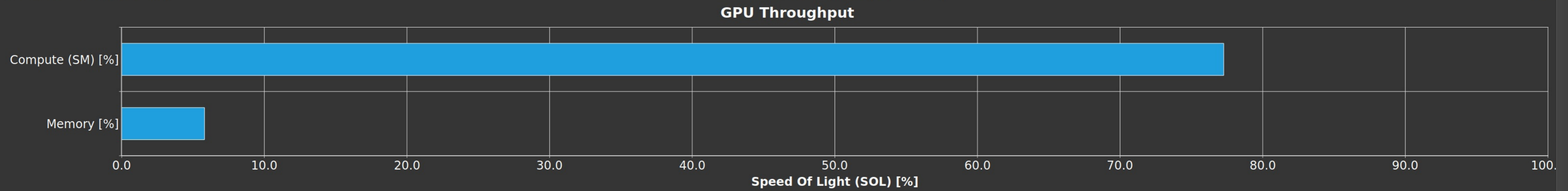


GPU Speed Of Light Throughput

All

High-level overview of the throughput for compute and memory resources of the GPU. For each unit, the throughput reports the achieved percentage of utilization with respect to the theoretical maximum. Breakdowns show the throughput for each individual sub-metric of Compute and Memory to clearly identify the highest contributor.

Compute (SM) Throughput [%]	77.27	Duration [msecond]	6.9
Memory Throughput [%]	5.81	Elapsed Cycles [cycle]	4,515,91
L1/TEX Cache Throughput [%]	6.35	SM Active Cycles [cycle]	4,130,981.2
L2 Cache Throughput [%]	4.40	SM Frequency [usecond]	652.7



Recommendations

**Bottleneck** [Error] Rule Bottleneck returned an error: Metric sm\_sol\_pct not found

**Bottleneck** [Error] <built-in function IAction\_metric\_by\_name> returned a result with an error set /usr/lib/nsight-compute/sections/SpeedOfLight.py:28 /usr/lib/nsight-compute/sections/NvRules.py:274

GPU Speed Of Light Roofline Chart

High-level overview of the utilization for compute and memory resources of the GPU presented as a roofline chart.

Compute Workload Analysis

Detailed analysis of the compute resources of the streaming multiprocessors (SM), including the achieved instructions per clock (IPC) and the utilization of each available pipeline. Pipelines with very high utilization might limit the overall performance.

Executed Ipc Elapsed [cycle]	1.13	SM Busy [%]	84.4
Executed Ipc Active [cycle]	1.24	Issue Slots Busy [%]	31.0
Issued Ipc Active [cycle]	1.24		

Memory Workload Analysis

All

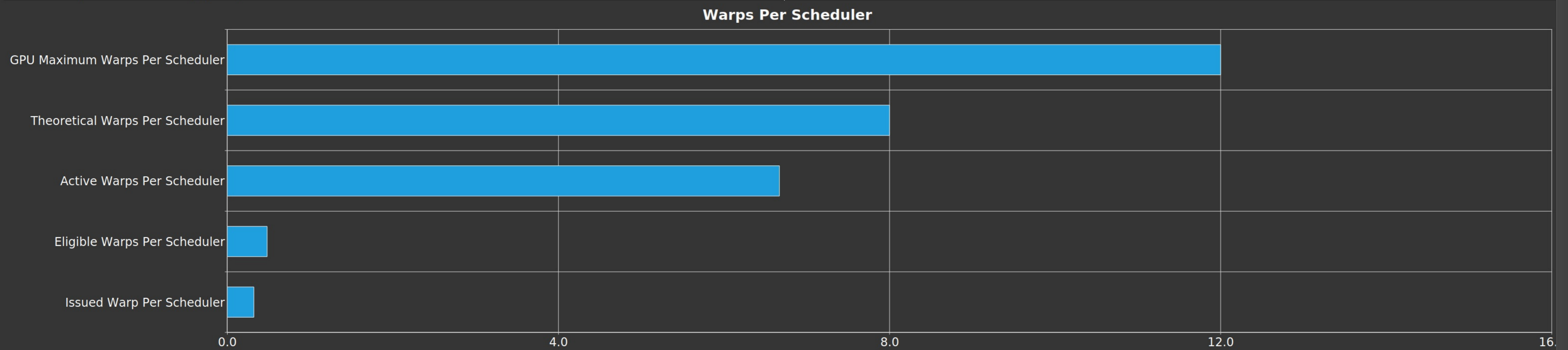
Detailed analysis of the memory resources of the GPU. Memory can become a limiting factor for the overall kernel performance when fully utilizing the involved hardware units (Mem Busy), exhausting the available communication bandwidth between those unit (Max Bandwidth), or by reaching the maximum throughput of issuing memory instructions (Mem Pipes Busy). Detailed chart of the memory units. Detailed tables with data for each memory unit.

