DOI: http://dx.doi.org/10.29069/forscience.2016v4n1.e164

AVALIAÇÃO DE DESEMPENHO E CONSUMO ENERGÉTICO PARA CONFIGURAÇÕES DE *WAVEFRONT POOLS* DE UMA GPU AMD

Ariel Gustavo Zuquello[1]
Emanuel Felipe Duarte[2]
Lucas Pupulin Nanni[3]
Rômulo de Aguiar Beninca[4]
Yoji Massago[5]

RESUMO

O uso de sistemas heterogêneos CPU-GPU para atender à crescente demanda por aplicações com grande paralelismo de dados resulta na necessidade de estudar e avaliar tais arquiteturas para melhorá-las continuamente. Neste artigo foram feitas simulações da execução de uma suíte de *benchmark* em uma GPU *AMD ATI Radeon*™ HD 7970, de modo a avaliar o impacto sobre o desempenho e o consumo energético quando alterado o número de *Wavefront Pools* presentes em cada *compute unit* da GPU, que é 4 por padrão. O resultado mais significante evidencia um aumento de velocidade de cerca de 5,7% para a configuração com duas *Wavefront Pools* em conjunto com um aumento no consumo de energia de cerca de 5,1%. Todavia, as outras configurações avaliadas também representam opções para diferentes tipos de necessidades, conforme a categoria de demanda computacional.

Palavras-chave: Sistemas heterogêneos. Simulações. Desempenho.

[1] Mestre em Ciência da Computação pela Universidade Estadual de Maringá (UEM). Atualmente é professor colaborador da Universidade do Estado de Santa Catarina (UDESC) Campus Chapecó e da Universidade Federal da Fronteira Sul (UFFS) Campus Erechim. (http://lattes.cnpq.br/7843368686233904). E-mail: arielzuquello@gmail.com.

[2] Doutorando em Ciência da Computação pela Universidade Estadual de Campinas (UNICAMP). Mestre em Ciência da Computação pela UEM e graduado em Tecnologia em Sistemas para Internet pelo Centro Universitário de Maringá, CESUMA. (http://lattes.cnpq.br/3718325359953351). E-mail: contato@emanuelfelipe.net.

[3] Mestre e graduado em Ciência da Computação pela UEM, atualmente é professor assistente na mesma instituição e atua no Grupo de Sistemas Interativos Inteligentes do CTC/DIN nas linhas de Recuperação Adaptativa de Informação na Web, Visualização de Informação na Web e Busca Exploratória. (http://lattes.cnpq.br/5145451839280335). E-mail: lucasnanni@gmail.com.

[4] Mestre em Ciência da Computação pela UEM, atualmente professor do Instituto Federal de Santa Catarina (UFSC), atuante nas atividades de pesquisa, ensino e extensão. (http://lattes.cnpq.br/7486046766117014). E-mail: rbeninca@gmail.com.

[5] Mestre e graduado em Ciência da Computação pela UEM. (http://lattes.cnpq.br/9137805402174351). E-mail: yojimassago@gmail.com.

1 INTRODUÇÃO

Nos últimos anos, as *Graphics Processing Units* (GPUs) ganharam grande destaque na área da computação de alto desempenho. Cada vez mais os sistemas heterogêneos CPU-GPU são utilizados para atender a crescente demanda de aplicações com paralelismo de dados, como é o caso de processamento de imagens (ASANO *et al.,* 2009), aplicações destinadas ao mercado financeiro (GRAUER-GRAY *et al.,* 2013), programas de simulação de modelos moleculares dinâmicos (YANG *et al.,* 2007) e projetos de sequenciamento de *DeoxyriboNucleic Acid* (DNA) (TUMEO; VILLA, 2010). Em todos os exemplos supracitados, e muitos outros, a GPU emerge como um acelerador integrado para processamento paralelo de dados.

Como consequência do evidente crescimento da demanda por computação heterogênea CPU GPU, cresce também a necessidade mercadológica de se estudar e avaliar tais arquiteturas com o objetivo de melhorá-las continuamente (BAKHODA *et al.,* 2009). Entretanto, ao passo que novos projetos arquiteturais de CPU e GPU são propostos, surge também uma grande quantidade de possíveis combinações dessas arquiteturas, e inevitavelmente, gerar protótipos para cada combinação proposta pode se mostrar uma tarefa árdua e de alto custo. Ou seja, conforme surgem novas ideias de abordagens arquiteturais, a avaliação física das combinações resultantes pode rapidamente se revelar uma tarefa inviável do ponto de vista comercial.

Dessa forma, um ambiente capaz de simular arquiteturas heterogêneas CPU-GPU por softwares elimina a necessidade de se criar inúmeros protótipos, se mostrando fundamental na tarefa de viabilizar a avaliação de uma grande quantidade de novas abordagens arquiteturais. As inúmeras combinações provenientes de diferentes parâmetros e abordagens de *design* podem ser avaliadas de forma automatizada, permitindo os mais variados níveis de detalhe, e com um custo relativamente baixo quando comparado a fabricar inúmeros protótipos e os avaliar individualmente.

A proposta deste artigo consiste em avaliar o desempenho e o consumo de energia de uma arquitetura heterogênea CPU-GPU em um ambiente de simulação que permita verificar o impacto ocasionado por um conjunto de modificações realizadas no projeto da GPU. O ambiente escolhido foi o *Multi2Sim* (UBAL *et al.,* 2012) por sua capacidade de simular a arquitetura abordada neste artigo. A avaliação por sua vez foi baseada na execução de alguns

programas da *Polybench-GPU* (GRAUER-GRAY *et al.,* 2012), uma suíte de *benchmark* capaz de explorar o paralelismo computacional fornecido pela GPU. As modificações realizadas consistem da alteração do número de *Wavefront Pools* presentes em cada compute *unit* pertencente à GPU. O objetivo de tais modificações é explorar diferentes organizações dos *Wavefronts* armazenados pelas *Wavefront Pools* e o impacto na performance e consumo de energia ocasionado pelas mesmas.

