VOZES DIVERSAS
DIFERENTES SABERES

CONHECIMENTO FORMAÇÃO INOVAÇÃO Salão UFRGS 2018

SALÃO DE INICIACAO CIENTIFICA XXX SIC

15 A 19 OUTUBRO CAMPUS DO VALE

UFRGS PROPESQ

Workload Distribution in Heterogeneous Devices to improve efficiency and reliability

Gabriel Piscoya Dávila
gpdavila@inf.ufrgs.br
Orientador: Philippe Navaux

INSTITUTO DE INFORMÁTICA UFRGS

Introdução

Sistemas Heterogêneos consistem tipicamente numa CPU acompanhada de um acelerador, ambos colocados no mesmo chip. Os aceleradores mais utilizados são GPUs (Graphics Processing Units), FPGAs (Field Programmable Gate Array) e ASICs (Application Specific Integrated Circuits) desenvolvidos para mercados altamente específicos, como telecomunicações e automobilístico.

Entre os dispositivos heterogêneos disponíveis no mercado, destacamos NVIDIA Tegra X2 e as APU (Accelerated Processing Units) da AMD, ambos utilizam uma GPU acompanhada à unidade de processamento central para aumentar o desempenho de aplicações altamente paralelizáveis.

Existem dois padrões colaborativos principais de distribuição de carga: Particionamento de dados ou de tarefas. No primeiro, todas as unidades computacionais executam o mesmo algoritmo, cada uma numa porção independente dos dados. Na última, cada unidade computacional executa uma tarefa, que colabora com o dispositivo, para atingir a solução. Particionamento de dados pode ser obtido particionando os dados de entrada ou de saída.

Trabalhos já realizados mostraram que uma distribuição de carga que considera as diferentes naturezas das unidades computacionais pode aumentar significativamente o desempenho e reduzir o consumo de energia. Uma distribuição ótima deve ser realizada utilizando uma abordagem colaborativa de programação, onde a CPU não está ociosa esperando o acelerador terminar o processamento. Por tanto, a CPU deve executar partes relevantes do problema e participar colaborativamente com o acelerador para encontrar o resultado no menor tempo possível.

A execução colaborativa de um algoritmo entre a CPU e GPU pode não ter efeitos triviais na confiabilidade de um dispositivo heterogêneo. Por um lado, o uso de ambas unidades computacionais aumenta a quantidade de recursos utilizados para a computação, o que aumenta as taxas de erro. Porém, dependendo do algoritmo, o uso colaborativo de ambas unidades impacta positivamente o tempo de execução, aumentando o número de execuções corretas antes de uma falha.
Neste trabalho realizamos particionamento de dados de entrada por dois motivos principais: (1) a arquitetura do dispositivo utilizado, especialmente a memória compartilhada entre CPU e GPU e (2) na distribuição de tarefas, CPU y GPU executariam algoritmos totalmente diferentes, aumentando a dificuldade de identificar se os efeitos observados são causados pela unidade utilizada,o algoritmo ou uma combinação de ambos.

Metodologia

Primeiro foi realizada uma busca dos parâmetros que oferecem o maior desempenho em cada configuração. Para isto realizaram-se testes abrangendo todas as combinações.
A forma mais comum de avaliar a confiabilidade de dispositivos eletrônicos é mediante experimentos utilizando um feixe de nêutrons. Esses experimentos emulam os efeitos dos raios cósmicos em todas as estruturas do dispositivo, outorgando uma medida realista da taxa de erro do dispositivo (FIT, erros x 10^9 horas de funcionamento) .
Os experimentos foram realizados em Junho de 2018 nas instalações do ChipIR, localizado na Inglaterra. O fluxo experimental é cerca de 6 a 8 ordens de magnitude maior do que o fluxo de nêutrons ao nível do mar.
Os experimentos foram realizados por mais de 500 horas de tempo de feixe. Quando dimensionados para o ambiente natural, os resultados cobrem pelo menos 5108 horas de operações normais, que são 57.000 anos.

