

Fluid Computing: uma proposta para utilização de uma máquina química abstrata como plataforma paralela distribuída

Rui R. de Mello Jr.^{1 2}, Gabriel A. L. Paillard³, Felipe M. G. França¹

¹Universidade Federal do Rio de Janeiro (UFRJ)
Rio de Janeiro – RJ – Brasil

²Instituto de Pesquisas da Marinha (IPqM)
Rio de Janeiro – RJ – Brasil

³Universidade Federal do Ceará (UFC)
Ceará – CE – Brasil

ruirodrigues@cos.ufrj.br, gabriel@virtual.ufc.br, felipe@cos.ufrj.br

Abstract. *This paper describes the exploitation of parallelism power expressed by Gamma (General Abstract Model for Multiset mAnipulation) computational model in order to provide a parallel and distributed computational platform using information centric networks (ICN) and approximate computing.*

Resumo. *Este artigo descreve o trabalho de pesquisa que explora o poder de paralelismo expresso pelo modelo computacional Gamma (General Abstract Model for Multiset mAnipulation) para fornecer uma plataforma computacional paralela e distribuída, através da utilização de redes centradas em conteúdo (ICN) e computação aproximativa.*

1. Introdução

A computação paralela tem sido amplamente utilizada como ferramenta para obtenção de desempenho em cenários onde a arquitetura de Von Neumann está próxima dos limites de exploração de desempenho do hardware [Junior et al. 2016]. Apesar da validade da lei de Moore, possíveis melhorias tecnológicas não estão sendo convertidas em desempenho de maneira proporcional. Assim, o uso de processadores multi-core vem crescendo atualmente. Entretanto, programar de maneira paralela é uma tarefa bastante árdua, tendo em vista a realização de controle de carga, controle e distribuição de tarefas entre processadores, etc. Tais atividades tornam complexa a tarefa de programar de maneira paralela.

Nesse contexto, o uso de modelos onde a expressão do paralelismo ocorra de forma natural vem aumentando. Entre eles, destaca-se o modelo dataflow com um grande potencial para expor o paralelismo. Neste modelo, em contraste com o modelo de Von Neumann, onde a execução é guiada pelo fluxo de controle, a execução do programa é conduzida pelos dados. Desta forma, o modelo dataflow pode ser representado por um grafo direcionado, onde os vértices representam as operações a serem realizadas e as arestas representam as dependências de dados entre essas operações. Portanto, uma vez que seus operandos estejam disponíveis, a operação pode ser executada.

Por outro lado, Gamma (*General Abstract Model for Multiset manipulation*) foi proposto como um formalismo para a especificação de programas baseado na reescrita paralela de multiconjuntos [Banâtre and Le Métayer 1986]. O modelo de execução é não determinístico, uma vez que seus elementos podem reagir livremente de forma naturalmente paralela, tornando os detalhes da implementação do paralelismo transparentes ao programador. O modelo compreende uma base de dados única, o multiconjunto, onde os dados são representados por elementos deste multiconjunto. Gamma apresenta uma metáfora de reações químicas, onde o multiconjunto (solução química) é composto de vários elementos (moléculas). Ações (reações químicas) são especificadas para executar nos elementos do multiconjunto, de acordo com um conjunto de condições (condições de reação). Este paradigma pode ser aplicado a distintos domínios de aplicação, como processamento de Imagens e fusão de dados para acompanhamento de contatos [Junior et al. 2016]. Em [Paillard et al. 2001] é sugerida uma “*Proposta de um Escalonador para Gamma*” onde acredita-se que a implementação deste trabalho possa alavancar o desempenho das implementações de Gamma que utilizam um ambiente de hardware paralelo, especialmente [de Almeida et al. 2016].

Desta forma, esta pesquisa pretende demonstrar pela primeira vez a equivalência entre os modelos computacionais Gamma e Dataflow, tornando possível explorar benefícios de ambos, o que confere maior robustez para os modelos. Além disso, o estudo pretende transformar o paradigma Gamma em um modelo onde a computação permeie os recursos computacionais disponíveis de maneira robusta, distribuída e onde seja possível equalizar desempenho e precisão.

