Utilização de aceleradores embarcados de baixo consumo na implementação de sistemas de HPC *

Emilio Hoffmann de O.[1], Jorge X. Silva Jr.[2], Edson L. Padoin[1,2], Philippe O. A. Navaux[2]

[1]Universidade Reg. do Noroeste do Estado do Rio G. do Sul (UNIJUI) - Ijuí - RS - Brasil

emiliohoffmann@hotmail.com, padoin@unijui.edu.br

[2]Universidade Federal do Rio Grande do Sul (UFRGS) - Porto Alegre - RS - Brasil

jorge.junior0124@gmail.com, navaux@inf.ufrgs.br

Abstract. *This work aims to analyze the performance and energy efficiency of low-power embedded accelerators to implement HPC systems due current power consumption limitations. Tests using the* 3 *levels of SHOC benchmark were performed on* 5 *NVIDIA conventional GPUs accelerators and a low-power accelerator embedded on MPSoC Jetson. Conventional accelerators such as the NVIDIA K80 achieved performance of up to* 3750 *GFLOPS and energy efficiency of* 25 *GFLOPS/W, whereas with the low power accelerator TK1 was obtained performances of only* 301 *GFLOPS and a higher energy efficiency, equivalent to* 26, 2 *GFLOPS/W.*

Resumo. *Este trabalho tem como objetivo analisar o desempenho e a eficiência energética de aceleradores embarcados de baixo consumo para implementação de sistemas de HPC frente às atuais recomendações de consumo estabelecidas. Testes foram realizados utilizando os 3 níveis do benchmark SHOC em 5 aceleradores GPUs convencionais da NVIDIA e em um acelerador de baixo consumo embarcado na placa MPSoC Jetson. Aceleradores convencionais como NVIDIA K80, alcançaram desempenho de até* 3750 *GFLOPS e eficiência energética de* 25 *GFLOPS/W, enquanto que, o acelerador embarcado de baixo consumo TK1 obteve desempenho de apenas* 301 *GFLOPS e eficiência energética superior, equivalente a* 26, 2 *GFLOPS/W.*

1. Introdução

O consumo energético é atualmente um dos maiores problemas para a construção de supercomputadores com desempenho *exascale*. O limite de consumo de 20 MW, definido pelo DARPA[1] tem motivado pesquisas em que empregam diferentes abordagens em seus projetos [Bergman et al. 2008]. Um exemplo é o Projeto Mont-Blanc, que tem como objetivo construir um supercomputador a partir de soluções com grande eficiência energética utilizando processadores ARM e aceleradores *(Graphics Processing Units)* (GPUs) [Montblanc Project 2015].

*Trabalho parcialmente apoiado por CNPq, CAPES, FAPERGS e FINEP. Pesquisa realizada no contexto do Laboratório Internacional Associado LICIA e projetos HPC-GA e HPC4E. Esta pesquisa tem sido parcialmente financiada pela bolsa da CAPES sob processo número 3471-13-6.

[1]Defense Advanced Research Projects Agency

Segundo [Schäppi et al. 2009], aproximadamente 50% do consumo total de um supercomputador é destinado a infraestrutura, como por exemplo refrigeração e iluminação. Do restante, 10% é destinado à interconexão e sistema de armazenamento, restando aproximadamente 40% para os nós de processamento. Nesse sentido, melhorias na eficiência energética tem sido buscadas nos nós de processamento, o que, indiretamente também reduz os custos de refrigeração. Contudo, para superar a limitação do DARPA, faz-se necessário alcançar uma eficiência energética de 50 GFLOPS/W, valor este que representa cerca de dez vezes maior do que a atual eficiência energética dos supercomputadores presentes na lista do Green500 [Feng and Lin 2010].

Muitas pesquisas têm sido desenvolvidas em busca de alternativas para melhorar a eficiência energética. Dentre elas, destacam-se aquelas que têm utilizado em sua composição uma arquitetura heterogênea agrupando CPUs, GPUs e processadores ARM presentes em *Multiprocessor System-on-Chip* (MPSoCs). Inicialmente as GPUs eram unidades com uma função fixa, construída sobre um pipeline gráfico que se sobressaía apenas em processamento de gráficos em três dimensões. Com a incorporação de diversas melhorias em nível de hardware, os modelos atuais passaram a dispor de unidades de processamento com grande capacidade aritmética e programável, não mais restrito à operações gráficas, surgindo assim o conceito de *General Purposes Computation on Graphics Processing Unity* (GPGPU) [Huang et al. 2009, Zanotto et al. 2012].

A partir destas evoluções, aliado ao grande desempenho e baixo consumo energético, aceleradores gráficos passam a ser vistos como uma alternativa para projetos de processamento de alto desempenho. Nesse contexto, este trabalho tem como objetivo analisar o desempenho e a eficiência energética de aceleradores embarcados de baixo consumo para implementação de sistemas de HPC. Para tanto testes foram realizados em 5 aceleradores GPUs convencionais de arquitetura Kepler da fabricante NVIDIA e um acelerador GPU de baixo consumo embarcado na placa MPSoC Jetson TK1.