A motivação para essa avaliação é a necessidade de se avaliar arquiteturas heterogêneas CPU-GPU com a finalidade de melhorá-las continuamente, avançando o estado da arte. Tal melhoria, por sua vez, é motivada pela crescente demanda por aplicações com grande paralelismo de dados.

A contribuição deste artigo é que ao simular alterações no parâmetro *Wavefront Pools* se obteve evidências de aumento de desempenho de cerca de 5,7% com um aumento do consumo energético de cerca de 5,1%. Isso evidencia que com um maior número de simulações pode haver a descoberta de melhoras significativas no desempenho ou na redução do consumo energético, permitido que sejam encontradas configurações com otimização ideal desse parâmetro para diferentes necessidades computacionais.

O trabalho está estruturado da seguinte forma: A Seção 2 apresenta o ambiente experimental avaliando a arquitetura existente e as modificações realizadas na mesma, análise das configurações realizadas e também as principais dificuldades encontradas. A Seção 3 apresenta os dados referentes a execução dos testes e uma avaliação focada no desempenho e consumo de energia das configurações simuladas. A Seção 4 explora possíveis trabalhos futuros. Por fim, as conclusões são apresentadas na Seção 5.

2 AMBIENTE EXPERIMENTAL

A metodologia do artigo é baseada na experimentação e na avaliação da execução de uma suíte de *benchmark* em um ambiente de simulação configurado com uma arquitetura computacional heterogênea CPU-GPU. O experimento consiste na obtenção dos relatórios gerados pelo simulador a partir das diversas execuções de cada aplicação da suíte de *benchmark*, cada uma sob efeito de algumas modificações selecionadas para a arquitetura da GPU. As modificações realizadas resultam em alterar o número de *Wavefront Pools* presentes em cada *compute unit* da GPU.

Uma *Wavefront Pool* pode ser descrita como uma unidade destinada ao armazenamento de *Wavefronts*, ou *work-groups*, os quais são constituídos de 64 *work-itens* que executam o mesmo conjunto de instruções. Dessa forma, a *Wavefront Pool* pode ser entendida como um local onde as instruções aguardam serem buscadas por cada unidade *Single Instruction Multiple Data* (SIMD) (MULTI2SIM, 2013) presente na GPU. Conforme pode ser observado na Figura 1, a unidade de busca coleta instruções de cada *Wavefront Pool* em uma sequência cíclica determinada por um algoritmo de escalonamento de *Wavefronts*.

A avaliação consiste na análise de desempenho e consumo energético da arquitetura simulada de acordo com cada modificação realizada. A infraestrutura de avaliação, incluindo a suíte de *benchmark* executada, o ambiente de simulação utilizado, bem como as adaptações necessárias para possibilitar a realização do estudo são discutidas nas subseções seguintes.

2.1 Infraestrutura de avaliação

A infraestrutura de avaliação consiste da execução da suíte de *benchmark Polybench-GPU* no ambiente de simulação *Multi2Sim*, modelado com uma arquitetura heterogênea especificada para assemelhar-se a uma CPU *IntelR Core*^{TM} i7-3820 da arquitetura *Sandy Bridge* (INTEL, 2013) associado a uma GPU *AMD Radeon*^{TM} HD 7970 *Graphics* da arquitetura *Southern Islands* (AMD, 2013). Os parâmetros utilizados para assemelhar as arquiteturas avaliadas àquelas dos modelos supracitados são provenientes das informações oficiais fornecidas pelo fabricante e de consulta bibliográfica (HENNESSY; PATTERSON, 2012) para a CPU, e configurações presentes no simulador *Multi2Sim*, como arquivos de exemplo para a arquitetura Southern Islands, para a GPU (FIG. 1).

A escolha do *Polybench-GPU* como suíte de *benchmark* a ser executada se deve a sua capacidade diferenciada de explorar uma arquitetura heterogênea CPU-GPU de modo a executar os mesmos fluxos de trabalho tanto para a CPU quanto para a GPU, evidenciando dessa forma as possíveis diferenças de desempenho. Já a escolha do *Multi2Sim* se deve a sua capacidade de simular arquiteturas heterogêneas CPU-GPU e gerar relatórios referentes a simulação. As especificações de CPU e GPU analisadas foram tomadas de maneira que correspondessem com a tecnologia disponível no mercado.

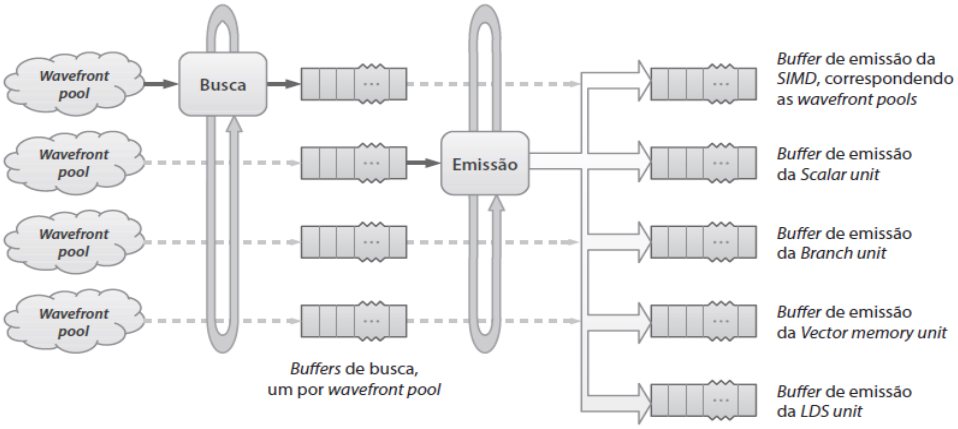

Figura 1 - Diagrama do *front-end* da GPU, com destaque para as *Wavefront Pools*
Fonte: Multi2sim (2013).

