\alpha|^2$ e $|\beta|^2$, respectivamente. Ao tentarmos medir o estado $|\varphi_1\rangle$, teríamos probabilidade de $\frac{1}{4}$ de obter o estado[1] $|00\rangle$ e probabilidade $\frac{3}{4}$ de obter o estado $|11\rangle$. Em outras palavras, embora tenhamos uma quantidade infinita de informação, só podemos medir dois valores. Há uma perda de informação ao efetuarmos esta medição, e só poderíamos obter uma medição completa da informação, caso preparássemos o mesmo estado infinitas vezes e o testássemos infinitas vezes.

[1] usaremos a notação $|\psi\varphi\rangle$ para indicar o estado composto pelos estados $|\psi\rangle$ e $|\varphi\rangle$. Para mais informações, consultar [1]

Para evitar erros de medição, uma solução básica, à primeira vista, seria preparar o mesmo estado várias vezes e medir várias vezes a saída, até que a probabilidade de erro seja suficientemente baixa para que o dado seja considerado seguro. Porém, este excesso de medidas diminuiria o desempenho, fazendo com que o desempenho dos algoritmos quânticos talvez se igualassem, ou mesmo fossem inferiores ao dos algoritmos clássicos. Mais a frente veremos métodos para aumentar a probabilidade de medirmos corretamente um estado.

Um importante princípio ainda não mencionado é o princípio da superposição:

Princípio da Superposição de Ondas: *Se duas ondas oriundas de uma mesma fonte atingirem uma mesma tela, suas amplitudes (e não suas intensidades) são somadas algebricamente*

Este princípio não é um fato exclusivo da mecânica quântica. Ele pode ser observado em ondas de várias naturezas (ver figuras 1 e 2).

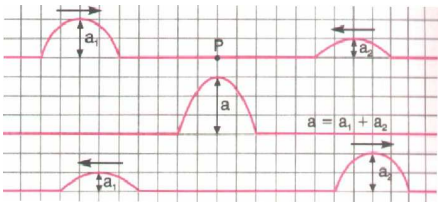

Figura 1: Superposição de ondas de amplitudes com sinais iguais

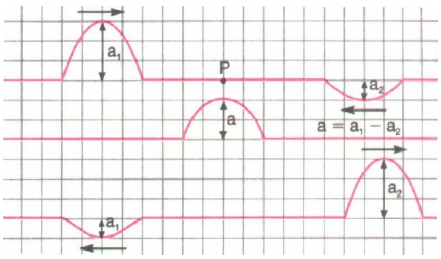

Figura 2: Superposição de ondas de amplitudes com sinais opostos

O que aconteceria se fôssemos capazes de aplicar uma operação nas ondas no momento em que elas estão superpostas? (representado em ambas figuras na linha do meio) É exatamente esta propriedade que chamamos de *paralelismo quântico*. Na computação clássica, para testar o resultado de uma mesma operação em N estados distintos, precisamos efetuar a operação N vezes. Na computação quântica, podemos superpor todos estes estados e aplicar a operação uma única vez e obtermos o resultado. Uma outra visualização do paralelismo quântico pode ser feita tendo em mente uma árvore de dados. Para percorrer a árvore, temos que escolher qual nó filho iremos seguir. Num computador quântico, podemos seguir todos os nós filhos de uma única vez, graças à superposição.

4 Funcionamento do Algoritmo de Grover

Trata-se de um algoritmo quântico que tem por finalidade encontrar um determinado elemento i_0 em uma lista L não ordenada, com N elementos, sendo N uma potência de 2. Para tanto, tomemos um número n natural como o número de q-bits. Assim, definimos $N = 2^n$.

Utilizaremos dois registradores para este algoritmo. O primeiro registrador terá n q-bits, e o segundo registrador terá um q-bit apenas. A idéia básica é testar cada elemento da lista. Caso seja igual ao i_0 procurado, então marcamos o segundo registrador. No final, pegamos o elemento cujo segundo registrador foi marcado e temos o elemento procurado (supomos que haverá apenas um elemento igual a i_0 na lista). Do Postulado 2 e da Equação (3), temos que é possível escrever esta operação (marcação do elemento procurado) por um operador unitário (em função de i_0), que chamaremos de U_f. Se representarmos cada elemento de L como um vetor, então buscar i_0 nada mais é do que aplicar um operador unitário sobre um vetor.