O restante deste trabalho está assim organizado. A Seção 2 descreve as arquiteturas dos aceleradores GPU. Na Seção 3 é apresentada a metodologia de análise utilizada. Na Seção 4 são apresentados os resultados obtidos com a realização dos testes. Na sequência, na Seção 5 são destacados os trabalhos relacionados. Por fim, na Seção 6 são descritas nossas contribuições e propostas de trabalhos futuros.

2. Aceleradores GPGPU

Os aceleradores GPU, que até poucos anos eram utilizados exclusivamente para processamento gráfico, hoje caracterizam-se como uma das principais alternativas para a computação de alto desempenho. Isto justificado pelo seu grande poder de processamento e por sua grande eficiência energética.

Inicialmente as GPUs possuíam uma funcionalidade fixa, no entanto, devido uma série de inovações que foram incorporadas ao projeto de hardware, os modelos atuais possuem uma alta programabilidade, além de apresentarem elevado desempenho. Aceleradores hoje são utilizados em sistemas de HPC fazendo frente aos processadores de arquitetura x86 além de estarem presentes nos atuais projetos que almejam desempenho exaflop com um consumo de energia aceitável.

Dentre as características dos atuais aceleradores destaca-se alta capacidade de processamento massivo paralelo, principalmente em cálculos que exigem um grande volume

de dados. Os modelos da fabricante NVIDIA são construídas utilizando SPs *(Streaming Processors)*, que permite o processamento paralelo de uma série de operações em diversos fluxos de dados com altos níveis de eficiência e desempenho. Estes SPs são organizados em SMs *(Streaming Multiprocessors)*, que possuem memórias *caches*, escalonadores, registradores e outras unidades especializadas [NVIDIA 2015, Zanotto et al. 2012].

Os modelo de aceleradores GPUs da NVIDIA são classificados de acordo com a sua arquitetura de fabricação. De acordo com cada arquitetura, os principais detalhes de configuração que refletem no desempenho e no consumo de energia são:

- *Tesla* é baseada em um vetor de processadores escalonáveis, em 2007 a NVIDIA lançou a arquitetura Tesla. Estes aceleradores possuíam 128 SPs, organizados em 16 SMs, divididos em 8 unidades de processamento independentes, denominados de TPCs (Texture/Processor clusters). Nesta arquitetura os SMs são formados por 8 SPs, 2 SFUs *(Special Functions Unit)*, memória cache de instruções e memória de constantes apenas para leitura [Lindholm et al. 2008].
- *Fermi* é implementada em uma nova geração de SMs com 32 SPs que possuem pipeline completo de operações aritméticas e de ponto flutuante. Cada SM possui 16 unidades de carregamento e armazenamento, permitindo que os endereços de origem e destino possam ser calculados em paralelo por 16 *threads*. Os SMs também contam com 4 SFUs que são responsáveis por executar instruções como seno, cosseno e raiz quadrada [Nvidia 2009].
- *Kepler* é a arquitetura mais recente da NVIDIA. Nesta arquitetura os SMs passaram a ser denominados de SMX *Streaming Multiprocessor* e os SPs de CUDA Cores, onde cada SMX possui 192 CUDA Cores e 32 SFUs [NVIDIA 2014a]. Nesta arquitetura foi implantado o paralelismo dinâmico que permite a um *kernel* já em execução realizar a execução de outro *kernel*, diminuindo desta forma a dependência da CPU [NVIDIA 2014a].

3. Proposta de Análise dos Aceleradores GPU

Este trabalho tem como objetivo analisar o desempenho e a eficiência energética de aceleradores GPU visando a construção de novos sistemas para computação de alto desempenho. Para tanto serão analisados o desempenho, a velocidade de barramento e memória e a eficiência energética de dois tipos de aceleradores.

Nesta seção são descritas as especificações do ambiente de testes utilizado. A Subseção 3.1 apresenta as GPUs utilizadas e a Subseção 3.2 destaca as configurações do *benchmark* utilizado.

3.1. Ambiente de Testes

O ambiente de execução é composto por dois grupos de aceleradores. O primeiro grupo é formado por 5 GPUs convencionais da arquitetura NVIDIA Kepler. Dentre os quais foram selecionados os modelos K10, K20, K20X, K40 e K80. No segundo grupo destaca-se os novos aceleradores de categoria embarcados com baixo consumo. Para tanto foi selecionado a GPU presente no MPSoC Jetson TK1.

3.1.1. Aceleradores Convencionais

Este grupo de aceleradores está presente nos principais sistemas que lideram a lista Top500 [Dongarra et al. 2015]. Dentre os modelos, foram selecionados os principais aceleradores da arquitetura Kepler voltados para HPC. Para a realização dos testes com este grupo de aceleradores foram utilizados os equipamentos disponibilizados nos laboratórios do *NVIDIA Technology Center*.