2.1.1 Polybench-GPU

A suíte de *benchmark Polybench-GPU* é uma modificação diretamente baseada na *Polybench* 2.0 (POUCHET, 2013), a qual foi originalmente escrita em C. Entretanto, a *Polybench-GPU* se trata de uma conversão destinada a explorar também a GPU com o uso de *Compute Unified Device Architecture* (CUDA) (NVIDIA, 2013), *Open Computing Language (OpenCL*) (KHRONOS GROUP, 2013), que é a opção utilizada neste estudo, ou *Hybrid Multicore Parallel Programming* (HMPP) (CAPS, 2012). O uso da GPU para as mais diversas finalidades deu origem ao termo *General Purpose Graphics Processing Unit* (GPGPU), destinado a representar de maneira mais precisa o novo contexto em que se encontram as GPUs. Os programas da *Polybench-GPU* executados são descritos na TAB. 1.

Tabela 1 - Programas da suíte de benchmark Polybench-GPU

Programa	Descrição
2MM	Multiplicação entre as matrizes A e B.
3MM	Multiplicação entre as matrizes A, B, C e D.
ATAX	Multiplicação entre a matriz transposta de A e o vetor V.
BICG	*Sub kernel* BiCG do *solver* linear BiCGStab.
CORRELATION	Computação de correlação entre dados.
COVARIANCE	Computação de covariância entre dados.
FDTD-2D	*Kernel* de domínio de tempo de diferenças finitas 2D.
GEMM	Multiplicação de matrizes.
GESUMMV	Multiplicação de escalar, vetor e matriz.
GRAMSCHMIDT	Processo de Gram-Schmidt para ortogonalização de vetores.
MVT	Multiplicação entre uma matriz e um vetor, seguido de transposição.
SYR2K	Operações simétricas *rank-2K* com matrizes.
SYRK	Operações simétricas *rank-k* com matrizes.

Fonte: Adaptado de Benchmarks (Polybench/c version 3.2) (2013[1]).

2.1.2 Multi2Sim

O *Multi2Sim* é um *framework* (ou ambiente) de simulação para computação heterogênea CPU-GPU escrito em C. Ele é capaz de simular modelos de CPU superescaleres, *multithread* e *multicore*, bem como arquiteturas de GPU. Além disso, ele permite que a simulação seja conduzida tanto de maneira funcional, sem nenhum detalhamento especial além de exibição de erros e tempo de simulação, quanto detalhada, situação onde podem ser gerados relatórios da execução com um grande nível de detalhamento.

Para este artigo, o ambiente de simulação foi configurado com especificações similares à CPU *IntelR Core*^{TM} i7-3820 e à GPU *AMD Radeon*^{TM} HD 7970, a fim de se obter um sistema relevante à uma tecnologia recente. Dessa forma, as simulações foram conduzidas utilizando o modo “detalhado", permitindo que relatórios de execução da CPU e GPU fossem emitidos com todos os detalhes necessários para a avaliação da arquitetura modelada.

2.1.3 McPAT

O *Multicore Power, Area, and Timing* (McPAT) (LI *et al.*, 2009) é descrito como um *framework* de modelagem de potência, energia, área ocupada pelo *chip* e tempo de execução para arquiteturas *multithread*, *multicore* e *manycore*. Ele é capaz de modelar a potência, a área funcional do *chip* e o tempo de execução simultaneamente de maneira consistente para diversos projetos de arquitetura. O *McPAT* fornece a possibilidade de especificar todos os componentes para um *chip* completo, incluindo núcleos com processamento em ordem e fora de ordem, redes *on-chip*, *caches* compartilhadas e controladores integrados de memória.

2.2 Metodologia de execução e análise

Resgatando o objetivo de avaliar o desempenho e o comportamento do consumo energético da GPU especificada sob o efeito da modificação da quantidade de *Wavefront Pools* presentes em cada *compute unit* da GPU, apresentamos a seguir a metodologia aplicada para a simulação da execução da suíte de *benchmark* e análise dos dados gerados pelo mesmo. Para avaliar o impacto do número de *Wavefront Pools* no desempenho e no consumo energético da GPU, foram geradas cinco configurações distintas compostas de uma

configuração padrão: duas modificações com aumento no número de *Wavefront Pools* (uma com o dobro e outra com o quádruplo) e duas modificações com redução no número de *Wavefront Pools* (uma com a metade e outra com um quarto). A configuração padrão consiste do arquivo de configuração da GPU fornecido ao *Multi2Sim* com o parâmetro *NumWavefrontPools* associado ao valor 4: *NumWavefrontPools* = 4 (configuração padrão WFP4). Deste modo, as quatro variações foram criadas, cada uma, com os seguintes valores: *NumWavefrontPools* = 1 (WFP1), *NumWavefrontPools* = 2 (WFP2), *NumWavefrontPools* = 8 (WFP8) e *NumWavefrontPools* = 16 (WFP16).

