

Modelagem e Implementação dos Construtores de Macro, Não-determinismo e Projeções no Ambiente VirD-GM*

Anderson Pinheiro,[‡] Felipe Munhoz, Renata Reiser, Adenauer Yamin

¹Centro Politécnico – Universidade Católica de Pelotas (UCPEL)
Rua Félix da Cunha, 412 – 96.010-000 – Pelotas – RS – Brazil

{andersonbp, fmunhoz, reiser, adenauer}@ucpel.tche.br

Abstract. *The aim of this work is the modeling and implementation of process constructors: macro, non-deterministic sum and projection, providing simulation of quantum applications developed in the visual programming environment VPE-qGM and performed by parallel and/or distributed approach in the execution environment VirD-GM.*

Resumo. *Este trabalho tem por objetivo a modelagem e implementação dos construtores: macro, soma não-determinística e projeção, viabilizando a simulação de aplicações quânticas modeladas no ambiente VPE-qGM e com suporte à execução paralela e/ou distribuída no VirD-GM.*

1. Introdução

Este trabalho colabora com a integração dos ambientes VPE-qGM (*Visual Programming Environment for the Quantum Geometric Machine Model*) [Maron et al. 2009] e VirD-GM (*Virtual Distributed Geometric Machine Model*) [Fonseca 2008] visando a simulação paralela de algoritmos quânticos. Neste contexto, as possibilidades de paralelismo são especificadas no ambiente de programação visual VPE-qGM, através da aplicação de construtores de processos e inicialização de estados (clássicos ou quânticos), ambos indexados por posições de memória global simulando a base computacional do sistema quântico. O ambiente de execução VirD-GM, por sua vez, recebe os arquivos descritores de processos e memória no formato XML. Após a interpretação destas descrições, a disponibilidade das unidades computacionais é analisada para distribuição dos processos com base na estruturação do sistema multiprocessado e no estado de ocupação dos nodos.

O objetivo deste trabalho é a expansão do VirD-GM pela modelagem e implementação de construtores específicos, complementando as funções que o ambiente de desenvolvimento VPE-qGM já disponibiliza para modelagem do paralelismo quântico e sua aplicação na simulação de operadores de medida e transformações unitárias [Nielsen and Chuang 2000]. O paralelismo quântico é obtido pela sincronização de n transformações unitárias gerando a sobreposição de 2^n estados clássicos, representando a base computacional. Mais especificamente, a proposta consiste na implementação dos construtores: macro (Mac), soma não-determinística (Snd) e projeção (Proj). Este artigo está dividido da seguinte forma: na Seção 2 os principais componentes do VPE-qGM são brevemente descritos, na Seção 3 são apresentadas as estruturas dos construtores propostos, seguida pelas considerações finais.

*Projeto parcialmente financiado: Processo 476933/2007-2 Edital MCT/CNPq 15/2007 Universal Faixa A e Processo 502999/2008-0 Apoio Técnico

[‡]Bolsista BIC/UCPEL

2. Ambiente de Desenvolvimento VPE-qGM

Na Figura 1 é possível observar a modelagem, no editor de processos do VPE-qGM, do algoritmo quântico de Grover[Nielsen and Chuang 2000] para busca de dados não ordenados. Os três primeiros processos são macros que representam as etapas executadas pelo algoritmo: (i) inicialização dos estados quânticos, (ii) superposição dos estados para identificação do elemento procurado, (iii) redistribuição da probabilidade com ênfase no elemento procurado. As duas últimas etapas referem-se operação de medida quântica, a qual é definida pela composição de dois construtores: (i) operador de projeção requerido pelas posições associadas ao estado de memória quântica de acordo com a dimensão da aplicação corrente; e (ii) operador de probabilidade. A expansão das macros, com a correspondente composição de processos elementares quânticos, pode ser obtida pela interface da ferramenta[Maron et al. 2009].