- K10 é um acelerador construído com a primeira versão da arquitetura Kepler, a GK104. Este acelerador conta com um total de 8 GB de memória GDDR5 com frequência de *clock* de $2,5$ GHz e interface de 256 bits. A K10 conta com 1536 CUDA Cores distribuídos em 8 SMX em dois chips, totalizando 16 SMX e 3072 CUDA Cores de 745 MHz. A interface de conexão usada é PCI-E 3.0 x16 e a sua demanda de potência é de $117,5$ W por chip [NVIDIA 2015].
- K20 é produzido com o chip GK110. Possui 5 GB de memória GDDR5 com *clock* de $2,6$ GHz e interface de 30 bits. Essa GPU conta com 2496 CUDA Cores de 706 MHz distribuídos em 13 SMX, com uma interface de conexão é PCI-E 2.0 x16. A demanda de potência deste modelo é 225 W [NVIDIA 2015].
- K20X semelhante ao modelo anterior, este acelerador também foi implementado com o chip GK110. Possui 6 GB de memória GDDR5 com *clock* de $2,6$ GHz e interface de 384 bits. Conta com um SMX a mais que a K20, totalizando 14 SMX e 2688 CUDA *Cores* de 732 MHz. Utiliza uma interface de conexão PCI-E 2.0 x16. A demanda de potência deste modelo é 235 W [NVIDIA 2015].
- K40 é produzido com o chip GK110B, que possibilita a configuração da frequência de *clock*. Possui um total de 12 GB de memória GDDR5 de $3,0$ GHz e interface de 384 bits. Com 15 SMX totaliza 2880 CUDA Cores de 745 MHz que podem ser configurados para até 875 MHz. Neste modelo a interface de conexão utilizada é PCI-E 3.0 x16. A demanda de potência é de 235 W [NVIDIA 2015].
- K80 é a última geração de aceleradores produzido pela NVIDIA e emprega os chips GK210. Semelhante a K40, possui frequência configurável de 560 à 875 MHz em seus 2496 CUDA Cores presentes em cada um dos dois chips. Possui 24 GB de memória GDDR5 com frequência de $2,5$ GHz e interface de 384 bits. Adota uma interface de conexão PCI-E 3.0 x16. Em função da quantidade de CUDA Cores presentes, este modelo apresenta uma demanda de potência de 150 W por chip [NVIDIA 2015].

3.1.2. Aceleradores Embarcados de Baixo Consumo

Este grupo de aceleradores está sendo introduzido pela NVIDIA almejando atender projetos de HPC com boa eficiência energética. Uma restrição neste grupo é que ainda existem poucos modelos disponíveis para comercialização. Para realização dos testes, foi selecionado o acelerador presente na MPSoC Jetson TK1.

- TK1 é implementada na MPSoC Jetson TK1. Esta MPSoC possui um processador ARM Cortex-A15 de quatro Cores com *clock* de $2,32$ GHz e um acelerador GPU de chip GK20a com 192 CUDA Cores com *clock* de 852 MHz. Diferentemente dos aceleradores convencionais, esta GPU não possui memória própria, ela

compartilha com o processador os 2 GB de memória DDR3L de 933 MHz e uma interface de 64 bits [NVIDIA 2014b].

Na Tabela 1 são relacionadas as principais características destes equipamentos selecionados, denominados neste trabalho de TK1 (acelerador embarcado de baixo consumo) e K?? (aceleradores convencionais).

Tabela 1. Configuração dos equipamentos utilizados nos testes

	TK1	K10	K20	K20X	K40	K80
Modelo Chip	GK20a	GK104	GK110	GK110	GK110B	GK210
#CUDA Cores	192	1536	2496	2688	2880	2496
SMX	1	8	13	14	15	13
Frequência de Clock (MHz)	852	745	706	732	745	560
Memória GDDR5 (GB)		8	5	6	12	24
Memória DDR3L (GB)	2					
Frequência Clock de Memória (GHz)	0,93	2,5	2,6	2,6	3,0	2,5
Interface de Memória (bits)	64	256	320	384	384	384
Interface PCI-E x16	-	3.0	2.0	2.0	3.0	3.0
Potência (W)	11,5	117,5	225	235	235	150

No servidor da NVIDIA foi instalado o sistema operacional GNU/Linux CentOS release 6.6 com kernel versão 2.6.32 nos aceleradores convencionais e o sistema operacional Debian release 3.3, com kernel versão 3.4.106 para o acelerador de baixo consumo. O compilador utilizado em ambos os equipamentos foi o GCC versão 4.8.4 e CUDA versão 7.0.

Em uma primeira análise, pode-se destacar a grande diferença de potência entre os aceleradores, esta que varia de $11,5$ W até 235 W. Contudo, observa-se que neste trabalho foi utilizado somente a potência do acelerador e não a potência de todo o sistema.

3.2. Metodologia de Mensuração

Para a realização dos testes foi utilizado apenas um equipamento de cada modelo. Destaca-se que os aceleradores de modelo K10 e K80 possuem dois chips integrados na mesma GPU. Assim, Para uma comparação mais precisa com os demais aceleradores foi utilizado apenas um dos chips para que os testes fossem exatamente iguais em todos os aceleradores. Desta forma, a quantidade de cores, demanda de potência e resultados apresentados são apenas de um chip utilizado.

Para o cálculo da eficiência energética utilizou-se a demanda de potência especificada pela fabricante para cada um dos modelos testados. Os resultados representam uma média de dez execuções, sendo que a configuração de tamanho dos problemas utilizada nos testes foi sempre igual a quatro, configuração esta indicada para GPUs com grande quantidade de memória.

