

# Em Direção à um *Benchmark* de *Workload* Sintético para Paralelismo de *Stream* em Arquiteturas *Multicore*

Carlos A. F. Maron<sup>1</sup>, Dalvan Griebler<sup>1</sup>, Luiz Gustavo Fernandes<sup>1</sup>

<sup>1</sup> Programa de Pós-Graduação em Ciência da Computação  
Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)  
Grupo de Modelagem de Aplicações Paralelas (GMAP)  
Porto Alegre – RS – Brasil

[carlos.maron,dalvan.griebler]@acad.pucrs.br, luiz.fernandes@pucrs.br

**Resumo.** *O processamento de fluxos contínuos de dados (stream) está provocando novos desafios na exploração de paralelismo. Suítes clássicas de benchmarks não exploram totalmente os aspectos de stream, focando-se em problemas de natureza científica e execução finita. Para endereçar este problema em ambientes de memória compartilhada, o trabalho propõe um benchmark de workload sintético voltado para paralelismo stream.*

## 1. Introdução

Frequentemente, sistemas paralelos necessitam uma nova proposta de software ou algoritmo. Estas propostas geralmente são métodos diferentes das convencionais implementações de códigos sequenciais. Atualmente, processamento de *stream* vem se intensificando devido a natureza infinita de dados, áudio e vídeo, e por isso, uma nova proposta de paralelismo tem se fortalecido neste tipo de domínio.

Segundo as estatísticas do YouTube<sup>1</sup>, desde 2014 o número de horas de visualizações de vídeos dos usuários do YouTube vêm crescendo mais de 40% a cada ano. De certa forma, isso também tem refletido no aumento do tráfego de internet e a quantidade de dados gerados pelo uso dela. Desse modo, as empresas estão buscando o processamento desses dados considerando o fator de tempo "agora", e isso tem preocupado pois as arquiteturas precisam estar preparadas para esse tipo de processamento.

Portanto, esse tipo de processamento tem motivado pesquisas com objetivo de desenvolver *frameworks* [Griebler et al. 2015, Aldinucci et al. 2011] e técnicas de paralelismo de *stream* em arquiteturas de memória compartilhada. Porém, suítes de *benchmarks* e *micro-benchmarks* [Bailey et al. 1991, Bienia et al. 2008] que geralmente são usados na avaliação de desempenho não abrangem totalmente os aspectos de um processamento e até o paralelismo de *stream* (Múltiplas entradas e saídas, tamanho de *buffer*, quantidade e tipos de estágios, entre outros). Sendo assim, esta proposta busca desenvolver um *benchmark* de *workload* sintético que contemple de maneira mais ampla estas características em arquiteturas de memória compartilhada.

## 2. Trabalhos Relacionados

Clássicas suítes utilizadas para avaliação de desempenho em HPC (*High Performance Computing*) [Bailey et al. 1991, Bienia et al. 2008] utilizam algoritmos do domínio de *stream*, porém, suas execuções não refletem totalmente este tipo de processamento, sendo finito e com uma única entrada de dados. [Berrendorf et al. 2015], propõe um conjunto de *micro-benchmarks* para determinar parâmetros básicos de desempenho em arquiteturas *single-node* para HPC com placas gráficas e coprocessadores. É avaliado as-

<sup>1</sup><https://www.youtube.com/yt/press/statistics.html>

pectos referentes à acesso a memória, sincronização, comunicação e computação, utilizando chamadas das bibliotecas de programação OMP (*Open Multi-Processing*) e *micro-benchmarks* (STREAM<sup>2</sup>). Nos testes, as variáveis são: latência e *bandwidth* de memória RAM, atualização atômica, barreira, *reduction*, comunicação, DGEMM e SPMV.

Sabe-se que o experimento é uma das principais etapas de um método científico e na computação são realizados por ferramentas que avaliam tal aspecto desejado. Portanto, isso ocorre através de programas como os *benchmarks*. Porém, grande parte das suítes não abrangem totalmente as características do domínio de *stream* e não se aproximam de um ambiente real.

### 3. Proposta

Considerando importante a avaliação e as métricas de desempenho em HPC, propõe-se um *benchmark* de *workload* sintético que explore mais as propriedades de *stream* e sua paralelização em memória compartilhada. O objetivo nesse sentido é oferecer à possibilidade das características de *stream* serem ajustáveis durante os testes com o *benchmark*, com o propósito de aproximá-las à um ambiente real.

A Figura 1 representa um diagrama do *benchmark* proposto. **1:** realizará a leitura dos parâmetros que definem as características de uma aplicação de *stream* (quantidade de entradas e os tipos de *stream*, número e tipos de *workers*, entre outros). **2:** preparação do ambiente para o início dos testes (Ex: alocação em memória). **3:** realização da computação e processamento do ambiente *stream* com a coleta das métricas. **4:** etapa final de armazenamento dos resultados. Por fim, pretende-se validar este *benchmark* comparando os seus resultados com aplicações reais do domínio de *stream*.



Figura 1. Fluxograma do *benchmark*.

### Referências

- [Aldinucci et al. 2011] Aldinucci, M., Danelutto, M., Kilpatrick, P., and Torquati, M. (2011). Fastflow: high-level and efficient streaming on multi-core.(a fastflow short tutorial). *Programming Multi-core and Many-core Computing Systems, Parallel and Distributed Computing*.
- [Bailey et al. 1991] Bailey, D. H. et al. (1991). The NAS parallel benchmarks. *International Journal of High Performance Computing Applications*, 5(3):63–73.
- [Berrendorf et al. 2015] Berrendorf, R., Ecker, J. P., Scholl, S. E., and Mannus, F. (2015). Using Application Oriented Micro-Benchmarks to Characterize the Performance of Single-node Hardware Architectures. In *ADVCOMP 2015 : The Ninth International Conference on Advanced Engineering Computing and Applications in Sciences*, volume 6<sup>a</sup>, pages 28–35, Nice, France. IARIA.
- [Bienia et al. 2008] Bienia, C., Kumar, S., Singh, J. P., and Li, K. (2008). The parsec benchmark suite: Characterization and architectural implications. In *Proceedings of the 17th international conference on Parallel architectures and compilation techniques*, pages 72–81. ACM.
- [Griebler et al. 2015] Griebler, D., Danelutto, M., Torquati, M., and Fernandes, L. G. (2015). An Embedded C++ DomainSpecific Language for Stream Parallelism. In *International Conference on Parallel Computing, ParCo'15*, page 10, Edinburgh, Scotland, UK. IOS Press.

<sup>2</sup>Micro-benchmark utilizado para avaliar o desempenho de memória RAM.