A suíte de *benchmark* foi executada no simulador para cada configuração gerada, resultando na emissão dos relatórios de estatísticas da CPU e GPU para cada programa da *Polybench* GPU. Esses relatórios trazem estatísticas referentes a quantidade de ciclos realizados pelas unidades de processamento, bem como a quantidade e o tipo de instruções processadas pelas mesmas. Após a emissão dos relatórios da simulação, o McPAT foi aplicado individualmente às estatísticas de cada programa da suíte de *benchmark*, para cada configuração simulada. Por fim, os relatórios de consumo energético emitidos pelo McPAT foram adicionados aos outros relatórios, constituindo dessa forma a base de estatísticas utilizada para a análise e avaliação das modificações realizadas na GPU.

As estatísticas analisadas por este artigo se restringem ao número de ciclos, à taxa de instruções executadas por ciclo (IPC) e energia consumida pela CPU e GPU para cada programa da suíte de *benchmark*. A proposta da análise das estatísticas não é mensurar os valores absolutos obtidos com as simulações, mas sim obter uma relação de ganho e perda entre as configurações consideradas. Dessa forma, espera-se que a avaliação do impacto das modificações na quantidade de *Wavefront Pools* seja realizada sem ser comprometida pela validade absoluta dos dados experimentados.

A análise foi realizada por diferentes aspectos, identificando os possíveis *speedups* e *slowdowns* de desempenho e ganhos e perdas no consumo energético ocasionados pelas modificações realizadas. Partindo de um aspecto granular mais fino, no qual as estatísticas foram analisadas em nível de programa da suíte de *benchmark*, é possível verificar o comportamento individual das simulações para cada modificação implementada. Já em um aspecto mais geral, de granularidade mais grossa, as mesmas estatísticas foram analisadas, mas dessa vez em nível de configuração, estabelecendo uma relação entre as mesmas.

3 AVALIAÇÃO E DISCUSSÃO

Uma vez executada a suíte de *benchmark* para cada uma das configurações geradas, as estatísticas discutidas pela Seção anterior foram obtidas dos relatórios de simulação, e os dados relevantes foram relacionados em gráficos para uma melhor visualização dos dados.

De maneira geral não foram observadas alterações relevantes nas estatísticas obtidas da CPU. Para todos os programas da suíte de *benchmark* a relação de instruções por ciclo e alterações no consumo de energia obtiveram variações inferiores à 0.05%. Os dados sugerem que as alterações no número de *Wavefront Pools* realizadas não ocasionam alterações relevantes no comportamento da CPU em relação a configuração padrão WFP4.

A FIG. 2 ilustra o gráfico construído a partir do número de ciclos consumidos pela execução de cada programa da *Polybench*-GPU, agrupados pelas diferentes configurações simuladas. Os valores ilustrados referem-se ao percentual de ciclos amostrados em relação a configuração padrão WFP4. É possível verificar o impacto das modificações WFP1, WFP2, WFP8 e WFP16 sobre a quantidade de ciclos consumidos pela execução de cada programa da suíte de *benchmark*. Percebe-se como este impacto é dependente não só da configuração utilizada, mas também da característica de cada programa executado. Essa constatação pode ser notada, por exemplo, para os programas 2MM, 3MM e GEMM, os quais apresentaram valores análogos possivelmente por desempenharem tarefas semelhantes (todos realizam multiplicações de matrizes).

Ademais, foi possível observar os *speedups* e *slowdonws* resultantes das modificações realizadas na arquitetura da GPU. A configuração com maior *speedup* muda de acordo com cada programa, então, foi possível analisar que a modificação WFP2 ocasionou o maior número de *speedups* e também os mais significativos em relação a configuração padrão WFP4, sendo superada em casos específicos apenas por uma proporção marginal. Com a configuração WFP2, para os programas da suíte de *benchmark* 2MM, 3MM, GEMM e GRAMSCHIMIDT os *speedups* atingiram até 10% em relação à WFP4. Em contrapartida, a configuração WFP16 ocasionou *slowdowns* em todos os programas, atingindo uma redução de desempenho de até 60%, 75% e 65% para os programas 2MM, 3MM e GEMM respectivamente.

O gráfico ilustrado pela FIG. 3 apresenta uma visão alternativa aos dados discutidos anteriormente. A métrica de IPC permite observar de forma mais concisa os *speedups* e

slowdowns adquiridos com as modificações na arquitetura da GPU. Para essa estatística é possível observar novamente a modificação WFP2 como a que ocasionou maiores *speedups*, atingindo um ganho de até 15% para o programa GRAMSCHMIDT e entre 10% e 12% para os programas 2MM, 3MM e GEMM. Por outro lado, mas com constatação análoga à métrica de ciclos, a modificação WFP16 foi a que provocou maior redução de desempenho, atingindo *slowdowns* de mais de 40% para o programa 3MM, 40% para o programa GEMM e até 35% para os programas 2MM e GRAMSCHMIDT. Mais uma vez, observa-se que o impacto no desempenho é dependente tanto da modificação realizada quanto da característica de cada programa executado.

Outro aspecto analisado por este artigo é o impacto referente ao consumo energético ocasionado pelas modificações arquiteturais realizadas. A FIG. 4 ilustra o gráfico de porcentagem de consumo de energia de cada programa da suíte de *benchmark*, para cada modificação, em relação a configuração base WFP4. De maneira análoga à análise da estatística de IPC realizada anteriormente, a modificação WFP2 provocou um aumento no consumo energético (11 casos), atingindo entre 10% e 14% para os programas 2MM, 3MM, GEMM e GRAMSCHMIDT. Ainda analogamente, a modificação WFP16 resultou em uma redução no consumo de energia para todos os programas executados, atingindo entre 58% e 70% do consumo energético da configuração padrão WFP4 para os programas 2MM, 3MM, GEMM e GRAMSCHMIDT.

