

# Projeto de Pesquisa para Iniciação Científica

## Comparando a utilização de soluções livres e proprietárias para programação geral de GPUs

Isabella Basso do Amaral

Orientador: Alfredo Goldman  
Instituto de Matemática e Estatística  
Universidade de São Paulo  
São Paulo, Brasil  
[gold@ime.usp](mailto:gold@ime.usp)

*Resumo*—meow meow meow

*Palavras-chave*—GPU, CUDA, Vulkan, Compute, OpenCL, software, livre, aberto, proprietário

### I. INTRODUÇÃO

O computador tornou-se parte indispensável da pesquisa acadêmica, tendo funcionalidades que abrangem desde facilitar a leituras de artigos e livros até a resolução (aproximada) de problemas numéricos, que seriam insolúveis analiticamente.

Quanto ao último exemplo, gostaríamos de ressaltar, em especial, a capacidade de paralelizar tais problemas numéricos, de tal forma que conseguimos reduzir o tempo de execução. Otimizações no tratamento desses problemas são uma necessidade para o desenvolvimento da pesquisa científica como se dá atualmente.

#### A. A GPU e sua importância na pesquisa contemporânea

Um componente especializado em paralelização é a GPU (*Graphics Processing Unit*), ela vem sendo extensivamente utilizada para auxiliar na resolução de problemas físicos, biológicos, químicos e até mesmo da matemática moderna, tornando-se, por vezes, mais importante do que a CPU (*Central Processing Unit*) do dito sistema.

Haja vista que a GPU é um componente de hardware, o que significa que é necessário comandá-la, idealmente abstraindo detalhes do hardware específico, de tal forma que tenhamos fino controle sobre sua utilização porém sem comprometer seu desempenho (i.e. a abstração deve otimizar o que quer que seja tendo em mente o hardware específico).

#### B. Compiladores

Classicamente, utilizamos *shaders* para a programação de GPUs, sendo mais ou menos análogos à programação de CPU. O código do *shader* é, então, compilado em etapas, sendo primeiramente convertido para uma linguagem intermediária, onde é otimizado com base em conceitos primitivos próprios para abordagens de paralelização,

como o *thread* ou o *block*. Depois, sendo convertido para um binário otimizado para a GPU específica onde será executado, e então enviado para o *hardware*.

#### C. A relevância do software livre no contexto científico

Chamamos de **software livre** aquele que não depende de corporações (embora possa ser auxiliado por estas) para que se mantenha. Possuindo comunidades autônomas de usuários e desenvolvedores que o mantêm de acordo com interesses pessoais e plurais, porém não necessariamente sem o envolvimento de capital.

O *software* livre permeia todo o contexto computacional moderno, sendo utilizado extensivamente na internet (e.g.  $\approx X$  dos servidores utilizam *Linux*), no contexto do desenvolvimento de *software* e, especialmente, na pesquisa científica.

Contrastamos tal com o **software proprietário**, onde a contribuição de um indivíduo sempre será atrelada com interesses corporativos pois pertence à uma empresa e é secreto sendo, portanto, impossível auditá-lo.

Existe ainda o *superset* de *software* livre, que é o **software aberto**, onde pode existir uma empresa que o mantém de acordo com seus interesses, porém este não é secreto, podendo ser auditado e aceitando contribuições individuais<sup>1</sup>.

Note, então, que a utilização de *software* proprietário não cabe no contexto científico, pois demanda desmedida confiança à interesses corporativos, indo totalmente de encontro a princípios fundacionais da ciência como a temos hoje, onde reprodutibilidade e transparência são essenciais.

Com isso em mente, advogamos pelo uso exclusivo do *software* livre nesse contexto o que, infelizmente, ainda não é totalmente plausível.

<sup>1</sup>No presente projeto, no entanto, dedicaremos-nos exclusivamente à problemática do *software* livre X *software* proprietário.

*D. Comparando o software livre com o proprietário para aplicações científicas*

Como dito anteriormente, utilizamos *shaders* para a programação de GPUs, e no contexto específico de aplicações numéricas, nosso interesse é voltado à **programação geral** com GPUs (GPGPU), onde temos utilização extensiva da linguagem **CUDA**, desenvolvida pela empresa NVIDIA.

Dada sua simplicidade e abrangência, o CUDA segue invicto em aplicações científicas, superando alternativas livres por uma grande margem.

Esse fato nos traz diversos questionamentos que devem ser abordados no presente projeto, como:

- 1) Onde o CUDA se sobressai em comparação às alternativas livres?
- 2) Como podemos diminuir essa diferença e facilitar a adoção de alternativas livres?
- 3) Quais são as dificuldades para que a migração ocorra e como podemos facilitá-la?

Como alternativas ao CUDA, destacamos o **OpenCL** e o **Vulkan**, que serão utilizados nas comparações.

#### REFERÊNCIAS

- [1] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” *Phil. Trans. Roy. Soc. London*, vol. A247, pp. 529–551, April 1955.
- [2] J. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
- [3] I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in *Magnetism*, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
- [4] K. Elissa, “Title of paper if known,” unpublished.
- [5] R. Nicole, “Title of paper with only first word capitalized,” *J. Name Stand. Abbrev.*, in press.
- [6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” *IEEE Transl. J. Magn. Japan*, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetism Japan, p. 301, 1982].
- [7] M. Young, *The Technical Writer’s Handbook*. Mill Valley, CA: University Science, 1989.

IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove the template text from your paper may result in your paper not being published.