

AVALIANDO A SIMULAÇÃO DE TRANSFORMAÇÕES QUÂNTICAS A PARTIR DE PROCESSOS QUÂNTICOS NO VPE-qGM

ADRIANO MARON¹; RENATA REISER¹; MAURÍCIO PILLA¹

¹Programa de Pós Graduação e Ciência da Computação
Universidade Federal de Pelotas
{akmaron,reiser,pilla}@inf.ufpel.edu.br

1. INTRODUÇÃO

A Computação Quântica (CQ) (NIELSEN; CHUANG, 2003) fundamenta o desenvolvimento de algoritmos quânticos, os quais são capazes de solucionar problemas complexos exigindo menos recursos de tempo e memória.

Na CQ, o *qubit* é a unidade básica de informação quântica, vetorialmente representado por $(a,b)^t$, sendo a e b amplitudes associadas aos estados básicos $|0\rangle$ e $|1\rangle$ do *qubit*. Sistemas compostos por muitos *qubits* tem seu espaço de estados gerado a partir do produto tensorial dos espaços de estados de cada *qubit*. Para um sistema quântico de 2 *qubits*, tem-se o vetor $(a,b,c,d)^t$, com cada amplitude associada aos estados $|00\rangle$, $|01\rangle$, $|10\rangle$ e $|11\rangle$, respectivamente.

A evolução do estado é feita por transformações quânticas, definidas por matrizes quadradas ortonormalizadas de ordem 2^Q , sendo Q a quantidade de *qubits* do sistema. Assim, o próximo estado do sistema é obtido por um produto matriz-vetor de complexidade exponencialmente maior que seu tamanho.

O ambiente VPE-qGM (*Visual Programming Environment for the qGM Model*) (MARON et. al, 2011) está em desenvolvimento para suporte a modelagem e simulação distribuída de algoritmos da CQ, a partir de abstrações do modelo qGM (REISER; BURLAMAQUI, 2010).

Considerando o alto custo computacional inerente à simulação de algoritmos quânticos em computadores clássicos, a principal contribuição deste trabalho consiste na **extensão das capacidades de simulação do VPE-qGM pela implementação dos conceitos de Processos Quânticos (QPs) e Processos Quânticos Parciais (QPPs)**. De acordo com as especificações do modelo qGM, estes novos conceitos podem ser utilizados para modelar, de forma mais eficiente, transformações quânticas (controladas ou não).

2. MATERIAL E MÉTODOS

A interpretação de QPPs se dá a partir da aplicação parcial de uma transformação quântica. Considere a aplicação da porta H^2 ao estado clássico $|01\rangle$, conforme

$$\begin{aligned}
 |\Phi_1\rangle &= H \otimes H |\Phi_0^{01}\rangle \\
 &= \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \otimes \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \\
 &= \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 \\ -1 \\ 1 \\ -1 \end{pmatrix}
 \end{aligned}$$

Figura 1: Definição matricial de H^2

Cada conjunto unitário nesta construção pode interpretar um QPP, correspondendo na notação matricial, a uma matriz com apenas uma linha definida, sendo as demais desconhecidas (\square). Considerando como contexto os valores (0 ou 1) do primeiro *qubit*, tem-se os estados parciais descritos abaixo.

$$|\Phi_1^{0,x}\rangle_{\perp} = H_{\perp} \otimes H |\Phi_0\rangle = \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \perp & \perp \end{pmatrix} \otimes \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \\ \frac{-1}{2} \\ \perp \\ \perp \end{pmatrix}$$

$$|\Phi_1^{1,x}\rangle_{\perp} = H_{\perp} \otimes H |\Phi_0\rangle = \begin{pmatrix} \perp & \perp \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \otimes \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \perp \\ \perp \\ \frac{1}{2} \\ \frac{-1}{2} \end{pmatrix}$$

Os estados parciais acima são aproximações do estado gerado na Figura 1. Essas interpretações são realizadas, neste trabalho, a partir de QPPs, conforme ilustrado nas Figuras 2 e 3, exemplificando casos de transformações não