A relação de analogia entre o consumo de energia e a IPC é evidenciada pela Figura 5, onde essas estatísticas são apresentadas em um gráfico de dispersão de pontos. O arranjo linear dos pontos sugere que o consumo de energia é diretamente proporcional ao número de instruções executadas por ciclo, o que geralmente não é constatado em sistemas reais. Tal comportamento pode ser atribuído a baixa especificidade das estatísticas dinâmicas repassadas ao McPAT, o qual gerou os valores de consumo baseado apenas no número de ciclos e de instruções executadas. Outro detalhe importante é o fato de que o McPAT não considera aspectos particulares de uma GPU, o que pode resultar em estatísticas com menor precisão (GOSWAMI, 2012). O gráfico ilustrado pela Figura 6 destaca essa mesma relação linear observada, porém com uma visão resumida para as diferentes configurações ao utilizar a média geométrica das razões obtidas para cada programa da suíte de *benchmark*.

De maneira geral, as modificações realizadas ocasionaram tanto *speedups* quanto *slowdonws* em relação à configuração padrão WFP4. Como avaliado anteriormente, a redução

do número de *Wavefront Pools* para 2 proporcionou *speedup* de IPC para a maioria dos *benchmarks*, porém provocou um aumento no consumo de energia para todas as situações. As FIG. 5 e 6 mostram um comportamento diretamente proporcional entre o IPC e o consumo de energia, um comportamento divergente com o que é observado em sistemas reais, os quais normalmente estabelecem uma curva de crescimento exponencial entre o consumo de energia e o desempenho (HENNESSY; PATTERSON, 2012).

Por não se tratar de um fator altamente restritivo para GPUs de *desktop*, o aumento de consumo de energia ocasionado pela modificação WFP2 pode ser encarado como um *trade-off* viável para a produção de unidades gráficas de alto desempenho com tal configuração. Outro ponto a ser observado é a possibilidade de redução da área e da complexidade do circuito integrado da GPU devido ao menor número de unidades de *Wavefront* Pools implementadas, reduzindo assim o custo de produção do mesmo.

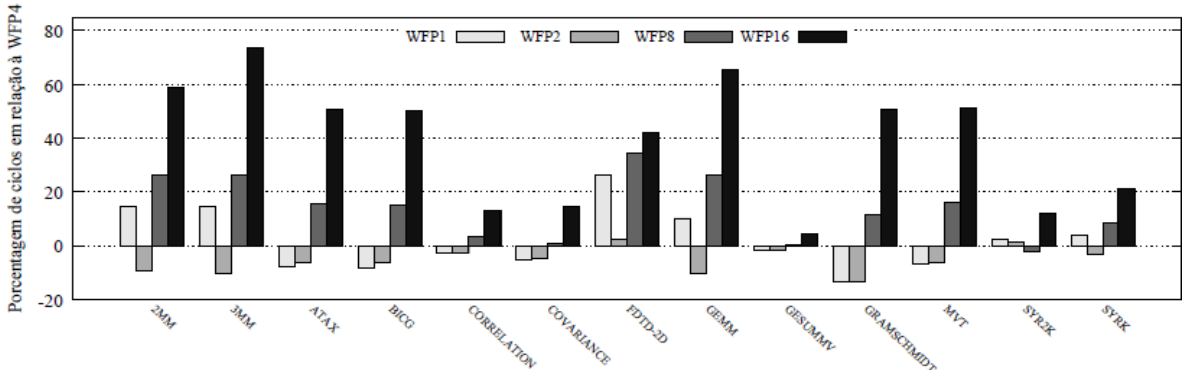

Figura 2 - Porcentagem do número de ciclos amostrados de cada programa da suíte de *benchmark*, para cada modificação em relação à WFP4
Fonte: Zuquello *et al*. (2013).

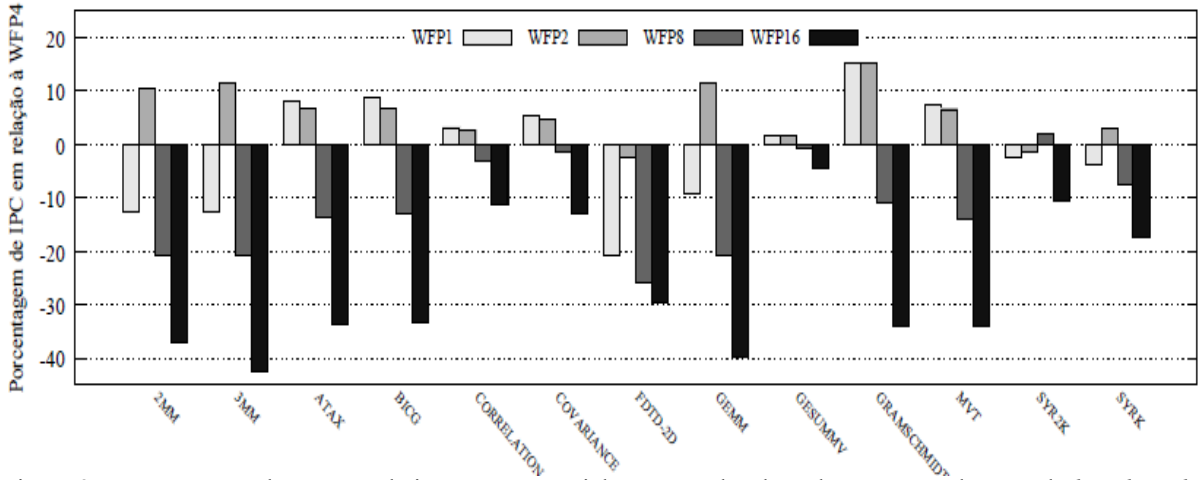

Figura 3 - Porcentagem do número de instruções por ciclo amostradas de cada programa da suíte de *benchmark*, para cada modificação em relação à WFP4
Fonte: Zuquello *et al*. (2013).

