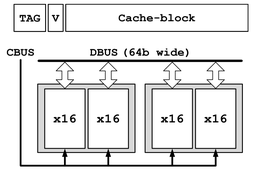
A Locality-Aware Memory Hierarchy for Energy-Efficient GPU Architectures

Minsoo Rhu Michael Sullivan Jingwen Leng Mattan Erez
The University of Texas at Austin

INTRODUCTION

Using GPUs for compute has become increasingly popular thanks to their high arithmetic throughput and peak memory bandwidth. Prior work assumes a *coarse-grained* (CG) memory hierarchy for GPUs, which is a perfect match for regular programs with high data locality, and provides high peak memory bandwidth while decreasing control overheads. Applications with irregular control and memory access patterns, however, suffer from inefficient caching, which leads to substantial waste in off-chip memory throughput under the baseline CG memory system.

We argue that GPU cache architecture and throughput-oriented design necessitates *fine-grained* (FG) data fetching capability across the memory hierarchy. By dynamically predicting optimal access granularity, our locality-aware memory hierarchy achieves the best of both CG and FG access granularity characteristics.



- **CBUS**: Control bus
- **DBUS**: Data bus
- **GDDR5 chips**: 2 chips/channel
- **Cache block size**: 128 Bytes

Fig. 1: Baseline CG-only memory system.

GPU CACHING EFFICIENCY

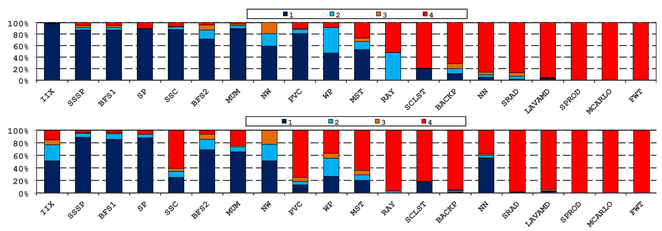


Fig. 2: Number of 32-Byte sectors referenced in L1 (top) L2 (bottom).

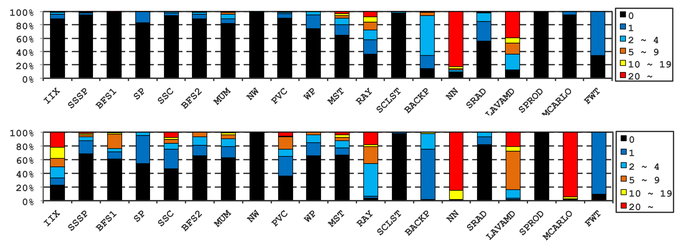


Fig. 3: Number of repeated accesses in L1 (top) L2 (bottom).

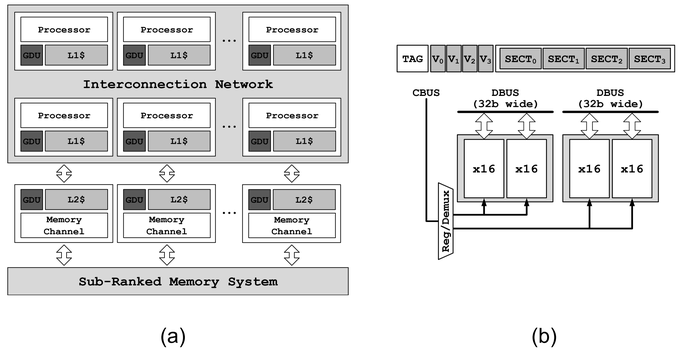
Analysis

- GPU shader core issues memory accesses in 32 -- 128 Bytes granularity
- CG-only memory system, which *always* fetches missed blocks in cache block size granularity, substantially wastes off-chip bandwidth.

LAMAR FOR GPUS

Our locality-aware memory hierarchy (LAMAR) is motivated by the dual observations that many GPU applications demonstrate highly dynamic and diverse levels of spatial locality, and that the low per-thread cache capacity limits the amount of temporal locality that can be exploited.