O *benchmark* utilizado para analisar o desempenho dos aceleradores foi o *Scalable Heterogeneous Computing Benchmark Suite* (SHOC) versão 1.1.4. Este *benchmark* foi escolhido por possuir uma grande abrangência de testes, além de permitir a mensuração de desempenho, disponibiliza uma série de testes que avaliam o desempenho, velocidade de barramento e memória, estabilidade de sistemas e ser compatível com os aceleradores escolhidos [Danalis 2010]. Outra característica importante deste *benchmark*, é a sua divisão em três níveis, o que possibilita uma análise com kernels artificiais e com aplicações reais. Os níveis de testes estão assim organizados:

- **Nível-0**: agrupa testes que mensuram características de baixo nível do hardware usando kernels artificiais. Dentre eles, foram selecionados os seguintes testes:
  - **MaxFlops** que mensura o desempenho máximo atingido com operações de ponto flutuante com valores de precisão simples e dupla;
  - **BusSpeed** que realiza a medição da banda de transferência de dados entre processador e o acelerador, por meio de repetidas operações alterando o tamanho dos dados, de 1 KB a 64 MB; e
  - **DeviceMemory** que mede a velocidade de acesso à memória de diferentes níveis da hierarquia e diferentes padrões de acesso.
- **Nível-1**: implementa testes que utilizam algoritmos paralelos tradicionais para análise do desempenho. Dentre eles, foram selecionados os seguintes testes:
  - **FFT** (*Fast Fourier Transform*) que mede o desempenho dos aceleradores utilizando valores de precisão simples e dupla no cálculo da FFT. Esse algoritmo é de grande importância para muitas aplicações como processamento digital de sinais para a resolução de equações diferenciais parciais ou algoritmos para multiplicação de grandes inteiros. Para um melhor desempenho na GPU o tamanho do problema é fixado em 512 elementos complexos; e
  - **GEMM** (*General Matrix Multiply*) que realiza multiplicação de matrizes com precisão simples e dupla. O tamanho do problema utilizado nos testes foi de 16 KB.
- **Nível 2**: agrupa testes com kernels de aplicações reais. Neste nível foi selecionado a seguinte aplicação:
  - **S3D** é uma aplicação real utilizada pelo DoE[2] para modelar a combustão de biocombustíveis. Resolve equações de Navier-Stokes para um domínio 3D regulares.

4. Resultados

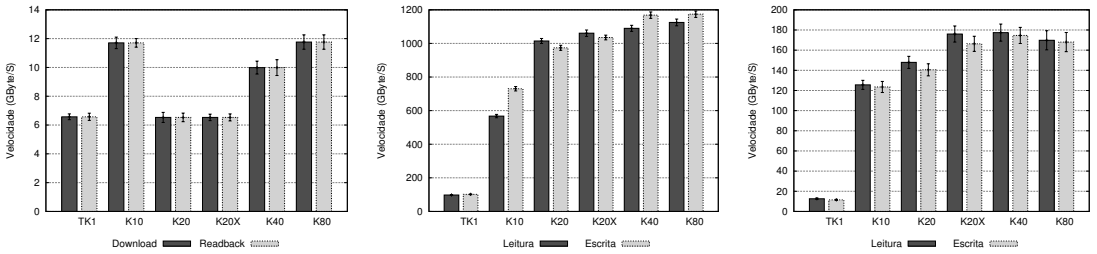
Nesta seção serão apresentados os resultados obtidos com a execução do *benchmark* nos equipamentos de teste. Na Seção 4.1 serão descritos os resultados relativos às características de baixo nível de hardware com os kernels artificiais. Na Seção 4.2 serão apresentados os resultados obtidos com os algoritmos paralelos básicos do *benchmark*, por fim, na Seção 4.3 será analisado o consumo energético utilizando uma aplicação real.

4.1. Resultados testes de hardware com Kernels Artificiais

Nesta primeira Seção serão apresentados os resultados de velocidade de barramento, acesso à memória, seguidos de desempenho e eficiência energética.

Na Figura 1(a) são destacados os resultados do teste de velocidade de barramento mensurado com o kernel artificial **BusSpeed**. Neste teste é possível analisar a velocidade de comunicação entre o processador e o acelerador GPU. Pode-se observar que os aceleradores K10, K40 e K80 alcançaram velocidades de aproximadamente 12 GBytes/s tanto nos testes de *Download* quanto para *Readback*, justificado pela presença de conexão PCI-E 3.0. Nos demais aceleradores que possuem PCI-E 2.0 a velocidade foi de aproximadamente 6,5 GBytes/s. O acelerador TK1 não possui conexão PCI-E, mas é um chip

[2]U. S. Department of Energy



(a) Velocidade de Barramento (BusSpeed)

(b) Velocidade de acesso à Memória Local (DeviceMemory)

(c) Velocidade de acesso à Memória Global (DeviceMemory)

Figura 1. Velocidade de Barramento e Memória nos testes com kernels artificiais

implementado direto na placa. Desta forma, as velocidades obtidas foram semelhantes às alcançadas com as conexões PCI-E 2.0.