Entretanto, esta idéia em nada supera o algoritmo clássico. Isto porque no algoritmo clássico o que fazemos é exatamente verificar a lista elemento a elemento até encontrar o elemento procurado. O que faremos é superpor todos os elementos da lista L em um único estado, que chamaremos de $|\psi\rangle$. Ao aplicar o operador unitário U_f, estaremos fazendo uso do paralelismo quântico, ou seja, é como se estivéssemos buscando o elemento i_0 em todos os elementos da lista de uma única vez. Assim, teremos o elemento procurado marcado em apenas um passo, embora esta informação esteja no nível quântico.

5 Formulação Matemática

Partiremos agora para uma breve descrição matemática do problema. Seja n o número de q-bits (logo, o primeiro registrador terá n q-bits), e $|i\rangle$ o estado de n q-bits associado ao i-ésimo elemento da lista L. Precisamos superpor todos os elementos de L num único estado que chamamos de $|\psi\rangle$. De (1), temos que esta superposição pode ser obtida com

$$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle. \tag{5}$$

Prepararemos o segundo registrador de uma maneira especial. Ele receberá o valor de $|1\rangle$ e lhe será aplicado o operador de Hadamard:

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

O operador de Hadamard é um operador unitário de um qbit largamente utilizado na computação quântica. Aplicando o operador de Hadamard sobre o estado $|0\rangle$, obtemos uma superposição dos estados $|0\rangle$ e $|1\rangle$ com igual amplitude. Mas o aplicando sobre $|1\rangle$ obtemos uma superposição de amplitudes opostas, embora ainda obedeçam à (2) :

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

onde $|-\rangle$ é o segundo registrador. Veremos mais a frente o porquê desta decisão com relação ao segundo registrador.

Agora vejamos o operador U_f, que muitas vezes é chamado de "função oráculo", pois simplesmente descobre o elemento procurado e o marca. A aplicação de U_f sobre um $|i\rangle$ (supondo que o segundo registrador seja $|0\rangle$) é:

$$U_f(|i\rangle|0\rangle) = \begin{cases} |i\rangle|1\rangle, & se \quad i = i_0 \\ |i\rangle|0\rangle, & se \quad i \neq i_0 \end{cases}. \tag{6}$$

A situação é análoga caso o segundo registrador seja $|1\rangle$.

Agora já podemos entender melhor o porquê da decisão que tomamos durante a criação do segundo registrador. Ao aplicar o operador unitário U_f sobre os ois registradores, temos o seguinte resultado:

$$\begin{aligned} U_f(|\psi\rangle|-\rangle) &= \frac{1}{\sqrt{N}} \left(\left(\sum_{i=0, i\neq i_0}^{N-1} |i\rangle|-\rangle \right) - |i_0\rangle|-\rangle \right) \\ &= \frac{1}{\sqrt{N}} \left(\left(\sum_{i=0, i\neq i_0}^{N-1} |i\rangle \right) - |i_0\rangle \right) |-\rangle. \end{aligned} \tag{7}$$

Ou seja, o elemento procurado, e somente ele, tem a sua amplitude alterada para $-\frac{1}{\sqrt{N}}$. Embora o segundo registrador não tenha seu valor alterado, ele é de fundamental importância para o funcionamento do algoritmo.[2]

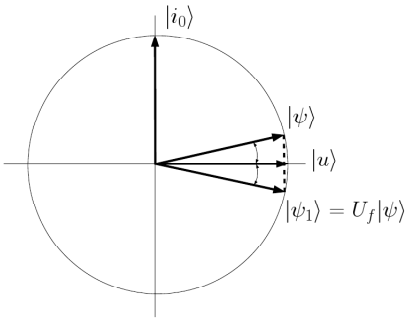

Figura 3: Efeito do operador U_f sobre o estado $|\psi\rangle$

Na figura 3, vemos uma forma de representar a aplicação do operador U_f sobre o $|\psi\rangle$, gerando o estado que chamaremos daqui pra frente de $|\psi_1\rangle$. O comprimento do vetor na figura está associado à soma do quadrado das amplitudes, que como vimos em (2), será sempre igual a 1. O ângulo formado está relacionado à probabilidade de obtermos $|i_0\rangle$ através de uma medição; quanto mais próximo de $\pm 90°$, maior a probabilidade.

