Cache-locality Based Adaptive Warp Scheduling for Neural Network Acceleration on GPGPUs

Weiming Hu[1], Yi Zhou[2], Ying Quan[2], Yuanfeng Wang[2], Xin Lou[1]

[1]*School of Information Science and Technology, ShanghaiTech University, Shanghai, China*

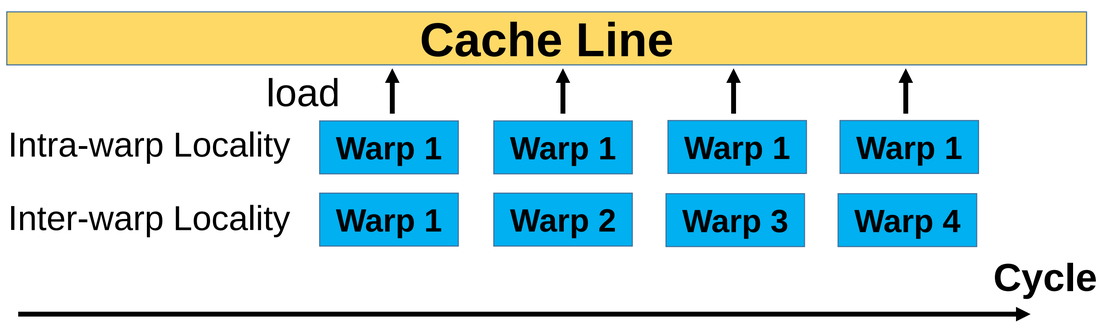
[2]*Glenfly Tech Co., Ltd., Shanghai, China*

{huwm1, louxin}@shanghaitech.edu.cn, {peterzhou, rainiequan, rogerwang}@glenfly.com

Introduction

Leveraging cache locality can effectively decrease the probability of memory stall in GPGPU. There are two cache locality types:

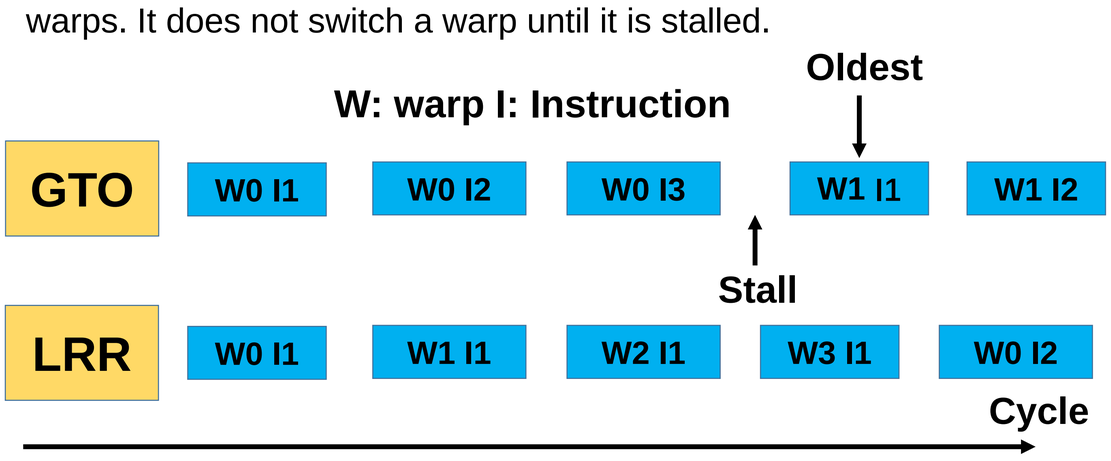
- **Inter-warp locality**: a cache line is accessed by the adjacent warp
- **Intra-warp locality**: a cache line is accessed by the same warp



In each cycle, the hardware warp scheduler selects a **ready warp** and issues it to the execution unit.

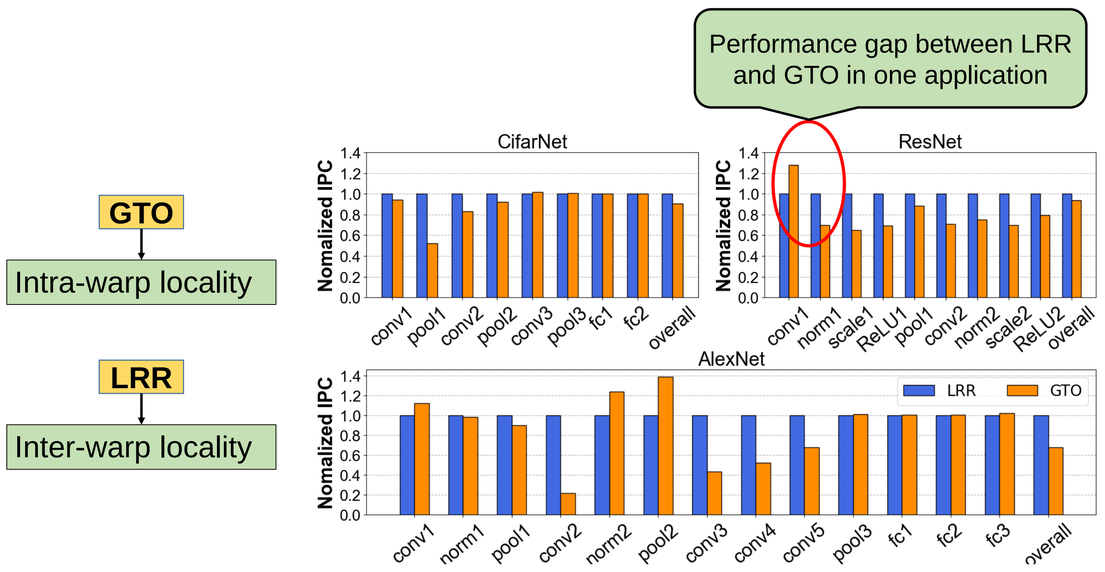
When one warp is stalled, the scheduler will issue another ready warp. Basically, there are two commonly used baseline scheduling policies in typical schedulers:

- **Loosely-Round-Robin (LRR):** equal priority has been given to each warp and switching warps in every cycle
- **Greedy-Then-Oldest (GTO):** prioritizing older warps over younger warps. It does not switch a warp until it is stalled.



