

# Comparando o Desempenho entre Computação em Nuvem e Servidor Local na Execução do Método Fletcher \*

Brenda S. Schussler<sup>1</sup>, Pedro H. C. Rigon<sup>1</sup>, Matheus S. Serpa<sup>1</sup>, Cristiano A. Künas<sup>1</sup>, Alexandre Carissimi<sup>1</sup>, Jairo Panetta<sup>2</sup>, Philippe O. A. Navaux<sup>1</sup>

<sup>1</sup>Instituto de Informática – UFRGS - Porto Alegre – RS – Brazil

{bsschussler, phcrigon, msserpa, cakunas, asc, navaux}@inf.ufrgs.br

<sup>2</sup>Divisão de Ciência da Computação – ITA - São José dos Campos – SP – Brasil

jairo.panetta@gmail.com

**Resumo.** *A Modelagem Fletcher é utilizada na simulação de reconhecimento de bacias petrolíferas e requer utilização de computação de alto desempenho (HPC). Uma alternativa para reduzir os custos de aquisição e manutenção de plataformas de HPC é o uso de recursos computacionais em nuvem. Neste artigo comparamos o desempenho do Método Fletcher, executando-o no servidor local e em nuvem, fazendo uso de arquiteturas de hardware semelhantes.*

## 1. Introdução

Na atualidade, a exploração geofísica é fundamental na busca de recursos energéticos como Petróleo e Gás. No entanto, seu uso é limitado devido aos altos custos de perfuração, com precisão inferior a 50%. [Lukawski et al. 2014]. Sendo assim, diversos algoritmos são adotados pela indústria visando a redução dos custos de exploração. Neste contexto, o Método Fletcher é utilizado como base para muitos desses softwares, através da simulação da propagação de ondas. Para que as simulações sejam executadas com eficiência e precisão, é necessário que haja integração entre os ambientes de software e hardware, considerando a gestão dos recursos da memória, a paralelização no processamento dos dados e a arquitetura em questão.

A nuvem é vista como uma solução viável para a indústria de exploração geofísica, pois oferece diferentes infraestruturas de hardware e plataformas de desenvolvimento, além da escalabilidade e flexibilidade necessárias para lidar com a grande quantidade de dados gerados pelas simulações sísmicas [Lu et al. 2017]. Desta forma, tendo em vista a popularidade e os crescentes avanços na utilização de computação em nuvem, torna-se relevante avaliar o desempenho dessas aplicações sísmicas em um ambiente de nuvem para determinar sua viabilidade em comparação com outras soluções.

Neste artigo, realizamos uma comparação entre a execução do Método Fletcher na nuvem utilizando a plataforma Google Cloud e localmente no Parque Computacional de Alto Desempenho da UFRGS (PCAD), fazendo o uso de arquiteturas semelhantes. Nosso objetivo é comparar a eficiência entre os dois ambientes, considerando fatores como desempenho, uso de recursos e escalabilidade.

---

\*Este estudo foi parcialmente apoiado pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001, pela Petrobras sob número 2020/00182-5 e pelo edital CNPq/MCTI/FNDCT - Universal 18/2021 sob número 406182/2021-3. Alguns experimentos deste trabalho utilizaram os recursos da infraestrutura PCAD, <http://gppd-hpc.inf.ufrgs.br>, no INF/UFRGS.

## 2. Apresentação do Método Fletcher e Metodologia Experimental

A aplicação Modelagem Fletcher simula a propagação de ondas através do tempo. Para isso, são realizados cálculos de derivadas cruzadas, que, conforme constatado, demandam 76% do tempo de execução da versão Original. Deste modo, buscando reduzir a quantidade de operações para calcular as derivadas cruzadas, foi elaborada a versão otimizada Der1Der1 que calcula a derivada cruzada em  $xy$  como primeira derivada em  $y$  da primeira derivada em  $x$  [Serpa et al. 2019].