Aplicações

Selecionamos um subconjunto de aplicações do “Collaborative Heterogeneous Applications for Integrated-architectures” pertencentes a diversas classes de algoritmos: Compute-bound, Memory-bound e processamento de imagens. Nesta ocasião apresentaremos os resultados para o cálculo das superfície de Bezier. BS é um algoritmo compute-bound usado amplamente em computação gráfica.

Resultados

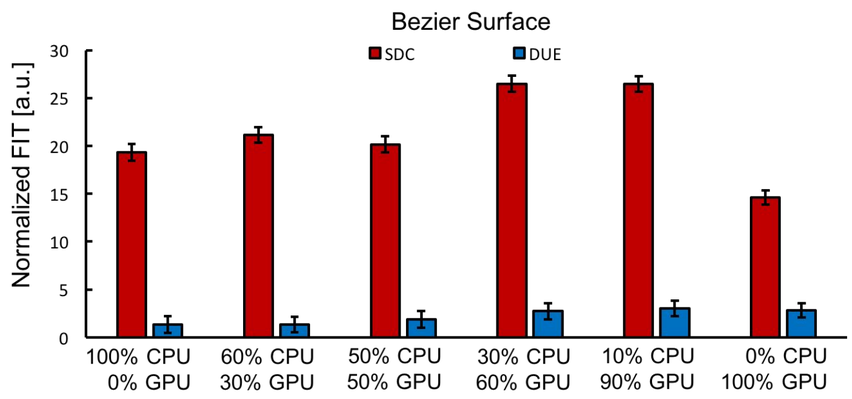

Figura 1 - FIT normalizado

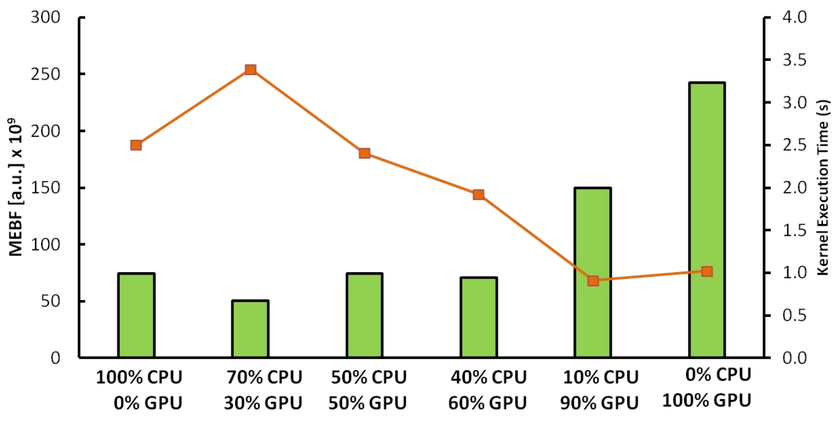

Figura 2 - MEBF e tempo de execução .

A Figura 1. Mostra o FIT para o BS, podemos observar que a GPU apresenta uma taxa de erros menor que a CPU, isto é devido à arquitetura das unidades, a GPU tem um Architectural Vulnerability Factor (AVF) 3 vezes menor que a CPU. Por outro lado, ao realizar uma distribuição de carga podemos observar que a taxa de erros aumenta, como era de esperar, utilizar ambas unidades computacionais aumenta a quantidade de recursos sendo expostos.
Na Figura 2. Pode-se observar que a configuração que oferece maior desempenho é a distribuição de 90% do trabalho na GPU. O aumento significativo no FIT devido aos recursos adicionais utilizados para realizar a computação não são compensados pelo pequeno ganho de desempenho. Assim quando executamos o algoritmo numa configuração de 100% na GPU obtemos o maior MEBF.

Conclusão

Este trabalho apresenta uma avaliação da confiabilidade da distribuição de carga de trabalho em dispositivos heterogêneos através de experimentos de radiação.
Mostra-se que, a distribuição de carga afeta a confiabilidade do dispositivo, além que existe um “trade-off” entre os ganhos de desempenho e o aumento de recursos necessários para a computação. Ambos devem ser levados em consideração para conseguir a maior quantidade de execuções corretas antes de uma falha acontecer.