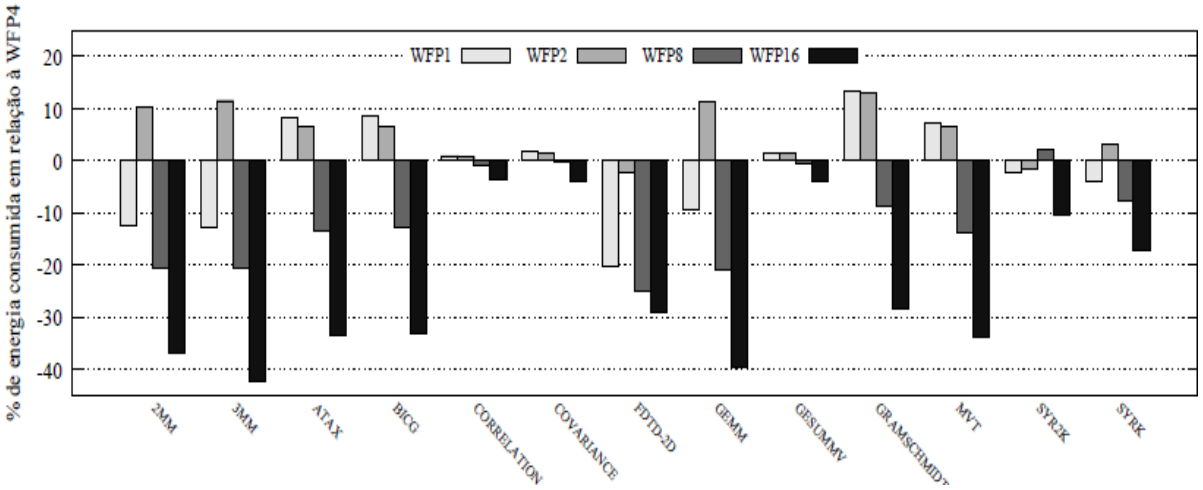

Figura 4 - Porcentagem do consumo de energia amostrado de cada programa da suíte de *benchmark*, para cada modificação em relação à WFP4
Fonte: Zuquello *et al*. (2013).

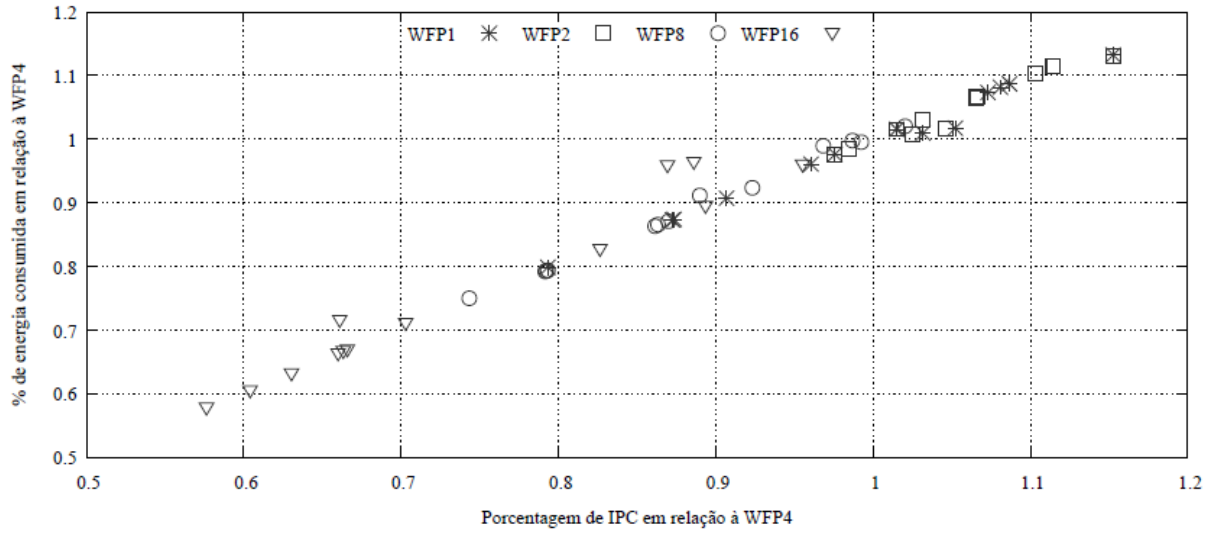

Figura 5 - Relação entre as métricas de IPC e consumo de energia para cada programa da suíte de *benchmark*
Fonte: Zuquello *et al*. (2013).