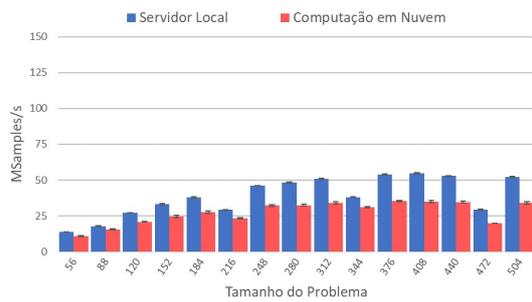
Os experimentos foram executados utilizando as versões do algoritmo do Método Fletcher, **Original** e **Der1Der1**. Para cada uma das versões, executamos as codificações **CUDA** e **OpenMP** em 15 grades de tamanho crescente. As execuções foram realizadas em 2 ambientes diferentes: servidor local e nuvem. Ambos os ambientes de execução utilizam a arquitetura Broadwell contendo dois processadores Intel Xeon (22 núcleos local, 22 vCPUs em nuvem) permitindo até 88 threads de execução; e a arquitetura Pascal, com uma GPU NVIDIA Tesla P100-SXM2-16GB com 3584 núcleos CUDA. Devido a dificuldades com a disponibilidade dos serviços de nuvem, tanto no servidor local quanto na nuvem, as execuções foram realizadas utilizando apenas 16 threads para a codificação OpenMP, e utilizando uma thread com uma GPU NVIDIA P100 para a codificação CUDA.

## 3. Apresentação dos Resultados e Conclusão

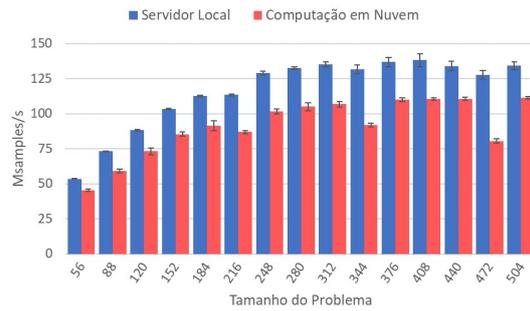
Nesta Seção apresentamos os resultados de desempenho em MSamples/s em função do tamanho do problema para as versões **Original** e **Der1Der1** com as codificações **OpenMP** executando em CPU em servidor local e em nuvem, e **CUDA** executando na GPU em servidor local e em nuvem. A métrica Msamples por segundo é definida como a divisão do número de pontos da grade calculados pelo tempo de execução da aplicação. Os gráficos apresentam os resultados obtidos através da média de 10 execuções e os intervalos de confiança de 95% segundo a distribuição T-Student.

Primeiramente, analisando os resultados obtidos com a codificação OpenMP (Figura 1), os gráficos demonstram que há diferença significativa no desempenho, seja comparando servidor e computação em nuvem, seja comparando as versões Original (Figura 1(a)) e Der1Der1 (Figura 1(b)). Os dados coletados com a codificação OpenMP mostram que o desempenho médio da versão Original em servidor local foi 27,76% superior ao desempenho na nuvem. A versão Der1Der1, por sua vez, obteve um desempenho médio 21,24% maior no servidor local em comparação com a execução na nuvem.

Estas diferenças de desempenho podem ser explicadas pela natureza das aplicações e pelas especificidades dos dois ambientes. A gestão da memória é de extrema importância em aplicações que exigem muito desse recurso, como é o do Método Fletcher. Nesse contexto, o ambiente compartilhado da nuvem pode resultar em uma menor disponibilidade de recursos de hardware, incluindo memória, para as aplicações que as demandam [Buyya et al. 2008]. Sendo assim, por tratar-se de um comparativo entre servidor local e ambiente de nuvem, é crucial considerar que, ao utilizar computação em nuvem, o acesso à memória é limitado pelo compartilhamento de recursos de hardware com outras aplicações e usuários, resultando em latências de acesso aos recursos mais elevadas, o que, por outro lado, não ocorre na infraestrutura local. Posto isso, os resultados obtidos demonstram-se coerentes.



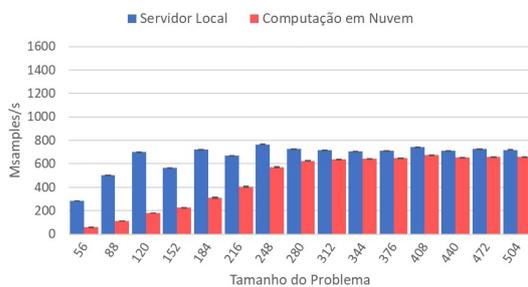
(a) Versão Original em codificação OpenMP.



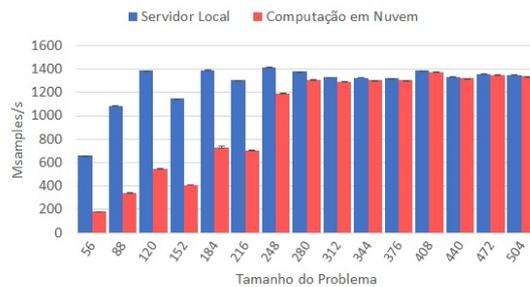
(b) Versão Der1Der1 em codificação OpenMP.