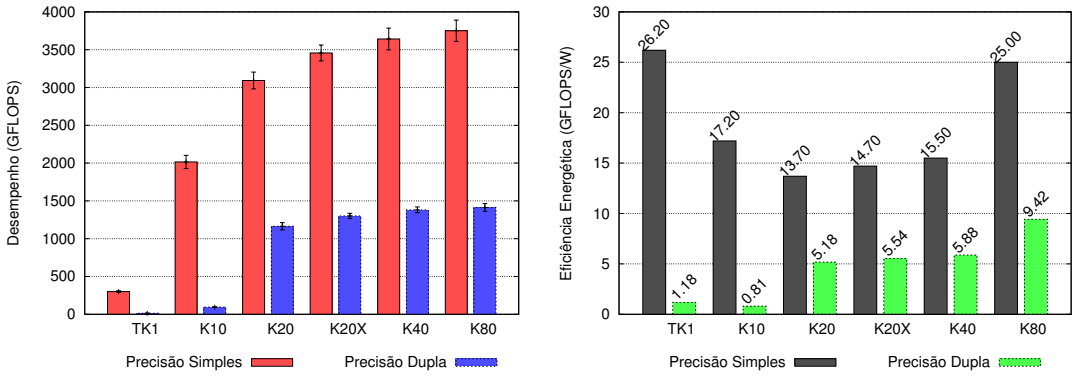
As Figuras 1(b) e 1(c) apresentam os resultados dos testes de velocidade de acesso à memória local e global mensurado com o kernel artificial **DeviceMemory**. Observa-se uma diferença de velocidade entre os aceleradores convencionais. Para acesso à memória local a velocidade varia de 567 Gbytes/s até 1174 Gbytes/s enquanto que para acesso à memória global varia de 123 Gbytes/s até 177 Gbytes/s. Essas diferenças de velocidade entre os aceleradores é justificada de acordo com a frequência de *clock* e interface de cada modelo, conforme apresentado na Tabela 1.

Observa-se que o acelerador TK1 alcançou velocidades de cerca de 5 vezes menor para acesso a memória local e 10 vezes menor para acesso a memória global se comparado com os demais aceleradores. Esta diferença é justificada pela ausência de um memória própria, uma vez que este acelerador faz uso da memória DDR3L compartilhada com o processador e possui uma frequência de *clock* e interface bem abaixo das demais.

Os resultados de desempenho máximo alcançado na execução com o kernel artificial **MaxFlops** são apresentados na Figura 2(a). Percebe-se que o desempenho mensurado com operações de ponto flutuante de precisão simples varia entre os aceleradores, principalmente em relação ao acelerador TK1. Esta variação é percebida de acordo com a quantidade de CUDA cores presentes em cada acelerador e a sua respectiva frequência de *clock*. Diferentemente, o acelerador K80 obteve um resultado superior as demais mesmo possuindo um menor número de cores e uma frequência menor se comparada com o acelerador K40. Isto é justificado pela diferença de tecnologia empregada na construção desta última versão de *chipset*, o GK210, que possui o dobro de registradores e memória compartilhada por SMX [NVIDIA 2014a].

Percebe-se também uma diferença de desempenho entre os resultados de precisão simples e dupla no acelerador K10. Esta GPU é uma das primeiras lançadas da arquitetura Kepler e possui um *chipset* modelo GK104. Nas demais GPUs foram adicionadas 64 unidades de precisão dupla para melhor desempenho com operações de precisão dupla.

Na Figura 2(b) é apresentada a eficiência energética de cada acelerador a partir do desempenho máximo alcançado na execução do kernel **MaxFlops**. Para operações com precisão simples, a TK1 obteve uma eficiência energética $26,2$ GFLOPS/W, acima do acelerador K80 que obteve uma eficiência de 25 GFLOPS/W. Esta diferença é carac-



(a) Desempenho (MaxFlops) (b) Eficiência energética (MaxFlops)

Figura 2. Desempenho e Eficiência Energética nos testes com kernels artificiais

terizada pela diferença de desempenho e demanda de potência das duas GPUs. A TK1 obteve um desempenho $12,45$ vezes menor (301 x 3750 GFLOPS), enquanto que a sua potência é $13,04$ vezes menor ($11,5$ x 150 W).

Para operações precisão dupla a GPU TK1 obteve uma eficiência energética de $7,98$ vezes menor ($1,18$ x $9,42$ GFLOS/W). Esta diferença se justifica pela grande diferença de desempenho obtido pela GPU K80.