2. Motivação

Os modelos computacionais Gamma e Dataflow apresentam uma surpreendente similaridade, onde vértices e arestas de um grafo dataflow podem ser transformados em reações Gamma e elementos do multiconjunto, respectivamente. Tal equivalência permite que códigos inicialmente desenvolvidos sob o paradigma Gamma, possam usufruir de benefícios de pesquisas realizadas sob o modelo dataflow, como por exemplo, execução especulativa e fora de ordem [Marzulo et al. 2008] e reuso de traço de instruções [Rouberte et al. 2018]. Além disso, o modelo dataflow poderia extrair benefícios do modelo Gamma, ao passo que este último tem excelente potencial para ser utilizado em ambientes distribuídos, inclusive como plataforma para *Internet of Things* - IoT. Desta forma, explorar esta similaridade traz maior robustez às aplicações. Por exemplo, é bastante complexo explorar características de reuso em um código em linguagem imperativa, como C. Entretanto, este código poderia ser convertido para Dataflow, e consequentemente Gamma, onde o reuso é possível e bastante explorado.

Gamma possui um excelente potencial para utilização em ambientes distribuídos. Este fato decorre da possibilidade de se distribuir o multiconjunto pela rede, explorando o conceito de localidade, onde os dados devem ser processados próximo ao local onde residem. Desta forma, as reações Gamma poderiam ser distribuídas de forma a serem executadas próximas aos locais onde os dados que serão utilizados estejam. Para tanto, seria inserido o conceito de membranas ao multiconjunto, onde o mesmo seria dividido em diversos subconjuntos. Da mesma forma, seria interessante inserir o conceito do Radnet-S [Lima et al. 2018] na implementação, um mecanismo que utiliza um protocolo centrado em interesse para a transmissão de logs através de canais não seguros. Assim reações po-

deriam “demonstrar interesse” em determinados subconjuntos do multiconjunto Gamma. Adicionalmente, pretende-se pesquisar conceitos e técnicas de computação aproximativa para a aplicação ao modelo Gamma, onde a relação entre desempenho e precisão do sistema poderia ser equalizada. Desta forma, estaríamos trabalhando por um modelo computacional naturalmente paralelo, robusto, seguro e distribuído.

Tendo em vista o desafio de fornecer uma plataforma onde a computação permeia a rede pela utilização robusta e segura dos recursos computacionais disponíveis, surge a motivação deste trabalho, que investe em um modelo computacional que fornece a possibilidade de expressar problemas de uma maneira simples e naturalmente paralela onde detalhes de implementação do paralelismo são transparentes para o programador. Assim, tal modelo computacional seria obtido através do estudo da similaridade entre Gamma e Dataflow, estudo de conceitos de computação aproximativa, estudo da proposta de um escalonador eficiente para Gamma, estudo de questões de localidade e protocolos seguros centrados em interesse.

3. Descrição da Pesquisa - Estágio Atual

Atualmente, o trabalho intitulado “*Exploring the Equivalence between Dynamic Data-flow Model and Gamma - General Abstract Model for Multiset manipulation*” foi aceito para ser apresentado no *8th MPP - Workshop on Parallel Programming Models*, um evento que ocorrerá durante o *33rd IPDPS - International Parallel and Distributed Processing Symposium*. Trata-se do primeiro passo para o estudo da equivalência entre os modelos Gamma e Dataflow. Aqui o foco inicial foi demonstrar a equivalência entre Gamma e Dataflow e apresentar os algoritmos de transformação necessários.

O segundo passo será a submissão de um estudo contendo as provas formais desta equivalência. Tal estudo já encontra-se em fase final de revisão e consiste na apresentação de uma série de provas formais realizadas pelo princípio da indução finita, onde a equivalência entre vértices e arestas e suas correspondentes reações e elementos são apresentadas, mostrando formalmente tal equivalência.

Por fim, as transformações entre os modelos serão implementadas, onde um estudo de caso mostrará os benefícios que podem ser obtidos através destas transformações. Com isso, o estudo acerca da equivalência entre os modelos estaria finalizado.