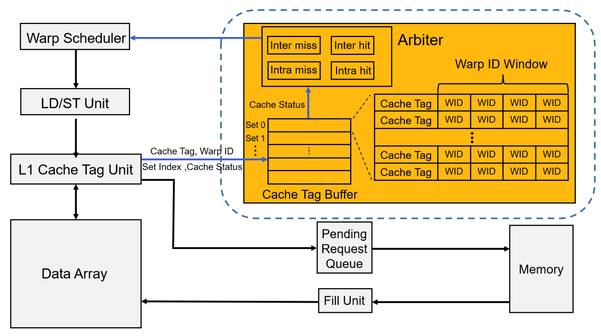
Motivation

- LRR-based scheduler in general can better exploit inter-warp locality
- GTO-based scheduler would be more suitable for intra-warp locality
- Single warp scheduling policy cannot fit all workloads

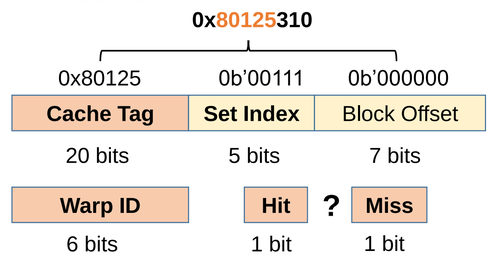


Method

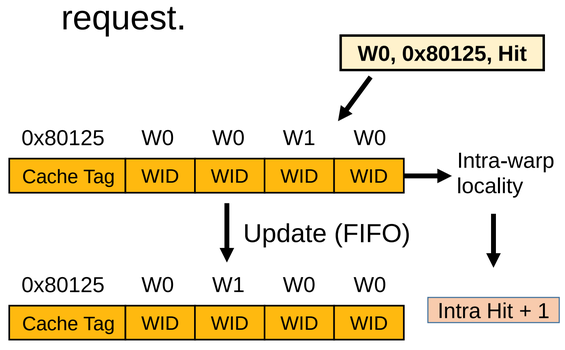
Configuration: 16KB L1 Data Cache
32 Set, 4 Way, 128B per cache line



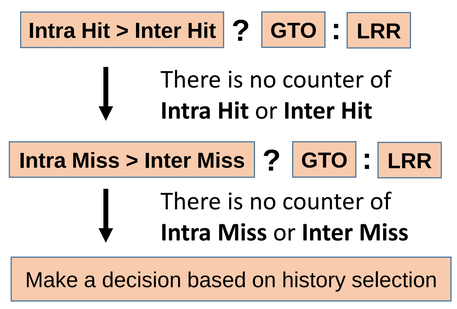
1 For every L1 data cache read request, the cache tag, warp ID, and cache status (hit or miss) are stored in the cache tag buffer



2 If there is an entry for this cache tag in the cache tag buffer, decide and count the cache locality type. Then update the entry. Otherwise, find a new entry for this read request.



3 Decide the warp scheduling policy every 2000 cycles based on the counters of locality type.

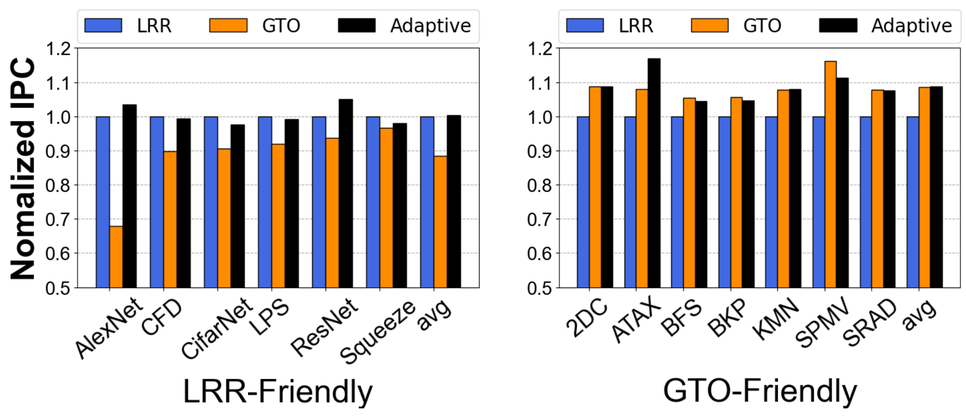


Evaluation

For most applications, the proposed scheduling mechanism is able to select the appropriate scheduling policy and approaches the better performance of two baseline schedulers.

GPGPU-Sim Configuration

Parameter	Value
Number of SMs	15
Max number of Warps per SM	48
Max number of Blocks per SM	8
Number of Schedulers per SM	2
Number of Registers per SM	32768
L1 Data Cache	16KB per SM (32-sets/4-ways)
L1 Inst Cache	2KB per SM (4-sets/4-ways)
L2 Cache	768KB (64-sets/8-ways)
DRAM	924MHZ, FR-FCFS scheduler



Conclusion

- we propose a scheduling mechanism that analyzes the cache locality type and adaptively selects the warp scheduler between GTO and LRR at runtime.
- Evaluation results show that the proposed scheduling mechanism can select the better scheduler between LRR and GTO in most cases.

IEEE SOCC 2022