

Understanding GPU performance

How to get peak FLOPS (GPU version)

Kenjiro Taura

Contents

1 Data Access Performance

Contents

1 Data Access Performance

Data access performance

- data access performance is important in GPU too
-

Memory organization

- Pascal (P100)

level	line size	capacity	associativity
L1	32B	24KB/SM	?
L2	32B	4MB/device	?
Global Memory		12/16GB	N/A
Shared Memory		64KB (*)	N/A

- Volta (V100)

level	line size	capacity	associativity
L1	32B	32-128 KB/SM (*)	?
L2	32B	6MB/device	?
Global Memory		16GB	N/A
Shared Memory		$\leq 96\text{KB (*)}$	N/A

* : 128KB is split between L1 and Shared Memory
(configurable)

source: <https://arxiv.org/abs/1804.06826>

Global vs. Shared Memory

- global memory and L1/L2 cache are the ordinary memory that make a hierarchy
 - cudaMalloc returns a global memory
 - accesses to global memory are transparently cached into L1/L2 caches
- shared memory is an explicitly-managed scratch memory
 - latency shorter than L1 (esp. on Pascal)
 - you explicitly move between global and shared memory
 - data shared only within a thread block
 - programming interface is covered shortly

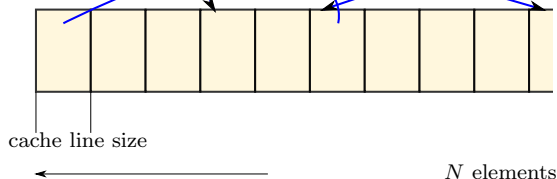
Latency measurement

- the same pointer chasing experiment as we did on CPU

```
1 for (N times) {  
2   p = p->next;  
3 }
```

next pointers

(link all elements in a random order)



Data size vs. latency

- even L1 cache hit takes 30 (Volta) - 100 (Pascal) cycles
out/tex/data/10mem_gpu/latency

Shared memory