Sabemos que todas as probabilidades são iguais após a aplicação de U_f, portanto, após uma medição de $|\psi_1\rangle$, teremos a probabilidade $\frac{1}{N}$ de encontrar o elemento que procuramos, o que é uma baixíssima probabilidade. Para aumentá-la, aplicamos então um *operador de reflexão*, que reflete o vetor $|\psi_1\rangle$ com relação a $|\psi\rangle$, gerando um vetor que chamaremos de $|\psi_G\rangle$.

Este operador de reflexão é dado por

$$2|\psi\rangle\langle\psi| - I, \tag{8}$$

[2]Para mais detalhes sobre este fato, consulte [3]

onde I é o operador identidade e $\langle\psi|$ é o dual[3] de $|\psi\rangle$.

O operador de reflexão aumenta a probabilidade do elemento marcado pelo U_f, refletindo-o para cada vez mais perto de i_0, conseqüentemente afastando-o cada vez mais dos outros estados, pois:

$$|i_0\rangle = 0|0\rangle + 0|1\rangle + ... + 1|i_0\rangle + ... + 0|N-1\rangle.$$

Após esta aplicação, a probabilidade de $|i_0\rangle$ ser medido é aumentada de $\frac{1}{N}$ para $\frac{3N-4}{N\sqrt{N}}$, e os demais estados dividem a mesma probabilidade de serem obtidos através de uma medição.

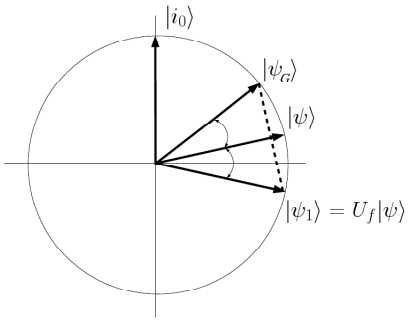

Figura 4: Efeito do operador de reflexao sobre o estado $|\psi_1\rangle$

Assim, embora haja uma melhoria considerável na probabilidade após a aplicação do operador de reflexão, para um número muito grande de q-bits, ela ainda é muito pequena. Podemos aplicar estes operadores várias vezes com objetivo de torná-lo cada vez mais próximo de i_0. O operador composto pelos dois operadores apresentados (U_f e o operador de reflexão) é chamado de *operador de Grover*, e o número de vezes que ele pode ser aplicado a fim de aproximar cada vez mais o estado atual de $|i_0\rangle$ é dado por:

$$k = \frac{arcos\left(\frac{1}{\sqrt{N}}\right)}{arcos\left(\frac{N-2}{N}\right)}. \qquad (9)$$

Aplicando o operador de reflexão, estamos deslocando $|\psi\rangle$ em θ graus. Após a k-ésima aplicação do operador de Grover, a aplicação do mesmo estaria na verdade afastando o estado $|\psi\rangle$ de $|i_0\rangle$.

[3] O estado $|\varphi\rangle^\dagger = (|\varphi\rangle^*)^T$, é chamado de dual de $|\varphi\rangle$

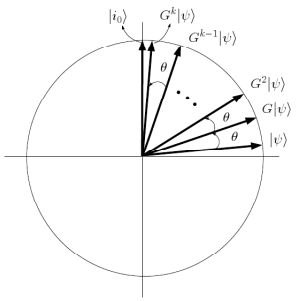

Figura 5: Efeito do operador de reflexao sobre o estado $|\psi_1\rangle$

Um fato que merece ser mencionado é que o ângulo θ está relacionado ao número de q-bits. Quanto maior a quantidade de q-bits, menor será o ângulo θ, e conseqüentemente, serão necessárias mais aplicações do operador de grover para aproximar o estado $|\psi\rangle$ de $|i_0\rangle$ (basta notar que a equação (9) é crescente). Entretanto, quão menor for θ melhor será a aproximação obtida após todas as aplicações do operador de Grover.

6 Implementação em FORTRAN

Implementamos uma simulação do algoritmo em Fortran95 utilizando o compilador Intel(R) Fortran 10.0.23. A princípio, o nosso objetivo foi atingir uma simulação para pelo menos 32 q-bits, e assim, traçarmos um paralelo com os computadores clássicos atuais.