4.2. Resultados testes com Algoritmos Paralelos Tradicionais

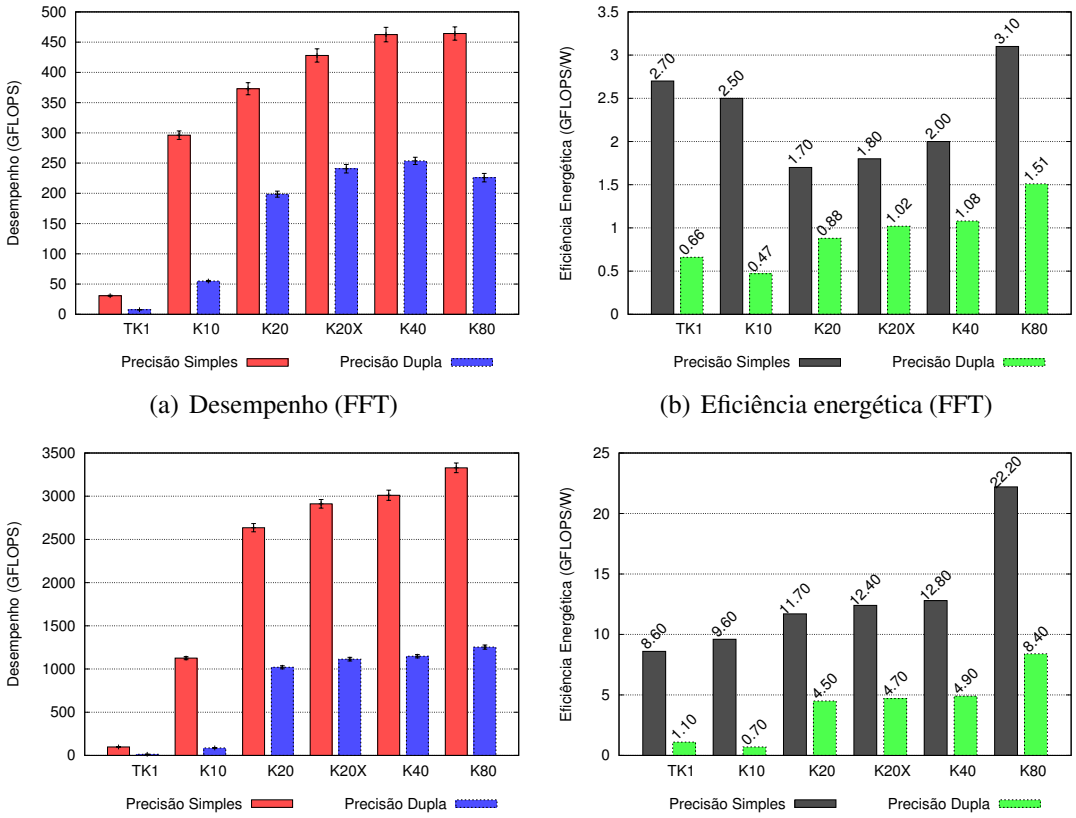
O objetivo desta seção é comparar o desempenho e a eficiência energética mensurados com algoritmos paralelos tradicionais nos aceleradores selecionados. Para tanto, serão analisados os resultados obtidos com os algoritmos **FFT** e **GEMM** do Nível-1 do *benchmark* SHOC.

Na Figura 3 são apresentados os desempenhos mensurados na execução dos testes. Percebe-se que os resultados de desempenho dos testes com FFT e GEMM apresentam variações semelhantes ao mensurados na execução do MaxFlops do Nível-0. Variações estas que acompanham o crescimento no número de CUDA cores presentes em cada acelerador, ou relativo à variação de frequência de *clock*, tanto dos cores quanto de memória.

Com base no desempenho alcançado com os algoritmos FFT e GEMM, foi calculada a eficiência energética de cada acelerador. Nas Figuras 3(b) e 3(d) são apresentados os resultados destacando a utilização de precisão simples e precisão dupla.

Semelhante ao observado com o kernel MaxFlops, percebe-se na Figura 3(b) que nos testes realizados com FFT, o acelerador TK1 obteve uma eficiência energética semelhante ao acelerador K80. Executando FFT com precisão simples no acelerador TK1 a eficiência energética alcançada foi de $2,70$ GFLOPS/W enquanto que na K80 foi de $3,10$ GFLOPS/W. Por outro lado, para operações de precisão dupla a GPU TK1 obteve uma eficiência energética de $2,28$ vezes menor ($0,66$ x $1,51$ GFLOPS/W).

Na Figura 3(d) podemos analisar que executando o algoritmo GEMM, o acelerador K80 obteve uma eficiência energética de $22,2$ GFLOPS/W para operações de precisão simples e $8,40$ GFLOPS/W para operações de precisão dupla. Tal eficiência é $2,5$ e $7,6$ superior às eficiências alcançadas com o acelerador TK1.



(a) Desempenho (FFT)

(b) Eficiência energética (FFT)

(c) Desempenho (GEMM)

(d) Eficiência energética (GEMM)

Figura 3. Desempenho e eficiência energética com algoritmos tradicionais

4.3. Resultados dos testes com Aplicação Real

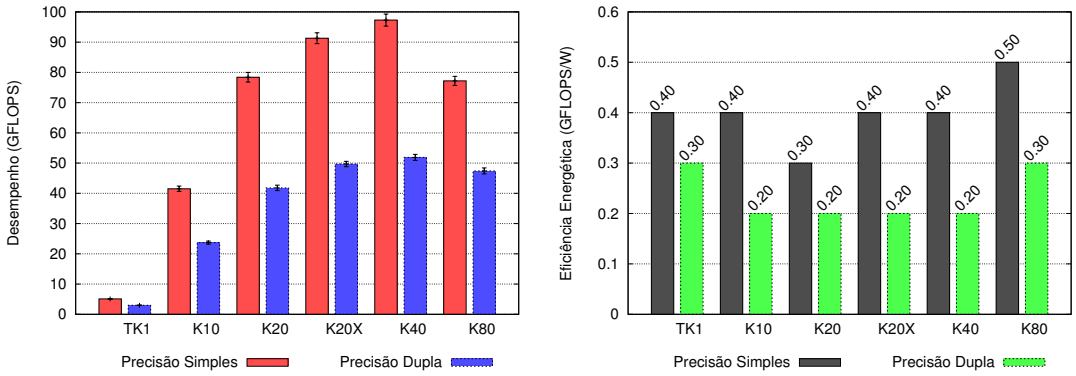
O objetivo desta seção é analisar o desempenho e a eficiência dos aceleradores com uma aplicação real. Para este fim, foi selecionada a aplicação **S3D**.