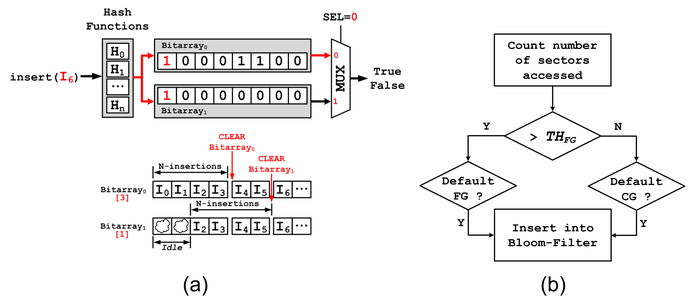
LAMAR enables *fine-grained* (FG) data fetching across the entire memory hierarchy, improving off-chip bandwidth utilization. FG accesses are provided by a sectored cache hierarchy and a sub-ranked memory system (Fig. 4). Within each cache level, the granularity-decision unit (GDU) determines whether to initiate each miss request in CG or in FG. By dynamically predicting access granularity, LAMAR achieves the best of both CG and FG access granularity characteristics.



(a) (b)

Fig. 4: (a) LAMAR overview. (b) FG-enabled memory hierarchy.

GRANULARITY PREDICTION



(a) (b)

Fig. 5: (a) Bi-modal granularity predictor, (b) Insertion algorithm.

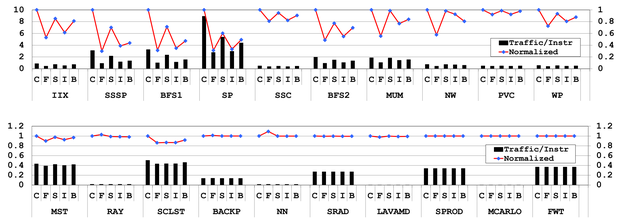
Dual-bitarray, bloom-filter based design

- Light-weight, ***temporally overlapped*** for history preservation
- Always maintain subset of insertion-history
- Balance size, false positive rate, and history depth
- ***Agree predictor**** inspired design

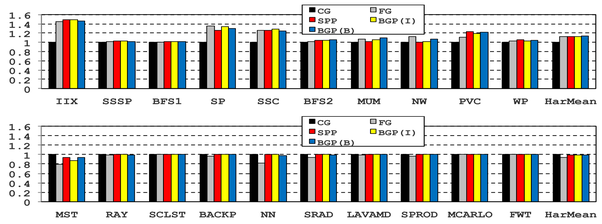
* Sprangle et al., "The agree predictor: A mechanism for reducing negative branch history interference", ISCA-1997

EVALUATION

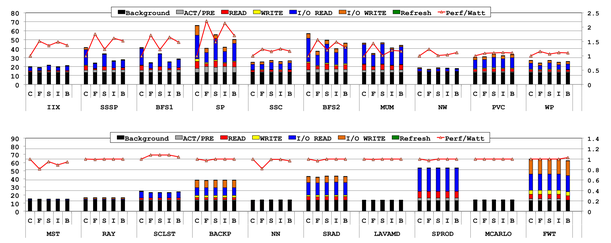
- Methodology
  - GPGPU-Sim + DrSim
    - GPGPU-Sim (http://gpgpu-sim.org)
    - DrSim (http://lph.ece.utexas.edu/public/Main/DrSim)
    - 15 SMs, 64KB on-chip SRAM (L1/scratchpad), 768KB shared L2
    - 8 memory channels (179.2 GB/sec), sub-ranked, FR-FCFS
    - Benchmarks: Rodinia, CUDA-SDK, LonestarGPU
- Evaluation Results
  - **Off-chip traffic**



  - **Performance**



  - **DRAM power consumption**



CONCLUSION

- LAMAR significantly improves energy-efficiency (*Perf/Watt*)
  - Memory hierarchy of GPUs necessitates fine-grained access granularity
    - Inherent data locality is rarely captured in GPUs
    - Minimizing useless overfetching (within cache block) improves bandwidth utilization
  - DRAM power consumption significantly reduced

Email: minsoo.rhu@utexas.edu