**Figura 1. Desempenho das versões Original e Der1Der1 em OpenMP medido em MSamples/s e calculado para 15 tamanhos de grade [56-504].**

Fazendo análise dos resultados obtidos utilizando CUDA (Figura 2), os gráficos expressam que a perda de desempenho comparando a execução em nuvem com a execução em servidor local se mantém, assim como, a diferença entre as versões Original (Figura 2(a)) e Der1Der1 (Figura 2(b)). Ao compararmos as execuções em CUDA da versão Original, observa-se que o servidor local obteve um desempenho médio 32,75% superior em relação às execuções na nuvem. Já ao compararmos as execuções da versão Der1Der1, em codificação CUDA, o servidor local apresentou um desempenho médio de 25,97% maior que na nuvem.



(a) Versão Original em codificação CUDA.



(b) Versão De1Der1 em codificação CUDA.

**Figura 2. Desempenho das versões Original e Der1Der1 em CUDA medido em MSamples/s e calculado para 15 tamanhos de grade [56-504].**

Outro aspecto a ser observado nos gráficos das codificações em CUDA (Figura 2), é o comportamento da variação do desempenho entre o servidor local e a nuvem inversamente proporcional ao tamanho do problema. Esse padrão ocorre pois, quanto menor a operação (menor o tamanho do problema), mais impactante é a latência associada à movimentação dos dados e à gestão dos recursos de memória. Desta forma, ao aumentar o tamanho do problema que faz o uso da GPU, a perda de desempenho relacionada a transferência de dados torna-se menos relevante, fazendo com que o desempenho da execução na nuvem aproxime-se cada vez mais do desempenho em servidor local.

Além disso, outro fator relevante a ser considerado como causador do overhead maior de transferência de dados na nuvem em comparação com a transferência em uma máquina local, é o gargalo gerado pelas interconexões de rede presentes na nuvem

[Roloff et al. 2017]. A transferência de informação entre redes interconectadas na nuvem é substancialmente mais lenta em comparação com a comunicação direta entre componentes em um servidor local. Isso ocorre pois, ao fazer uso de um ambiente de nuvem, os componentes podem estar fisicamente afastados, bem como, pode ser necessário que os dados atravessem várias camadas de redes e servidores, gerando a latência de rede.

Finalmente, foi possível concluir que os resultados obtidos, de um modo geral, indicam que as execuções em nuvem (Google Cloud) obtiveram um desempenho médio inferior em relação as execuções em servidor local (PCAD), devido a presença de recursos de hardware dedicados localmente, em oposição aos recursos possivelmente compartilhados da nuvem, além do importante fator relacionado com a latência de rede presente na nuvem.

Posto isso, a característica de compartilhamento intrínseca à computação em nuvem, resultou na menor disponibilidade de recursos, em especial a memória, e consequentemente, isso impactou negativamente o desempenho das execuções neste ambiente. No entanto, é necessário salientar que a codificação CUDA obteve ótimos resultados na execução em nuvem para tamanhos de problema superiores a 300, sendo que, nesses casos, a diferença de desempenho em comparação ao servidor local é de apenas 1,27% na versão **Der1Der1**. Deste modo, considerando que, em geral, as simulações sísmicas reais utilizam tamanhos de grade maiores, a utilização da nuvem torna-se uma alternativa interessante e viável, se levarmos em conta também fatores como economia de investimentos em infraestrutura própria e a sua atualização ao longo do tempo sem maiores custos.

## Referências

- Buyya, R., Yeo, C. S., and Venugopal, S. (2008). Cloud computing: Review and open research issues. *Journal of Network and Computer Applications*, 34(1):1–25.
- Lu, B., Dashti, M., Moreland, P. J., and Yeo, M. P. S. (2017). Performance evaluation of seismic simulations on the cloud. *Computers & Geosciences*, 110:15–25.
- Lukawski, M. Z., Anderson, B. J., Augustine, C., Capuano Jr, L. E., Beckers, K. F., Livesay, B., and Tester, J. W. (2014). Cost analysis of oil, gas, and geothermal well drilling. *Journal of Petroleum Science and Engineering*, 118:1–14.
- Roloff, E., Diener, M., Gaspar, L. P., and Navaux, P. O. A. (2017). Hpc application performance and cost efficiency in the cloud. In *2017 25th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP)*, pages 473–477.
- Serpa, M., Pavan, P. J., Panetta, J., Azambuja, A., Carissimi, A., and Navaux, P. O. (2019). Portabilidade e eficiência do método fletcher de aplicações sísmicas em arquiteturas multicore e gpu. In *Anais do XX Simpósio em Sistemas Computacionais de Alto Desempenho*, pages 169–180. SBC.