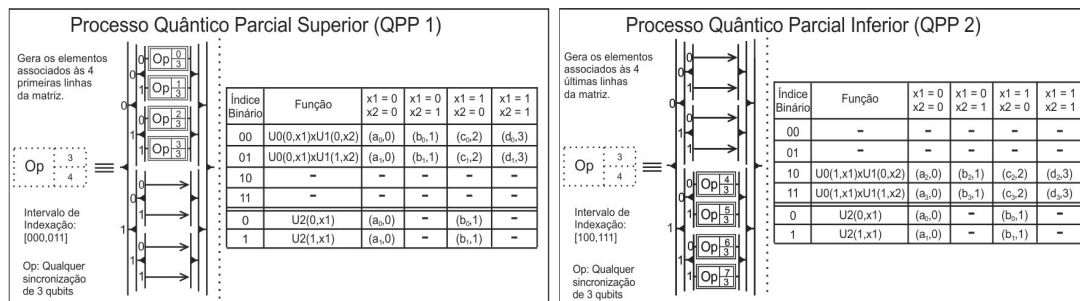


Figura 2: QPP que gera as amplitudes no intervalo [0,3] controladas de 3 qubits.

Figura 3: QPP que gera as amplitudes no intervalo [4,7]

Interpretações de transformações quânticas controladas a partir de QPPs também são contempladas por este trabalho. O conjunto de todos os QPPs que devem ser executados para uma determinada transformação quântica é chamado de *QPP_Set*. Entretanto, dependendo da configuração de transformação quântica a ser simulada, apenas um subconjunto desses QPPs (*QPP_Subset*) deve ser executado.

Genericamente, tem-se que $|QPP_Set| = |Exp| = 2^{nC}$, sendo nC a quantidade total de *qubits* de controle de todas as transformações quânticas modeladas. Entretanto, é necessário criar/executar apenas os QPPs pertencentes a um subconjunto (*QPP_Subset*) de *QPP_Set*. Se considerada apenas uma transformação controlada, tem-se que $|QPP_Subset| = 1$. Quando consideradas sincronizações de transformações controladas, tem-se $|QPP_Subset| = (2^{nC_1} + 2^{nC_2} + \dots + 2^{nC_k}) - 1$, sendo nC_k a quantidade de *qubits* de controle da k -ésima transformação. Quando da sincronização de transformações controladas com transformações não controladas (diferente de *Id*), tem-se que $QPP_Subset = QPP_Set$.

Após a criação dos QPPs, tem-se a aplicação de um operador recursivo que atua sobre as matrizes para cálculo das amplitudes no novo estado do sistema quântico. Essa função realiza uma combinação entre diferentes tuplas para geração dos valores correspondentes à matriz de definição da transformação

quântica aplicada. Além disso, é gerado um valor que indexa a amplitude do estado atual do sistema que deve ser multiplicada pelo valor gerado.

3. RESULTADOS E DISCUSSÕES

Para validação e análise de desempenho da simulação de algoritmos quânticos a partir de QPs e QPPs no VPE-qGM, dois estudos de caso foram considerados. O primeiro contempla *benchmarks*, compostos fundamentalmente por transformações controladas, selecionados dentre os disponíveis em (MASLOV; SCOTT, 2011). Essa escolha é justificada por dois pontos principais:

- Disponibilização de código fonte para geração de todos os algoritmos;
- Descrição de algoritmos quânticos com muitos *qubits* e dezenas/centenas de operadores.

O segundo considera transformações quânticas *Hadamard* de até 14 *qubits* ($H^{\square 11}$, $H^{\square 12}$, $H^{\square 13}$ e $H^{\square 14}$), validando, assim, as otimizações para operadores quânticos não controlados.

A metodologia dos testes utilizada contempla, para cada estudo de caso, a realização de 10 simulações. A máquina utilizada na simulação possui as seguintes características principais: processador Intel Core i5-2410M @ 2.3 GHz, 4GB RAM e sistema operacional Ubuntu 11.10 64 *bits*.