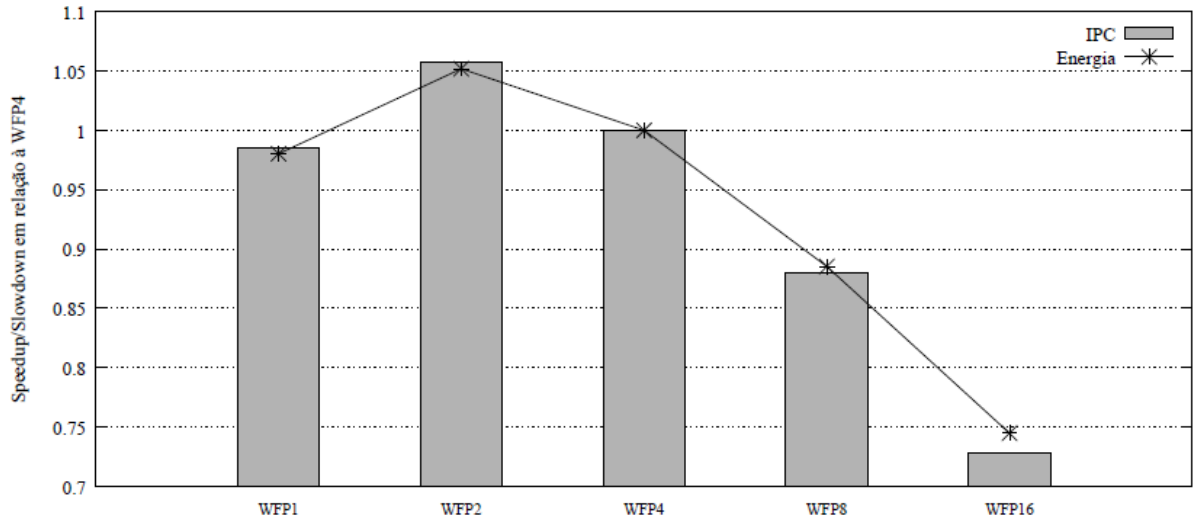

Figura 6 - Relação entre as métricas de IPC e consumo de energia sumarizado para cada configuração a partir da média geométrica das porcentagens dos programas da suíte de *benchmark*
Fonte: Zuquello *et al*. (2013).

4 Trabalhos futuros

A exploração de arquiteturas heterogêneas CPU-GPU para aplicações com alto paralelismo de dados é uma área ainda relativamente nova e, consequentemente, ainda pouco explorada, entretanto muito promissora e repleta de oportunidades para novos trabalhos. As subseções a seguir mostram alguns dos aspectos relacionados a este trabalho identificados pelos autores e que possivelmente permitem trabalhos futuros.

4.1 Simulador

Considerando primeiramente o simulador, o *Multi2Sim* é um software eficiente e bem projetado, entretanto ainda assim ele pode ser ampliado para suportar novas arquiteturas, possuir suporte a mais instruções, tipos de dados e chamadas *OpenCL* que ainda estão faltando. Tais melhorias permitiriam a simulação de uma gama ainda maior de programas ampliando as possibilidades de pesquisas a serem realizadas com este simulador.

4.2 Suíte de *benchmark*

Do ponto de vista da suíte de *benchmark Polybench-GPU*, existe margem para a correção de problemas notados como inconsistência entre os tipos de dados utilizados pelos programas e pequenos erros dentro do código. Outro aspecto interessante é o fato de que seria proveitoso se a suíte de *benchmark* fosse projetada de modo a facilitar a automatização de sua execução inúmeras vezes para diferentes configurações.

4.3 Ferramenta de análise de consumo energético

A ferramenta de análise de consumo energético utilizada, *McPAT*, mostrou-se bastante limitada no aspecto de medir o consumo energético de uma GPU (que o mesmo considera como uma CPU qualquer). A possibilidade de considerar os aspectos particulares de uma GPU provavelmente resultaria em uma análise mais precisa do consumo de energia da mesma, possibilitando dados absolutos mais próximos da realidade.

4.4 Diferentes abordagens de arquiteturas

Existe uma grande quantidade de possíveis alterações no projeto arquitetural de uma GPU, muitas das quais com grande potencial de resultar em combinações mercadologicamente interessantes da relação entre desempenho, consumo energético e custo. A automação do processo de modificar uma característica arquitetural, executar simulações e comparar os resultados obtidos pode facilitar em grande escala o processo de descoberta de novas combinações com potencial de exploração comercial.

5 CONCLUSÃO

Neste artigo foi possível evidenciar a importância da GPU como um acelerador integrado para processamento paralelo de dados para as mais diversas finalidades. Neste contexto, simular arquiteturas heterogêneas CPU-GPU por softwares emerge como uma alternativa de viabilizar a comparação entre inúmeras combinações de configurações de

projeto, passo importante no estudo e avaliação de tais arquiteturas com o objetivo de melhorá-las continuamente.

Com a modificação no número de *Wavefront Pools* presentes em cada *compute unit* da GPU, observou-se que essas alterações resultaram em uma relação inversamente proporcional entre consumo energético e desempenho. Entre todas as modificações, a WFP2 mostrou-se a mais interessante de acordo com o propósito de alto desempenho da GPU *AMD RadeonTM* HD 7970 *Graphics*, por possuir um *trade-off* viável entre desempenho e consumo energético.

PERFORMANCE EVALUATION AND ENERGY CONSUMPTION FOR SETTINGS OF WAVEFRONT POOLS OF A GPU AMD

ABSTRACT

The use of CPU-GPU heterogeneous systems to meet the growing demand for applications with large data parallelism results in the need to study and evaluate these architectures in order to improve them continuously. In this paper we made simulations of running a benchmark suite on an AMD GPU ATI Radeon™ HD 7970 in order to assess the impact on performance and power consumption when tuning the number of Wavefront Pools present in each GPU compute unit, which is 4 by default. The most significant result shows a speedup of about 5.7% for configuration with two Wavefront Pools in conjunction with an increase of about 5.1% in the energy consumption. However, the other evaluated configuration also represent options for different kinds of needs, according to the computational demand.

Keyworks: Heterogeneous systems. Simulation. Performance.

