

Explorando Oportunidades para Aceleração de Solvers OpenFOAM Utilizando GPU

Cirano Alves Belardony¹, João Vicente Ferreira Lima¹

¹Universidade Federal de Santa Maria – Centro de Tecnologia Av. Roraima, 1000 – CEP 97105-900 – Santa Maria – RS

cirano.belardony@acad.ufsm.br, jvlima@inf.ufsm.br

***Resumo.** Este trabalho explora a aplicação de GPUs para otimizar simulações de dinâmica dos fluidos computacionais (CFD) no software OpenFOAM, utilizando CUDA. Após identificar métodos de maior impacto no desempenho, destacou-se os solvers mais custosos em termos de tempo de execução. A pesquisa também avaliou a versão adequada para integração com GPUs*

1. Introdução

A dinâmica dos fluidos computacionais (CFD) é a análise de sistemas que envolvem o fluxo de fluidos, a transferência de calor e fenômenos associados, como reações químicas, por meio de simulação computacional [Versteeg et. al. 2007]. Existem diversos softwares disponíveis que possibilitam utilizar-se desta técnica, para esta pesquisa utilizou-se OPENFOAM, um conjunto eficiente e flexível de módulos escritos em C++ que podem ser usados para resolver problemas complexos de engenharia que envolvam operações e resoluções de campos tensoriais, utilitários para pré e pós-processamento de dados, modelagem de elementos físicos.

A utilização de GPU tem sido frequente devido sua capacidade de paralelizar e acelerar massivamente os cálculos matriciais, presentes em grande parte em simulação CFD [Posey 2019] [Fatahalian et. al. 2004]. O intuito deste trabalho é identificar, enunciar e explorar possíveis ferramentas que ajudarão a atingir o objetivo de acelerar os cálculos realizados pelo software OpenFOAM.

2. Metodologia

O OpenFOAM possui duas principais versões: a mantida pela OpenFOAM Foundation, que foca em acessibilidade, colaboração comunitária e transparência, e a da OpenCFD/ESI [OPENCFD], voltada para suporte comercial e integração com soluções proprietárias. A análise revelou que a versão OpenCFD é compatível com o driver de compilação Nvcc, permitindo a implementação de código. O solver PDRFoam apresentou-se com alto custo computacional em termos de tempo de execução em nossos testes preliminares, especialmente no teste “pipLattice”, já presente no código-fonte da versão OpenFOAM em questão. Com a ferramenta perf, é possível mensurar o desempenho de cada método no OpenFOAM, fornecendo uma ordem de prioridade para os solvers mais críticos e que podem se beneficiar significativamente de aceleração.

Na Tabela 1, é possível analisar a parcela do tempo de execução obtida ao realizar o teste em um processador de 16 threads, operando a 2.9–4.2 GHz, em uma plataforma equipada com 16 GB de RAM DDR4 3200 MHz e um SSD NVMe Gen3 x4 de 512 GB.

Tabela 1. Tempo de execução dos métodos que utilizam cálculo matricial

| Método | T. Exec. (%) | Método | T. Exec. (%) |
|------------------------------|---------------------|-----------------------------------|---------------------|
| fvMatrix<double>::solve | 22.50% | fvMatrix<double>::solveSegregated | 7.05% |
| fvMatrix<double>::PCG::solve | 15.50% | fvMatrix<double>::H | 2.16% |
| GAMGSolver::precondition | 14.41% | fvMatrix<double>::A | 1.75% |
| GAMGSolver::Vcycle | 6.38% | GAMGSolver::Amul | 1.16% |

3. Conclusão

A aceleração de desempenho no OpenFOAM, especialmente em simulações CFD intensivas, é viabilizada pelo uso de GPUs e plataformas como CUDA, Thrust e cuBLAS, reduzindo tempos de execução e aumentando a eficiência no processamento de grandes volumes de dados. Ferramentas como perf ajudam a identificar métodos críticos para otimização, como os solvers listados na Tabela 1, que representam mais de dois terços do tempo de execução. A versão OpenCFD/ESI é mais adequada para integrar códigos CUDA devido sua nativa compatibilidade com drivers de compilação que trabalham com GPU.

Para os próximos passos, será utilizada a biblioteca Kokkos, que oferece uma abordagem portátil para programação paralela em diversas arquiteturas, sua adoção permitirá maior flexibilidade na otimização de desempenho, facilitando a portabilidade do código entre diferentes plataformas de hardware.

Referências

- VERSTEEG, H; MALALASEKERA, W. Introduction to Computational Fluid Dynamics, An: The Finite Volume Method, 2nd edition. [S. l.]: Pearson, 2007. ISBN 9780131274983.
- FATAHALIAN, K; SUGERMAN, J; HANRAHAN, P. Understanding the efficiency of GPU algorithms for matrix-matrix multiplication. Graphics Hardware (2004) T. Akenine-Möller, M. McCool (Editors), Stanford University, p. 133-137, 1 ago. 2004. DOI <https://doi.org/10.1145/1058129.1058148>.
- POSEY, Stan; PARIENTE, Frederic. ESI OpenFOAM Conference. Opportunities for GPU Acceleration of OpenFOAM, [s. l.], 29 out. 2019.
- OpenFOAM | Free CFD Software | The OpenFOAM Foundation.
- OPENCFD. OpenFOAM® - Official home of The Open Source Computational Fluid Dynamics (CFD) Toolbox.