CT&I e SOCIEDADE
XVIII SEMINÁRIO DE INICIAÇÃO CIENTÍFICA
XV JORNADA DE PESQUISA
XI JORNADA DE EXTENSÃO
4 a 8 de OUTUBRO de 2010
UNIJUÍ UNIVERSIDADE REGIONAL

EXECUÇÃO PARALELA DE APLICAÇÕES ATRAVÉS DA ARQUITETURA GPU[1]

Mathäus Gregório Mendel[2], Paulo Sérgio Sausen[3], Edson Luiz Padoin[4]. UNIJUI

INTRODUÇÃO: O projeto dos fabricantes de processadores de vídeo, de implementar a abstração em nível de software para uso de seus recursos não somente para processamento gráfico, mas também para execução de algoritmos computacionais onde não há necessidade de renderização por parte da Graphic Processing Unit (GPU), tem se mostrado extremamente eficiente em seu propósito. O foco inicial do projeto é avaliar a escalabilidade, speed-up, performance, desempenho e eficiência das aplicações ao serem executadas pela GPU e implementar os algoritmos necessários para a paralelização dos mesmos. Este estudo não apenas envolve a análise dos recursos que a GPU oferece, mas também os já atualmente utilizados como threads, OpenMP. Atualmente existem diversas implementações de GPGPU para as mais diversas linguagens de programação. Dentre estas pode-se citar o OpenCL, AMD Stream, Direct Compute e CUDA, sendo este último o mais estável. Uma vez que o OpenCL tende a ser o padrão para GPGPU, sendo que várias GPUs já são habilitadas para rodá-lo, isso permite que o desenvolvimento em CUDA seja altamente viável, pois toda sua API é desenhada em cima desses padrões, tornando a migração de CUDA para OpenCL, e vice-versa, muito mais simples do que as tecnologias concorrentes. MATERIAL E MÉTODO: Foram estudadas diversas bibliotecas nesta fase inicial do projeto, dentre elas pode-se destacar: Posix Threads (pthreads), CUDA, Boost.Threads e OpenMP. O ambiente de modelagem escolhido foi a IDE CodeLite, todas rodando no sistema operacional GNU/Linux, distribuição Ubuntu versão 10.04, 32 bits. Para os testes, o dispositivo gráfico que está sendo utilizado é uma placa de vídeo dotada de um processador nVidia da série GeForce, modelo 8400 GS, este possuindo 512 MB de memória. O computador possui um processador Intel Core 2 Duo E7400 e 4 GB de memória RAM. No quesito de compilação foi utilizado o GCC versão 4.4.3. Os controladores de vídeo são os drivers proprietários da nVidia, CUDA 3.1. RESULTADOS: Espera-se que devido ao alto grau de paralelização da arquiteturas das GPU habilitadas para a tecnologia CUDA, incluindo a integração com as linguagens C e C++, os aplicativos testados obtenham um desempenho muito superior aos algoritmos executados pela CPU. CONCLUSÕES: Comparando-se a capacidade computacional de uma GPU com a de uma CPU, o custo por flop/s se torna muito baixo, viabilizando os estudos sobre computação paralela com GPU e sua respectiva aplicação. Considerando a alta portabilidade da plataforma CUDA e sua disponibilidade em vários sistemas operacionais, será possível comparar em qual ambiente o custo é menor em relação a performance e suas divergências com as tecnologias atualmente disponíveis.

1 Projeto de pesquisa da UNIJUI com apoio FAPERGS

2 Bolsista Fapergs, aluno do curso de Ciência da Computação, da UNIJUÍ.

3 Professor do DETEC/UNIJUI orientador

4 Professor do DETEC/UNIJUI orientador