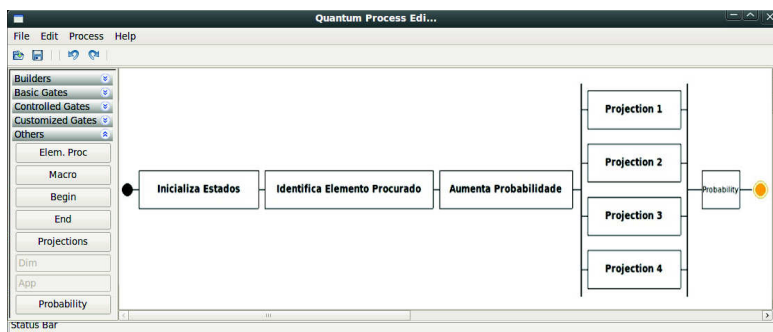


Figura 1. Modelagem do Algoritmo de Grover

3. Extensão do Ambiente VirD-GM para suporte aos novos construtores

O ambiente de execução VirD-GM, é composto pela integração dos módulos: VirD-Launcher, VirD-Loader e VirD-Exec [Munhoz et al. 2009], sobre os quais foram implementados os construtores descritos nesta seção.

3.1. Construtor de Macro

A componente macro no atual contexto do ambiente VPE-GM consiste em um único processo que referencia um arquivo descritor XML, o qual pode conter uma composição mais complexa de processos. Esta estrutura de dados agiliza a modelagem de novos programas através da reutilização de estruturas previamente desenvolvidas. A implementação do componente `Mac` é compatível com as abstrações do ambiente VPE-qGM, o qual considera uma estrutura interna diferenciada, definida recursivamente por composição de envelopes. A modelagem do construtor `Mac` segue as etapas apresentadas na Figura 2(a).

O trabalho considera a modelagem e implementação de dois tipos de estrutura: (i) `MacClassic` indicando a macro que contem somente um arquivo descritor modelando a abstração de um processo clássico; e (ii) `MacQuantum` para a estrutura de macro com a descrição recursiva dos processos concorrentes que modelam a abstração de processo quântico. A execução do construtor `Mac` implementado no módulo VirD-Loader

ocorre durante a leitura dos parâmetros dos processos descritos no XML. Salienta-se que a implementação suporta a composição de macros, as quais também podem ser sequencializadas ou paralelizadas.

As macros possibilitam a visualização e organização das etapas de simulação das aplicações, facilitando sua validação.

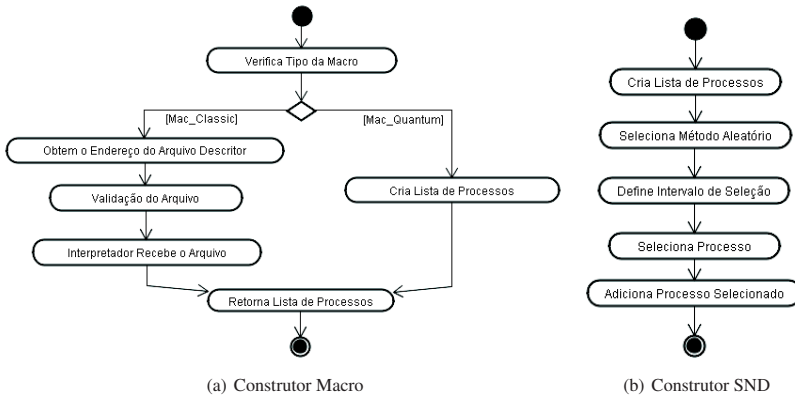


Figura 2. Diagrama do Construtor Macro e SND

3.2. Construtor Soma Não-Determinística

O construtor `Snd` é definido por dois ou mais processos elementares e/ou envelopes. Como principal característica, tem-se a aleatoriedade na escolha do processo a ser executado, dentre os processos que compõem o construtor. As funcionalidades do construtor estão organizadas nas etapas apresentadas no diagrama da Figura 2(b). De forma análoga, a implementação do construtor `Snd` também foi estruturada no componente `VirD-Loader`. Sua execução ocorre durante a interpretação do arquivo descritor de processos. Quando o construtor é identificado, aciona-se o método aleatório responsável pela seleção do processo ou envelope para execução.

