On Power-Performance Characterization of Concurrent Throughput Kernels

Nilanjan Goswami, Yuhai Li, Amer Qouneh, Chao Li, and Tao Li
Intelligent Design of Efficient Architecture Lab (IDEAL)
University of Florida, Gainesville, Florida, USA
nil@ufl.edu, liyuhai.cn@gmail.com, aqouneh@ufl.edu, lichao@cs.sjtu.edu.cn, taoli@ece.ufl.edu

***Abstract*— Growing deployment of power and energy efficient throughput accelerators (GPU) in data centers pushes the envelope of power-performance co-optimization capabilities of GPUs. Realization of exascale computing using accelerators demands further improvements in power efficiency. With hardwired kernel concurrency enablement in accelerators, inter- and intra-workload simultaneous kernels computation predicts increased throughput at lower energy budget. To improve Performance-per-Watt metric of the architectures, systematic empirical study of real-world throughput workloads (with simultaneous kernel execution) is required. To this end, we propose a multi-kernel throughput workload generation framework that will facilitate aggressive energy and performance management of exascale data centers and will stimulate synergistic power-performance co-optimization of throughput architectures.**

Keywords-GPGPU, Power-Performance Analysis, workload characterization.

I. Introduction

To improved energy efficiency and better performance, the datacenters and supercomputers (Tianhe-1A, Nebulae, Tsubame) are increasingly adopting throughput computing architectures such as GPUs (Nvidia, AMD), dedicated accelerators (Intel MIC), IBM Cell processors. Parallel thread processing in throughput processors often shares hardware structures (shared memory, scheduling hardware, issue/decode unit, etc.) to compensate individual thread processing energy overhead. As a result, overall energy efficiency is enhanced. In those processors, energy efficiency, concurrent execution paradigm and performance improvement are intertwined. For example, increased concurrency and performance do not always map to improved energy and power efficiency. In this paper, we focus on kernel level concurrency that has a significant impact on performance and power. A thorough exploration of power-performance characteristics of concurrent throughput kernels is still lacking. There is a need to identify a representative mix of workloads, which will reduce overall energy footprint and retain throughput. To this end, we propose a flexible methodology to amalgam emerging throughput workloads. To design an energy efficient architecture, architects need to understand energy and power implications of kernel level concurrency. Hence, we delve into the systematic exploration of throughput workloads that unleashes power-performance co-characterization.

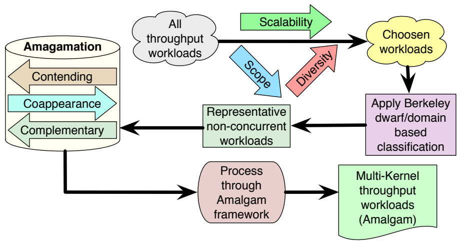
Amagamation
Contending
Coappearance
Complementary
All throughput workloads
Scalability
Scope
Diversity
Choosen workloads
Apply Berkeley dwarf/domain based classification
Representative non-concurrent workloads
Process through Amalgam framework
Multi-Kernel throughput workloads (Amalgam)

Fig 1. Methodology

II. Power Performance Co-Characterization

Figure 1 depicts the flow of operations for multi-kernel workload generation.

A. Throughput Benchmark Selection

We have used Berkeley Dwarves [1] based systematic approach for throughput workload selection. To choose representative workloads that cover the dwarves, we have investigated Nvidia GPU computing SDK, Rodinia, Parboil, and several third party benchmarks. Workload selection process scrutinized the application purview (data-centers, mobile, desktop, embedded) of the workload, characteristic diversity of the benchmark based on [2] and expansion (with growing load) capability of workloads in scalable emerging systems. Application scope ensures broader impact, characteristic diversity guarantees architecture exploration capability, and scalability captures workload augmentation capability with larger input/system.

TABLE I. Throughput Workload Synopsis

Bench (Acronym)	
Breadth First Search (BFS)	Computational Fluid Dyn. (CFD)
Sum of Abs. Diff. (SAD)	SparseMatrix DenseVector Mult. (SPMV)
LU Decomposition (LUD	Heart Wall (HW)
Matrix Mult. (MM)	Hybrid Sort (HY)
Black Scholes (BS)	Needleman-Wunsch (NW)
Binomial Options (BN)	N-Queen Solver (NQ)
Path Finder (PF)	Advanced Encryption Std (AES)
2D Convolution FFT (FFT)	Lava MD2 (LM)
Ray Trace (RAY)	

B. Performance-power Co-characterization Methodology

Selected benchmarks in Table I are characterized in terms of microarchitecture agnostic behaviors [2] and microarchitecture dependent power-performance characteristics. Microarchitecture agnostic metrics unleash the intrinsic characteristics using generic workload properties (dynamic instruction count, memory/branch/atomic/shared-memory instruction count etc.) and throughput workload specific properties (per-thread register usage, data transfer in between host and device, control flow divergence, memory access locality, thread-batch efficiency etc.). Contrarily, power-performance metrics expresses power and performance dissimilarities to help the co-characterization process. Precisely, power, energy, and temperature depict energy consumption aspect of the workloads; IPC indicates performance, communication overhead encapsulates performance degradation due to excessive host to device interaction and IPW/EDP captures co-optimization characteristics.

Next, workloads from Table I are executed on real Nvidia hardware such as Tesla M2050, Tesla K20X, and GTX470. Using Nvidia Nsight Eclipse profiler, we have collected all the metrics. We have performed two separate PCA and clustering (*hierarchical*, *kmeans*) analyses. Such analysis unleashes similarity and dissimilarity information across the benchmarks and assists in selecting representative kernels. Based on workload scattering in various PC domains, we confirm that microarchitecture independent characteristics based workload diversity is successfully retained. Across GPUs with different power efficiency, power behavior based clustering changes significantly.