Diferentemente dos demais testes realizados com kernels e algoritmos convencionais (Níveis 0 e 1), na execução de uma aplicação real, o acelerador K40 que possui 2880 CUDA cores, obteve o melhor desempenho como pode ser visto na Figura 4(a), obtendo $97,3$ GFLOPS para operações de precisão simples e $51,9$ GFLOPS para precisão dupla. Por outro lado, o acelerador K80 obteve uma eficiência energética maior, ($0,51$ GFLOPS/W e $0,32$ GFLOPS/W) devido a sua potência 150 W, que é relativamente inferior aos 235 W da potência da K40.

Se comparado os desempenhos mensurados no acelerador TK1 e K40, a segunda foi 19 vezes superior para operações de precisão simples e $17,5$ vezes superior para precisão dupla. Entretanto, analisando somente a eficiência energética dos acelerados com esta aplicação real, os aceleradores TK1 e K80, alcançaram os melhores resultados para precisão simples, $0,44$ GFLOPS/W e $0,51$ GFLOPS/W respectivamente como visto na Figura 4(b).

5. Trabalhos Relacionados

Diversos trabalhos têm avaliado desempenho e consumo de energia de GPUs. Huang *et al.* [Huang et al. 2009], comparam o desempenho entre CPUs e GPUs utilizando algorit-



(a) Desempenho (S3D) (b) Eficiência energética (S3D)

Figura 4. Desempenho e eficiência energética com a aplicação real

mos de multiplicação de matrizes. Os autores concluem que sistemas heterogêneos com GPUs obtêm desempenhos até 46 vezes superiores com consumo energia de até 17 vezes mais baixo. Semelhante, Jiao *et al.* [Jiao et al.] utiliza um conjunto de aplicações para analisar o desempenho e eficiência energética de CPUs e GPUs com frequência de *clock* alterados dinamicamente. Padoin *et al.* [Padoin et al. 2013] investigam a eficiência energética de um sistema heterogêneo (CPU + GPU) usando uma aplicação científica.

Em outra linha, Liu *et al.* [Liu et al. 2011] propõem um método de mapeamento de tarefas para clusters heterogêneos com aceleradores com objetivo de reduzir o consumo energético. Utilizando o método em um cluster composto de duas GPUs por CPU foi reduzido em até 50%.

Gulo *et al.* [Gulo 2012] apresentam uma abordagem de paralelização utilizando aceleradores GPGPUs com a plataforma CUDA semelhante ao implementado por Jin e Yang em [Jin and Yang 2011]. Os resultados dos testes indicam speedup de até 15 vezes no tempo de processamento se comparado com o algoritmo sequencial. Bellori *et al.* usam GPUs e a plataforma CUDA na solução de um computação científica utilizado o problema de Difusão de Calor. Os resultados obtidos com aceleradores GPU apresentam redução no tempo de execução de até 90% se comparados com CPUs [Bellorini and Galante 2009].

Diferente dos trabalhos aqui citados, que apresentam comparativos de desempenho e eficiência energética entre as GPUs convencionais e CPUs, a nossa pesquisa busca analisar a viabilidade de utilização de aceleradores embarcados de baixo consumo na computação de alto desempenho.

6. Conclusão e Trabalhos Futuros

Dada as restrições de consumo estabelecidas para construção dos futuros sistemas exascale, novas abordagens têm motivado pesquisas em busca de diferentes construções de sistemas. Nesse sentido, aceleradores têm sido uma das apostas para o aumento de desempenho e eficiência energética dos futuros sistemas de HPC.

Nesse sentido, visando a utilização de aceleradores embarcados de baixo consumo na concepção de novos sistemas de computação de alto desempenho, esse artigo apresentou uma análise de desempenho e eficiência energética entre dois tipos de aceleradores GPUs da NVIDIA.

Utilizando diferentes kernels artificiais, algoritmos convencionais e uma aplicação real foram comparados o desempenho, a velocidade de barramento e acesso à memória de aceleradores convencionais e aceleradores embarcados de baixo consumo produzidos com arquitetura Kepler.

Com relação ao desempenho, observou-se que o acelerador embarcado TK1 apresenta um desempenho consideravelmente menor se comparada com as GPUs convencionais. Os testes demonstraram um poder de processamento de até 301 GFLOPS, o que representa um desempenho 12 vezes menor ao alcançado pelos aceleradores convencionais.

Observando os resultados alcançados surgem duas alternativas para construção de sistemas de HPC. Utilizando aceleradores convencionais tem-se desempenhos superiores de até 12 vezes, porém estes possuem uma demanda de potência 20 vezes superior. Por outro lado, com o emprego de aceleradores embarcados tem-se uma demanda de potência reduzida, com uma eficiência energética um pouco maior, pois o acelerador convencional K80 alcançou eficiência energética de 25 GFLOPS/W, enquanto que o acelerador de baixo consumo TK1 obteve uma eficiência energética de $26,2$ GFLOPS/W.