3.3. Construtor de Projeção

Considerando o contexto no qual está inserida a operação de medida quântica para o ambiente VPE-qGM, a modelagem e implementação do construtor `Proj` determina o direcionamento do fluxo de dados em uma simulação, a partir de parâmetros como: (i) o qubit alvo da medida; (ii) o tipo de projeção; (iii) e cálculos probabilísticos dependentes dos parâmetros apontados pelo estado corrente da memória. Na Figura 3 é possível observar o diagrama de atividades para este construtor. O código que implementa o construtor de projeção está distribuído em módulos distintos: (i) no `VirD-Loader` (Figura 3(a)), viabilizando a seleção e preparação dos processos que definem os tipos de projeções; (ii) no `VirD-Launcher` (Figura 3(b)) considerando os dados da memória corrente, são realizados os cálculos referentes a probabilidade de execução de cada projeção.

Portanto, o processo `DetProj` determina, a probabilidade e seleciona a correspondente projeção na execução da operação de medida. Para realização dos cálculos,

utiliza-se uma lista de projeções representadas por intervalos, definidos através das probabilidades associadas as posições de memória da base computacional das respectivas projeções. Após a construção dessa lista, um valor rômico é gerado sendo comparado com os dados dos intervalos que compõem as projeções. O índice da lista, no qual o intervalo que contém este valor rômico está inserido, determina qual projeção será executada. Neste momento, as demais projeções são eliminadas da lista de execução.

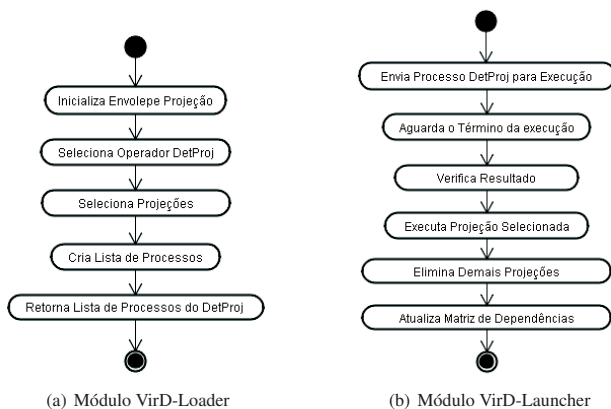


Figura 3. Diagrama do Construtor Projeção

4. Considerações Finais

A atual fase deste trabalho considera a execução de testes para validação dos construtores e suporte a execução paralela das aplicações já modeladas no VPE-qGM. Na sequência, tem-se a integração do ambiente VirD-GM com sistemas DSM (*Distributed Shared Memory*) contribuindo com a modelagem da base computacional que define o espaço de estados simulando a memória quântica, associado a aplicação quântica em desenvolvimento. Neste caso, tem-se como meta a análise da compatibilidade dos componentes existentes do ambiente VirD-GM com o sistema DSM Terracotta[Terracotta 2008].

Referências

- Fonseca, V. S. d. (2008). VirD-GM: Uma contribuição para o modelo de distribuição e paralelismo do projeto D-GM. Dissertação de mestrado em ciência da computação, UCPel, Pelotas,RS.
- Maron, A., Reiser, R., Yamin, A., and Pilla, M. (2009). Aplicações do protótipo VPE-qGM: Simulação via software do algoritmo de Grover. In *X WSCAD/WIC*, pages 1–4.
- Munhoz, F., Fonseca, V., Vivan, G., Reiser, R., and Yamin, A. (2009). Arquitetura de software da VirD-GM: Modelagem e funcionalidades. *VIII ERAD*, 1:209–212.
- Nielsen, M. A. and Chuang, I. L. (2000). *Quantum Computation and Quantum Information*. Cambridge University Press.
- Terracotta (2008). *The Definitive Guide to Terracotta: Cluster the JVM for Spring, Hibernate and POJO Scalability*. Apress.