Foram monitorados o tempo de execução de cada amostra e o correspondente consumo de memória. A principal comparação de desempenho se dá com a versão anterior da biblioteca *qGM-Analyzer*, a qual suporta a simulação de algoritmos quânticos a partir de *PEs*, considerando as descrições introduzidas em (MARON et. al, 2011). As principais características de cada algoritmo, juntamente com resultados obtidos pelas correspondentes simulações, são sumarizados na Tabela 1.

Tabela 1: Algoritmos quânticos simulados.

Algoritmo	Qubits	Transformações	Simulação com QPs		Simulação com PEs	
			Tempo (s)	Mem. (MB)	Tempo (s)	Mem. (MB)
9symd2	12	28	0,400	12	38,805	12
gf2 ⁴	12	19	0,236	12	26,352	12
gf2 ⁵	15	29	1,328	13	372,488	13
gf2 ⁶	18	41	11,264	24	5081,553	22
gf2 ⁷	21	55	135,013	92	NS	NS
gf2 ⁸	24	85	1532,015	524	NS	NS
ham15_1	15	132	6,872	13	1778,170	13
ham15_2	15	70	4,708	13	925,319	13
ham15_3	15	109	8,436	13	1400,632	13
mod1024adder	20	55	44,099	60	NS	NS
rc_adder	16	19	2,358	15	549,079	14
H ^{□11}	11	11	6,816	12	7,882	12
H ^{□12}	12	12	25,292	12	28,281	12
H ^{□13}	13	13	97,401	12	111,572	12
H ^{□14}	14	14	384,923	12	496,934	12

Como pode ser observado nos resultados descritos na Tabela 1, algoritmos quânticos com até 24 *qubits* foram simulados. O consumo de memória se mantém similar para ambas as bibliotecas. Como as otimizações propiciadas pelos QPs e QPPs contemplam apenas as transformações quânticas que modelam o algoritmo, a maior parte do consumo de memória se dá pela criação dos vetores que armazenam as amplitudes dos estados. A simulação de transformações quânticas controladas descritas a partir de QPPs apresentaram uma melhora de aproximadamente 99% frente a representação por *PEs*, uma vez que ocorre a execução de apenas um subconjunto do total de QPPs que modelam a transformação quântica. Considerando transformações *Hadamard*, não foi obtido um ganho proporcional aos *benchmarks* constituídos de transformações controladas, uma vez que, nesse caso, todas as amplitudes do espaço de estados são modificadas. Entretanto, a redução de até 29% no tempo de simulação se justifica pela geração de todos os elementos a partir de um único QP, enquanto que na modelagem por *PEs*, tem-se a execução de 2^q componentes distintos, resultando em um número maior de operações.

4. CONCLUSÕES

A possibilidade de descrição de transformações quânticas (controladas ou não), introduzida neste trabalho, reduz o número de operações necessárias para realizar a evolução de estado do sistema quântico. Dessa forma, o incremento no desempenho discutido neste trabalho viabilizou a simulação de algoritmos com até 24 *qubits*. Apesar dos ganhos apresentados neste trabalho, o tempo de execução para determinadas configurações de transformações quânticas, como a *Hadamard* permanece elevada e sofrendo do aumento exponencial no tempo necessário para simulação. Nesse sentido, considera-se na continuidade do trabalho, o desenvolvimento de extensões para integração da biblioteca com a plataforma CUDA (NVIDIA, 09), possibilitando a execução paralela das computações associadas aos QPs e QPPs.

5. REFERÊNCIAS BIBLIOGRÁFICAS

Maron, A; Ávila, A; Reiser, R; Pilla, M. Introduzindo uma nova abordagem para simulação quântica com baixa complexidade espacial. **Anais do DINCON 2011**. SBMAC. p. 1-6, 2011.

Maslov, D., Scott, N. Reversible logic synthesis benchmarks page. 2011, disponível por WWW em <<http://www.cs.uvic.ca/dmaslov>>(apr.2012).

Nielsen, M; Chuang, I. **Computação Quântica e Informação Quântica**. Bookman, 2003.

NVIDIA, **CUDA Programming Guide**. NVIDIA Corp., 2009.

Reiser, R; Amaral, R. The quantum states space in the qGM Model. **Anais III Workshop Escola de Computação e Informação Quântica**. Editora do LNCC. p. 92-101, 2010.