A partir dos resultados alcançados, torna-se possível uma melhor escolha de acelerador na implementação de novos sistemas de HPC. Como trabalhos futuros pretende-se dar continuidade neste trabalho investigando novos aceleradores de baixo consumo e aplicação em outras aplicações reais. A primeira, motivada pela demanda de novos modelos de aceleradores com maior eficiência, que faz com que sejam utilizados cada vez mais aceleradores para implementação de novos servidores de HPC. Também, uma nova implementação será desenvolvida utilizando um cluster de MPSoC Jetson. Com esta abordagem torna-se possível a análise de escalabilidade destas placas com este tipo de organização.

Outros pontos que podem ser analisados em trabalhos futuros é que ao empregar aceleradores embarcados pode-se reduzir consideravelmente o custo com refrigeração do sistema e também o seu custo substancialmente reduzido. A NVIDIA Jetson TK1 custa em média 20 vezes menos que um acelerador convencional e já possui processador, memória e GPU.

Referências

Bellorini, E. A. and Galante, G. (2009). Resolução do problema de difusão de calor usando gpus. In *Escola Regional de Alto Desempenho*, volume 9, pages 245–248. ERAD.

Bergman, K., Borkar, S., Campbell, D., Carlson, W., Dally, W., Denneau, M., Franzon, P., Harrod, W., Hill, K., Hiller, J., et al. (2008). Exascale computing study: Technology challenges in achieving exascale systems. *Defense Advanced Research Projects Agency Information Processing Techniques Office (DARPA IPTO), Tech. Rep*, 15.

Danalis, A. e. a. (2010). The scalable heterogeneous computing (SHOC) benchmark suite. In *Proceedings of the 3rd Workshop on General-Purpose Computation on Graphics Processing Units*, pages 63–74. ACM.

Dongarra, J., Meuer, H., and Strohmaier, E. (2015). *TOP500 Supercomputer Sites*.

Feng, W. and Lin, H. (2010). The Green500 List: Year Two. In *International Parallel and Distributed Processing Workshops (IPDPSW)*, Atlanta, Georgia, USA. IEEE.

Gulo, C. A. S. J. (2012). Técnicas de paralelização em gpgpu aplicadas em algoritmo para remoção de ruído multiplicativo. Dissertação (mestrado) - Universidade Estadual Paulista, Instituto de Biociências, Letras e Ciências Exatas. http://hdl.handle.net/11449/89336.

Huang, S., Xiao, S., and Feng, W.-c. (2009). On the energy efficiency of graphics processing units for scientific computing. In *Parallel & Distributed Processing, 2009. IPDPS 2009. IEEE International Symposium on*, pages 1–8. IEEE.

Jiao, Y., Lin, H., Balaji, P., and Feng, W. Power and performance characterization of computational kernels on the gpu. In *Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l Conference on & Int'l Conference on Cyber, Physical and Social Computing (CPSCom)*, pages 221–228. IEEE.

Jin, Z. and Yang, X. (2011). A variational model to remove the multiplicative noise in ultrasound images. *Journal of Mathematical Imaging and Vision*, 39(1):62–74.

Lindholm, E., Nickolls, J., Oberman, S. F., and Montrym, J. (2008). Tesla: A unified graphics and computing architecture. *IEEE Micro*, pages 39–55.

Liu, W., Du, Z., Xiao, Y., Bader, D., and Xu, C. (2011). A waterfall model to achieve energy efficient tasks mapping for large scale gpu clusters. In *Parallel and Distributed Processing Workshops and Phd Forum (IPDPSW), 2011 IEEE International Symposium on*, pages 82–92. IEEE.

Montblanc Project (2015). European approach towards energy efficient high performance. http://montblanc-project.eu/.

Nvidia (2009). NVIDIA's Next Generation CUDA Compute Architecture: FERMI. http://www.nvidia.com/content/pdf/fermi/whitepaper.pdf.

NVIDIA (2014a). NVIDIA's Next Generation CUDA Compute Architecture: Kepler GK110/210. http://international.download.nvidia.com/pdf/kepler/NVIDIA-Kepler-GK110-GK210-Architecture-Whitepaper.pdf.

NVIDIA (2014b). Whitepaper: NVIDIA Tegra K1. http://www.nvidia.com/content/PDF/tegra_white_papers/tegra-K1-whitepaper.pdf.

NVIDIA (2015). Publicações sobre o produto. http://www.nvidia.com.br/object/tesla_product_literature_br.html.

Padoin, E. L., Pilla, L. L., Boito, F. Z., Kassick, R. V., Velho, P., and Navaux, P. O. A. (2013). Evaluating application performance and energy consumption on hybrid CPU+GPU architecture. *Cluster Computing*, 16(3):511–525. 10.1007/s10586-012-0219-6.

Schäppi, B., Przywara, B., Bellosa, F., Bogner, T., Weeren, S., Harrison, R., and Anglade, A. (2009). Energy efficient servers in europe. http://ec.europa.eu/energy/intelligent/projects/sites/iee-projects/files/projects/documents/e-server_e_server_final_publishable_report_en.pdf.

Zanotto, L., Ferreira, A., and Matsumoto, M. (2012). Arquitetura e Programação de GPU Nvidia. pages 1–7.