To choose a set of representative multi-kernel throughput workloads, we assign a *relation score* to each throughput benchmark and create a workload database. The score is given to each benchmark in a cluster. There are multiple such clusters generated from the power and characteristics analysis. To avoid clustering artifact, we performed *hierarchical* and *kmeans* clustering simultaneously on all data. Since workload characteristics define the execution pattern, we assign greater weight to it. We have used cluster ensemble analysis on characteristic and power clusters for each GPU. The analysis provides the final set of clusters mentioned in Table II, and individual benchmark scores in the workload database.

TABLE II. THROUGHPUT WORKLOAD CLUSTERS

Kmeans	BN, CFD	NW, LM	AES, RAY, BFS	BS, HY	NQ, MM	SAD, PF	LUD, FFT	SPMV	HW
Hierarchical	BN, BS, HW	AES, SPMV	SAD, MM	BFS, NQ	LUD, FFT	NW, PF	LM, CFD	HY	RAY
Consensus	BN, HW	BS, HY	AES, SPMV	MM, NQ	BFS, RAY	FFT, LUD	NW, PF	LM, CFD	SAD

III. RESULT AND CONCLUSION

Figures 4 show hierarchical clustering (based on PCA) of all concurrent and sequential kernel workloads respectively. In most of the cases, concurrent kernels have large linkage distance with sequential kernels. Interestingly, often one or more kernels in clustered-workload show characteristic dominance within the benchmarks. For example, in MM_LUD, LUD and SAD_LUD_BN, dominant behavior of LUD keeps them in close proximity. Quantitative evidence suggests that sequential and concurrent kernels are truly dissimilar in nature. Intrinsic workload characteristics of sequential kernels dominate or subdue co-existing kernels in concurrent workload behavior. Figures 2, 3 and 5 show power-performance co-characterization of concurrent kernel workloads. PCA is performed based on first 3 PCs. In Tesla M2050, Tesla K20 and GTX470, the 3 PCs retain 95%, 93% and 94% cumulative variance respectively. Clustering trends in three generations of throughput architecture are distinct. As claimed by Nvidia, GTX470, M2050 and K20 GPUs show prominently disparate power-performance traits. Both being Fermi architectures, GTX470 and Tesla M2050 have different power efficiency due to power overhead of graphics capability of GTX470. Order of magnitude power efficiency improvement of K20 is clearly visible from M2050 and K20 clusters. Interestingly, concurrent kernels (SPMV_RAY, SPMV_AES) with a common sequential kernel (SPMV) do not cluster together in any dendrogram. Unlike, intrinsic workload characteristics, individual workload dominance or subduing trend is absent in power-performance behavior of concurrent kernels.

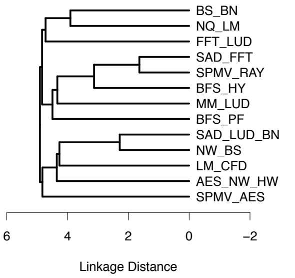

Fig 2. Dendrogram using power-performance characteristics Tesla K20

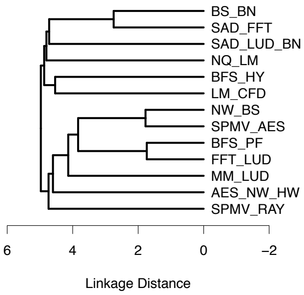

Fig 3. Dendrogram based on Power-Performance Characteristics GTX470

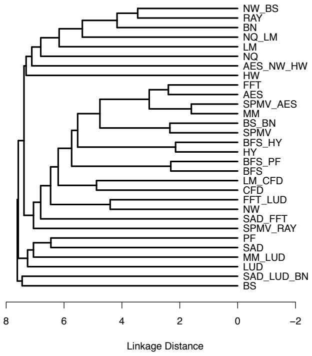

Fig 4. Dendrogram based on microarchitecture independent characteristics of all benchmarks

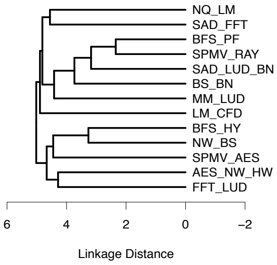

Fig 5. Dendrogram based on power-performance characteristics of Tesla M2050

In conclusion, we introduce a novel framework for multi-kernel throughput workload generation and perform a thorough study of the proposed workloads in terms of performance, power, energy, utilization and interactions between them. Using real Nvidia GPUs of different generation and by varying application scope, we show that power-profile and concurrency are highly correlated.

ACKNOWLEDGMENT

This work is supported in part by NSF grants. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not reflect the views of the NSF. Authors acknowledge UF HPC center for providing computational resources.

REFERENCES

[1] K. Asanovic, R. Bodik, B. C. Catanzaro, J. J. Gebis, P. Husbands, K. Keutzer, *et al.*, "The Landscape of Parallel Computing Research: A View from Berkeley," EECS Department, University of California, Berkeley UCB/EECS-2006-183, December 18 2006.

[2] N. Goswami, R. Shankar, M. Joshi, and T. Li, "Exploring GPGPU workloads: Characterization methodology, analysis and microarchitecture evaluation implications," in *Workload Characterization (IISWC), 2010 IEEE International Symposium on*, 2010, pp. 1-10.