Estávamos trabalhando com um número muito grande de operações, como multiplicações entre $|\psi\rangle|-\rangle$ e U_f, o que gerou porblemas de armazenamento e processamento. O operador U_f pode ser representado da seguinte forma:

$$U_f = \begin{bmatrix} 1 & & & & \\ & \ddots & & & \\ & & NOT & & \\ & & & \ddots & \\ & & & & 1 \end{bmatrix},$$

com NOT definida por

$$NOT = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

A porta NOT é situada na submatriz $U_f(2i_0+1, 2i_0+1)$, $U_f(2i_0+2, 2i_0+1)$, $U_f(2i_0+$

$1, 2i_0 + 2)$, $U_f(2i_0 + 2, 2i_0 + 2)$. É importante lembrar que U_f é uma matriz $2N \times 2N$, e que para representar um sistema de dezesseis q-bits, por exemplo, temos $N = 2^{16} = 65536$.

Outro empecilho foi a matriz de reflexão, que tem tamanho $N \times N$. Esta matriz tem o formato a seguir:

$$2\times\begin{bmatrix} \alpha^2 - \frac{1}{2} & \alpha^2 & \alpha^2 & ... & \alpha^2 \\ \alpha^2 & \alpha^2 - \frac{1}{2} & \alpha^2 & ... & \alpha^2 \\ \alpha^2 & \alpha^2 & \alpha^2 - \frac{1}{2} & ... & \alpha^2 \\ \vdots & & & \ddots & \vdots \\ \alpha^2 & \alpha^2 & \alpha^2 & ... & \alpha^2 - \frac{1}{2} \end{bmatrix}.$$

Além disto, percebemos que a cada aplicação do operador de reflexão, temos o valor do elemento procurado alterado, e o restante dos valores são alterados todos para o mesmo valor. Percebemos que estes cálculos poderiam ser simplificados, pois eram repetidos muitas vezes.

O terceiro problema é o produto tensorial entre vetores, que tende a ser muito grande. Por exemplo, utilizando 16 q-bits, $|\psi\rangle$ terá 2^{16} entradas. Ao executar o produto tensorial entre $|\psi\rangle$ e $|-\rangle$ temos um loop de 2^{17} iterações.

Com todas estas restrições, conseguimos um máximo de 10 q-bits, gerando "Falha de Segmentação"ao tentar extrapolar este limite, devido ao uso excessivo de memória para armazenar tais estruturas e efetuar seus cálculos. Para atingir o objetivo inicial, criamos uma nova rotina de aplicação do algoritmo (mas mantivemos a primeira rotina), e aplicamos métodos diferentes para resolver o mesmo problema.

Percebemos que o produto final da aplicação do algoritmo poderia ser reduzido a apenas dois valores: a probabilidade de encontrar o elemento procurado após uma medição, e a probabilidade de encontrar qualquer um dos outros elementos. Como os valores finais são sempre esses dois, decidimos por tratar o vetor estado sistema como se fosse apenas um vetor comum de duas posições, e tratar o elemento procurado como sendo sempre o elemento de número 1.

Sejam $|\chi\rangle$ um estado de um sistema quântico válido, β a amplitude do elemento procurado em $|\chi\rangle$ e α a amplitude dos demais elementos de $|\chi\rangle$. Consideremos estes dois valores (α e β) como entradas para um vetor que idealizamos, que chamaremos de χ_* (veja a equação (10)).

$$|\chi\rangle = \begin{bmatrix} \alpha \\ \vdots \\ \beta \\ \alpha \\ \vdots \end{bmatrix} \quad e \quad \chi_* = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}. \tag{10}$$

No primeiro passo do algoritmo, geramos uma superposição dos estados dos registradores. Para simular esta parte do algorimo, basta fazermos $\alpha = \beta = \frac{1}{\sqrt{N}}$.

O segundo passo do algoritmo consiste da marcação do elemento procurado. Porém, esta marcação é feita da mesma forma para qualquer quantidade de q-bits: altera o sinal do elemento procurado, os demais valores permanecem os mesmos. Assim, tudo que precisamos fazer é aplicar o operador U_f padrão para um q-bit sobre o vetor χ_*, considerando $|i_0\rangle$ como sendo 1.

O terceiro passo é aplicar o operador de reflexão. Este operador, conforme vimos em (8), está relacionado a $|\chi\rangle$, portanto, ao número de q-bits. Para tratar o problema, a forma de construção do operador de reflexão teve de ser alterada.

