

VPE-qGM: Uma Proposta de Simulação do Paralelismo Quântico Integrada à Programação Visual

Adriano Maron

Renata Reiser

Eduarda Monteiro

Universidade Católica de Pelotas

96010-000, Pelotas, RS

E-mail: {maron,reiser,eduardam}@ucpel.tche.br

RESUMO

O Projeto D-GM (*Distributed Geometric Machine*) provê suporte ao desenvolvimento de aplicações científicas fundamentadas no modelo GM, considerando uma execução paralela e/ou distribuída baseada em ambientes de computação em grade. No contexto deste trabalho, considera-se a extensão quântica do modelo GM (*qGM- Quantum Geometric Machine Model*) [1, 7] como fundamentação para construção do protótipo VPE-qGM (*qGM Visual Programming Environment*). Além de disponibilizar um ambiente de programação visual para o desenvolvimento de algoritmos quânticos, o VPE-qGM também permitirá a visualização da simulação das computações numa abordagem sequencial e paralela.

Substituindo a noção de circuito pela noção de processo computacional, a interpretação semântica de portas lógicas quânticas é obtida pela sincronização de processos elementares clássicos, considerados como transições de estados de uma memória global e compartilhada. O conjunto de processos que interpretam as portas lógicas quânticas não está definido a priori, e pode ser gerado a partir de processos elementares que satisfazem condições de sincronização e ortonormalidade [5]. A semântica para o paralelismo quântico é obtida pelo construtor Produto Paralelo e, na sua concepção dual, o construtor Soma Não-determinística provê interpretação para as operações de medidas, caracterizando o conflito de acesso às posições de memória.

No estágio atual, as bibliotecas e os métodos de suporte do VPE-qGM, integram recursos computacionais de três interfaces gráficas: (*i*) Editor de Processos Quânticos, viabilizando construção de operadores quânticos com ênfase nas operações de medidas e sincronização de portas quânticas unitárias; (*ii*) o Editor de Memória Quântica, possibilitando a configuração de estados com controle das condições de normalidade; e (*iii*) o Simulador, onde ocorrem as computações referentes à simulação da evolução temporal da aplicação.

O Editor de Processos Quânticos disponibiliza métodos que permitem a construção gráfica de portas unitárias e controladas, sendo possível sincronizá-las para obtenção de operadores multidimensionais. Incluem-se também métodos para geração dinâmica de processos, para o controle do aumento exponencial das posições de memória ocasionado pela expansão de operadores multidimensionais e para validação dos parâmetros dos processos quânticos.

A composição dos sistemas modelados não se restringe às portas quânticas existentes na biblioteca qGM-Analyser. Operadores customizados podem ser definidos e inseridos graficamente no sistema, desde que os mesmos atendam às condições de definição e de sincronização. Neste caso, os métodos de controle inseridos no construtor de operadores customizáveis garantem a validação de parâmetros dos componentes do sistema, incluindo mensagens de alerta que informam ao usuário os erros no processo construtivo. Após modelagem gráfica da aplicação, o ambiente gera um arquivo no formato XML, utilizado para posterior edição ou simulação.

No Editor de Memórias Quânticas, apresentam-se opções para criação e edição de memórias de acordo com a dimensão da aplicação corrente. Para configuração dos estados de memória

na inicialização de um processo quântico, considera-se o número de qubits, dispensando conhecimento das posições clássicas para construção dos estados da base computacional [3]. A instanciação de posições de memória podem ocorrer em conjunto ou individualmente, no respectivo campo na interface gráfica. Após configuração dos estados o VEP-qGM gera um arquivo no formato XML para posterior reconstrução ou inicialização de uma simulação.

Pela interface do Simulador, pode-se selecionar arquivos descritores de processos e de configuração de memória. Seguem-se a análise de parâmetros do arquivo descritor e o gerenciamento das operações e construtores que possibilitam a simulação sequencial. Assim, o método *produto-tensorial* da biblioteca qGM-Analyser recebe como parâmetros as matrizes associadas aos operadores quânticos básicos, retornando o operador resultante do produto tensorial. O armazenamento deste operador em uma lista temporária de execução ordenada informa sua composição e o estado global corrente da memória.

Cada sincronização de processos está associada a uma estrutura matricial na lista de execução, que dá semântica ao estado global do sistema. A utilização dessa metodologia permite que todas as matrizes associadas aos operadores e que definem a aplicação, sejam geradas dinamicamente pelo programa, eliminando a necessidade de armazená-las na memória. Estas matrizes são geradas de acordo com as solicitações do arquivo descritor, e serão apagadas no momento da finalização da simulação ou quando um novo arquivo descritor é gerado por novas especificações nas interfaces. Criação e/ou eliminação de matrizes correspondem a operações internas do VPE-qGM. Analogamente, no arquivo contendo uma configuração de memória para uma aplicação os valores das posições do estado corrente são normalizados, proporcionando valores corretos para aplicação dos operadores quânticos.

Quando uma sincronização de processos é detectada pelo Simulador, sua respectiva matriz é selecionada na lista de execução (total ou passo-a-passo) e os valores presentes nas posições de memória são transferidos para um vetor que representa o estado global da memória. A correta aplicação desta matriz sobre esse vetor é garantida pelo produto-tensorial, definido na biblioteca qGM-Analyser. O vetor resultante da aplicação do operador é o novo estado global de memória, que é atualizado na interface do Simulador.

Na busca de novos resultados, as atividades futuras consideram a integração do protótipo VPE-qGM ao ambiente de execução VirD-GM [4], para implementação paralela e/ou distribuída de algoritmos (básicos) da computação quântica. Salientam-se também a realização de testes envolvendo o VPE-qGM, com ênfase na validação do construtor soma não-determinística e aplicação de operações de medidas. Por fim, novas extensões do protótipo estão em desenvolvimento para suporte à interpretação da interferometria quântica [8].

Palavras-chave: *Computação Quântica, Programação Quântica Visual, Simulação Quântica*

Referências

- [1] R. Amaral and R. Reiser and A. Costa, Interpretações do Interferômetro de Mach-Zehnder no Modelo qMG, CNMAC 2008, 1–10.
- [2] M. Brown, Python: The Complete Reference, Osborne/McGraw-Hill, 2001.
- [3] I. Chuang and M. Nielsen, Quantum Computation and Quantum Information, Cambridge University, 2000.
- [4] V. Fonseca and R. Reiser and A. Yamin and M. Pilla, VirD-GM: Towards a Grid Computing Environment, Proceedings of CCGRID 2007, 2007, 1–6.
- [5] O. Pessoa, Conceitos de Física Quântica, Editora Livraria da Física, São Paulo.
- [6] R. Portugal and C. Lavor and N. Maculan, Uma introdução à Computação Quântica, Notas em Matemática Aplicada - SBMAC, SP, 2004.
- [7] R. Reiser and R. Amaral and A. Costa, Quantum Computing: Computation in Coherence Spaces, Proceedings of WECIQ 2007, 2007, UFCG, 1–10.
- [8] T. Ricci and F. Ostermann and S. Prado, O tratamento clássico do interferômetro de Mach-Zehnder: uma releitura mais moderna do experimento da fenda dupla na introdução da física quântica, *Revista Brasileira de Ensino de Física*, 29, 2007, 79–88.