4. Relevância da Pesquisa e conclusões

Gamma fornece um estilo de programação mais robusto que os paradigmas tradicionais, do ponto de vista do desenvolvimento de programas paralelos, uma vez que não existe a preocupação com restrições de sequencialidade. Por outro lado, desenvolver programas diretamente em dataflow consiste em apresentar um esquema bastante natural para o cliente que não possui formação em computação, tornando menos árduas etapas de validação e levantamento de requisitos. Dessa forma, explorar a equivalência entre Gamma e Dataflow é tornar mais robustos tais modelos, explorando os benefícios de ambos. Vale ressaltar que, pelo que se sabe, é a primeira vez que tal equivalência é proposta.

Assim, a pesquisa em pauta contribui para o próprio modelo Gamma, tendo em vista a pequena quantidade de implementações existentes, o fato de que a implementação do “Escalaonador para Gamma” tem grande potencial para melhorar o desempenho das

implementações propostas e também pela possibilidade de explorar diversos benefícios de dataflow. Da mesma forma, este estudo também contribui para o modelo dataflow, pois estudos desenvolvidos sob o paradigma Gamma poderão ser aplicados a dataflow indiretamente. Além disso, a introdução de conceitos de computação aproximativa tem a vantagem de tornar o modelo atraente para outros conjuntos de problemas. Adicionalmente, aplicar conceitos de protocolos baseados em interesse, como o RadNet-S, torna o Gamma mais robusto, seguro e ideal para ser utilizado em aplicações de fusão de dados.

Desta forma, a proposta principal desta pesquisa é unir o conceito de Gamma aos conceitos de computação aproximativa, protocolos baseados em interesse e dataflow, fornecendo uma plataforma computacional naturalmente paralela, robusta, segura e distribuída. Com isso, aplicações poderão ser escritas de maneira simplista e natural pelos programadores. Tal plataforma seria de grande relevância para a comunidade de desenvolvedores de soluções de problemas que exigem alto desempenho computacional, por ser genérica o suficiente para suportar uma vasta gama de diferentes classes de aplicações, aliado à possibilidade de se optar por não prosseguir de forma exaustiva com o processamento, e sim, aceitar respostas aproximadas em casos onde isso é aceitável, levando a uma economia de recursos computacionais significativa.

Referências

- Banâtre, J.-P. and Le Métayer, D. (1986). A new computational model and its discipline of programming. In: *Rapport de Recherche 566*, INRIA, France.
- de Almeida, R. H. P., Mello, Jr., R. R., Paillard, G. A. L., and França, F. M. G. (2016). A gpu-based implementation for the gamma multiset rewriting paradigm. In *Proceedings of the 31st Annual ACM Symposium on Applied Computing*, SAC '16, pages 1225–1230, New York, NY, USA. ACM.
- Junior, R. R., Almeida, R. H., França, F. M., and Paillard, G. A. (2016). A parallel implementation of data fusion algorithm using gamma. In *2015 International Symposium on Computer Architecture and High Performance Computing Workshop (SBAC-PADW)*, volume 00, pages 109–114.
- Lima, L., Filho, P. C., Dutra, D. L. C., Amorim, C. L., Macedo, E. L. C., Silva, R. S., Coutinho, M. A., and de Moraes, L. F. M. (2018). Radnet-s: Um mecanismo para transmissão segura e secreta de registros syslog. In *Anais do XXIII Workshop de Gerência e Operação de Redes e Serviços (WGRS - SBRC 2018)*, volume 23, Porto Alegre, RS, Brasil. SBC.
- Marzulo, L. A. J., Franca, F. M. G., and Costa, V. S. (2008). Transactional wavecache: Towards speculative and out-of-order dataflow execution of memory operations. In *2008 20th International Symposium on Computer Architecture and High Performance Computing*, pages 183–190.
- Paillard, G., Franca, F., and Filho, J. (2001). Uma proposta de um escalonador para gamma. In *II Workshop em Sistemas Computacionais de Alto Desempenho*, pages 47–54, Pirenópolis, GO.
- Rouberte, L., Sena, A. C., Nery, A. S., Marzulo, L. A. J., Alves, T. A. O., and França, F. M. G. (2018). Df-dtm: Dynamic task memoization and reuse in dataflow. *Concurrency and Computation: Practice and Experience*, 0(0):e4937. e4937 cpe.4937.