Nosso objetivo é criar um operador R_*, que seja capaz de alterar os valores de χ_* exatamente da forma que o operador $2|\chi\rangle\langle\chi| - \mathrm{I}$ faria ao estado $|\chi_1\rangle$. O resultado final da nossa operação, que chamaremos de χ_{G*}, deve ser equivalente ao $|\chi_G\rangle$, ou seja, o segundo elemento de χ_{G*} deve ser igual à amplitude do elemento procurado de $|\chi_G\rangle$, e o primeiro elemento de χ_{G*} deve ser igual às demais amplitudes de $|\chi_G\rangle$. Seja α_t e β_t as amplitudes atuais de um estado qualquer de $|\chi_G\rangle$ e do estado que queremos obter de $|\chi_G\rangle$, respectivamente.

Efetuando os cálculos[4], obtivemos a seguinte matriz:

$$R_* = \frac{1}{N}\begin{bmatrix} N-2 & 2 \\ 2(N-1) & 2-N \end{bmatrix}.$$

É importante notar que o operador R_* não é unitário, e que o vetor χ_* não é um estado válido de um sistema quântico (não normalizado), desobedecendo o postulado 2. Entretanto, nosso operador R_* e o vetor χ_* estão

[4]Para uma descrição completa de tais cálculos, consulte [1]

apenas simulando a operação, não se tratando de um sistema quântico real.

Outras alteração que fizemos foi reduzir o operador U_f à aplicação de NOT apenas no pedaço de $|\psi\rangle$ que é alterado. Assim, eliminamos a multiplicação matricial mais pesada da aplicação, e a transformamos em uma multiplicação de um vetor 1×2 por uma matriz 2×2. Já o Produto Tensorial deixou de ser um problema, pois passamos a trabalhar apenas com vetores de dimensão dois e quatro, que são pequenos o suficiente para o nosso objetivo. Apesar destas alterações, o desempenho ainda não havia se mostrado muito satisfatório, pois ainda contávamos com uma saída muito detalhada. Para obter a versão final do programa, retiramos as saídas que consideramos desnecessárias, como a exibição dos vetores e das matrizes. A única saída que permaneceu, e que foi considerada a saída chave do programa, foi a probabilidade de encontrar o elemento procurado, que é exibido a cada iteração do algoritmo de Grover.

7 Resultados

Executamos o nosso algoritmo em uma máquina Pentium 4, 3.00GHz, com 1GB de memória, rodando Debian Linux. Antes das simplificações no algoritmo, conforme foi dito, conseguimos um máximo de 10 q-bits, gerando "Falha de Segmentação"ao extrapolar este limite, devido ao uso excessivo de memória para armazenar as estrutudas de dados.

Para alguns valores de N, não foi nem mesmo possível executar o cálculo. Para n = 31, por exemplo, temos N = 2.147.483.648, mas este número não é suportado por uma variável inteira comum de Fortran95. Para fazermos testes com valores de n grandes, precisamos utilizar uma facilidade do Fortran, que nos permite alocar uma variável num espaço maior que o de costume. Uma variável do tipo INTEGER no Intel Fortran Compiler utiliza 4 bytes, mas podemos declará-la como INTEGER*2 para 2 bytes ou como INTEGER*8 para utilizar 8 bytes, e assim, aumentar a sua capacidade. Feito isto, conseguimos armazenar valores de N para n até 62. Além disso os arredondamentos gerados eram muit ruins, o que nos forçou a utilizar a mesma técnica para melhorar a precisão de pontos flutuantes. No caso, declaramos as variáveis reais para utilizarem 16 bits, a quantidade máxima permitida pelo Intel Fortran Compiler.

Deste modo, conseguimos executar o algoritmo satisfatoriamente para 62 q-bits. Após 1.686.629.713 aplicações do operador de Grover, obtivemos a probabilidade 1.000000 de encontrar o elemento procurado. Para valores acima de 62 q-bits, N possuia um valor maior que um INTEGER*8 consegue suportar e infelizmente, o Intel Fortran Compiler não nos fornece uma forma de armazenar inteiros com mais de 8 bytes.

Referências

[1] R. Lago, "Computação Quântica e Algoritmo de Grover, uma implementação clássica", Projeto Final de Graduação, IME-UERJ, 2008.

[2] M. A. Nielsen and I. L. Chuang, "Quantum Computation and Quantum Information", Cambridge University Press, Cambridge, 2000.

[3] R. Portugal, C. Lavor, L. Mariano e N. Maculan. , "Uma Introdução à Computação Quântica", Editora SBMAC, 2004.

[4] G. Strini, G. Casati and G. Benenti, "Principles of Quantum Computation and Information Volume I: Basic Concepts", World Scientific, 2004.