REFERÊNCIAS

ADVANCED MICRO DEVICES. **AMD Radeon™ HD 7970 Graphics**. [©2013]. Disponível em: <http://www.amd.com/enus/products/graphics/desktop/7000/7900>. Acesso em: 13 maio 2013.

ASANO, S.; MARUYAMA, T.; YAMAGUCHI, Y. Performance comparison of FPGA, GPU and CPU in image processing. In: INTERNATIONAL CONFERENCE ON FIELD PROGRAMMABLE LOGIC AND APPLICATIONS, 19., 2009. Prague, Czech Republic. **Proceedings**... Prague: [s.n.], p. 126-131, 2009. Disponível em: <http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5272532>. Acesso em: 13 maio 2013.

BAKHODA, A. *et al*. Analyzing CUDA workloads using a detailed GPU simulator. In: IEEE INTERNATIONAL SYMPOSIUM ON PERFORMANCE ANALYSIS OF SYSTEMS AND SOFTWARE (ISPASS)., 2009. **Proceedings…** [S.l.: s.n.] 2009. p. 163–174, 2009. Disponível em: <http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4919648>. Acesso em: 13 maio 2013.

CAPS. Openhmpp directives. **CAPS OpenACC Compiler**: HMPP workbench 3.2, 2012. Disponível em: < https://www.olcf.ornl.gov/wp-content/uploads/2012/10/HMPPOpenACC-3.2_ReferenceManual.pdf>. Acesso em: 13 maio 2013.

GOSWAMI, N. **GPU-PowerSim**. 2012. Disponível em: <http://www.nilanjan.info/softwares/gpu-powersim/copyright/>. Acesso em: 11 maio 2013.

GRAUER-GRAY, S. *et al*. Accelerating financial applications on the GPU. In: WORKSHOP ON GENERAL PURPOSE PROCESSING USING GPUS. 6, 2013, Houston, TX. [**Proceedings]**... Houston, TX., 2013. Disponível em: <https://cavazos-lab.github.io/FinanceBench/resources/AcceleratingFinancialApplicationsOnTheGPU-paper.pdf>. Acesso em: 28 abr. 2013.

GRAUER-GRAY, S. *et al*. Auto-tuning a high-level language targeted to GPU codes. In: INNOVATIVE PARALLEL COMPUTING, 2012, San Jose, California, USA. **Proceedings**... San Jose: IEEE, 2012. Disponível em: <https://www.eecis.udel.edu/~cavazos/autoTuneGpu.pdf>. Acesso em: 28 abr. 2013.

HENNESSY, J. L.; PATTERSON, D. A. **Computer architecture**: a quantitative approach. 5th. ed. Waltham, MA: Morgan Kaufman, 2012.

INTEL. **Intel^{R} Core^{TM} i7-3820 Processor (10MB cache, UP to 3.80 GHZ)**. Santa Clara, CA, USA: Intel Corporation, [2013]. Disponível em: <http://ark.intel.com/products/63698/Intel-Core-i7-3820-Processor-10M-Cache-up-to-3_80-GHz>. Acesso em: 29 abr. 2013.

KHRONOS GROUP. **Opencl**: the open standard for parallel programming of heterogeneous systems. Beaverton, OR, USA, [2013]. Disponível em: <https://www.khronos.org/opencl/>. Acesso em: 04 maio 2013.

LI, S. *et al*. McPAT: an integrated power, area, and timing modeling framework for multicore and manycore architectures. In: ANNUAL IEEE/ACM INTERNATIONAL SYMPOSIUM ON MICROARCHITECTURE, 42., 2009, New York. **Proceedings…** New York, NY, USA: ACM, p. 469-480, 2009. Disponível em: <http://cseweb.ucsd.edu/~tullsen/micro09b.pdf>. Acesso em: 04 maio 2013.

MULTI2SIM. **The Multi2Sim simulation framework**. 2013. Disponível em: <https://www.multi2sim.org/files/multi2sim-r277.pdf>. Acesso em: 10 abr. 2013.

NVIDIA. **CUDA Parallel programming and computing platform**. NVIDIA Corporation, [2013]. Disponível em: <http://www.nvidia.com/object/cuda_home_new.html>. Acesso em: 10 abr. 2013.

POUCHET, L. **Polybench**: the polyhedral benchmark suite. Disponível em: <http://web.cs.ucla.edu/~pouchet/software/polybench/>. Acesso em: 11 abr. 2013.

TUMEO, A.; VILLA, O. Accelerating DNA analysis applications on GPU clusters. In: SYMPOSIUM ON APPLICATION SPECIFIC PROCESSORS, 8., 2010, Anaheim, CA, USA. **Proceedings**... [S.l.:s.n.], p. 71–76, 2010. Disponível em: <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=5521145>. Acesso em: 11 abr. 2013.

UBAL, R. *et al*. Multi2sim: a simulation framework for CPU-GPU computing. In: INTERNATIONAL CONFERENCE ON PARALLEL ARCHITECTURES AND COMPILATION TECHNIQUES, 21., 2012, Minneapolis, MN, USA. **Proceedings**… New York: ACM, 2012. Disponível em: <https://www.multi2sim.org/files/conferences/pact-2012.pdf>. Acesso em: 11 abr. 2013.

YANG, J.; WANG, Y.; CHEN, Y. GPU Accelerated Molecular Dynamics Simulation of Thermal Conductivities. **Journal of Computational Physics**, v. 221, n. 2, p. 799-804, 2007. Disponível em: <http://dl.acm.org/citation.cfm?id=1224606>. Acesso em: 11 abr. 2013.

Recebido em: 13/01/2016

Aprovado em: 05/05/2016

Publicado em: 05/07/2016