Mem Busy [%]	5.81	L1/TEX Hit Rate [%]	97.6
Max Bandwidth [%]	4.40	L2 Hit Rate [%]	99.2
Mem Pipes Busy [%]	11.99	-	

Scheduler Statistics

Summary of the activity of the schedulers issuing instructions. Each scheduler maintains a pool of warps that it can issue instructions for. The upper bound of warps in the pool (Theoretical Warps) is limited by the launch configuration. On every cycle each scheduler checks the state of the allocated warps in the pool (Active Warps). Active warps that are not stalled (Eligible Warps) are ready to issue their next instruction. From the set of eligible warps the scheduler selects a single warp from which to issue one or more instructions (Issued Warp). On cycles with no eligible warps, the issue slot is skipped and no instruction is issued. Having many skipped issue slots indicates poor latency hiding.

Active Warps Per Scheduler [cycle]	6.67	No Eligible [%]	67.9
Eligible Warps Per Scheduler [cycle]	0.48	One or More Eligible [%]	32.0
Issued Warp Per Scheduler [cycle]	0.32		



**Issue Slot Utilization** [Error] Rule Issue Slot Utilization returned an error: Metric smps\_issue\_active\_avg\_per\_active\_cycle not found

**Issue Slot Utilization** [Error] <built-in function IAction\_metric\_by\_name> returned a result with an error set /usr/lib/nsight-compute/sections/IssueSlotUtilization.py:28 /usr/lib/nsight-compute/sections/NvRules.py:274

Warp State Statistics

Analysis of the states in which all warps spent cycles during the kernel execution. The warp states describe a warp's readiness or inability to issue its next instruction. The warp cycles per instruction define the latency between two consecutive instructions. The higher the value, the more warp parallelism is required to hide this latency. For each warp state, the chart shows the average number of cycles spent in that state per issued instruction. Stalls are not always impacting the overall performance nor are they completely avoidable. Only focus on stall reasons if the schedulers fail to issue every cycle. When executing a kernel with mixed library and user code, these metrics show the combined values.

Warp Cycles Per Issued Instruction [warp/inst]	20.80	Avg. Active Threads Per Warp [thread/inst]	10.4
Warp Cycles Per Executed Instruction [warp/inst]	20.88	Avg. Not Predicated Off Threads Per Warp [thread/inst]	8.3

Instruction Statistics

Statistics of the executed low-level assembly instructions (SASS). The instruction mix provides insight into the types and frequency of the executed instructions. A narrow mix of instruction types implies a dependency on few instruction pipelines, while others remain unused. Using multiple pipelines allows hiding latencies and enables parallel execution. Note that 'Instructions/Opcode' and 'Executed Instructions' are measured differently and can diverge if cycles are spent in system calls.

Executed Instructions [inst]	20,474,857	Avg. Executed Instructions Per Scheduler [inst]	1,279,678.5
Issued Instructions [inst]	20,548,208	Avg. Issued Instructions Per Scheduler [inst]	1,284,26

NVLink Topology

NVLink Topology diagram shows logical NVLink connections with transmit/receive throughput.

NVLink Tables

Detailed tables with properties for each NVLink.

Launch Statistics

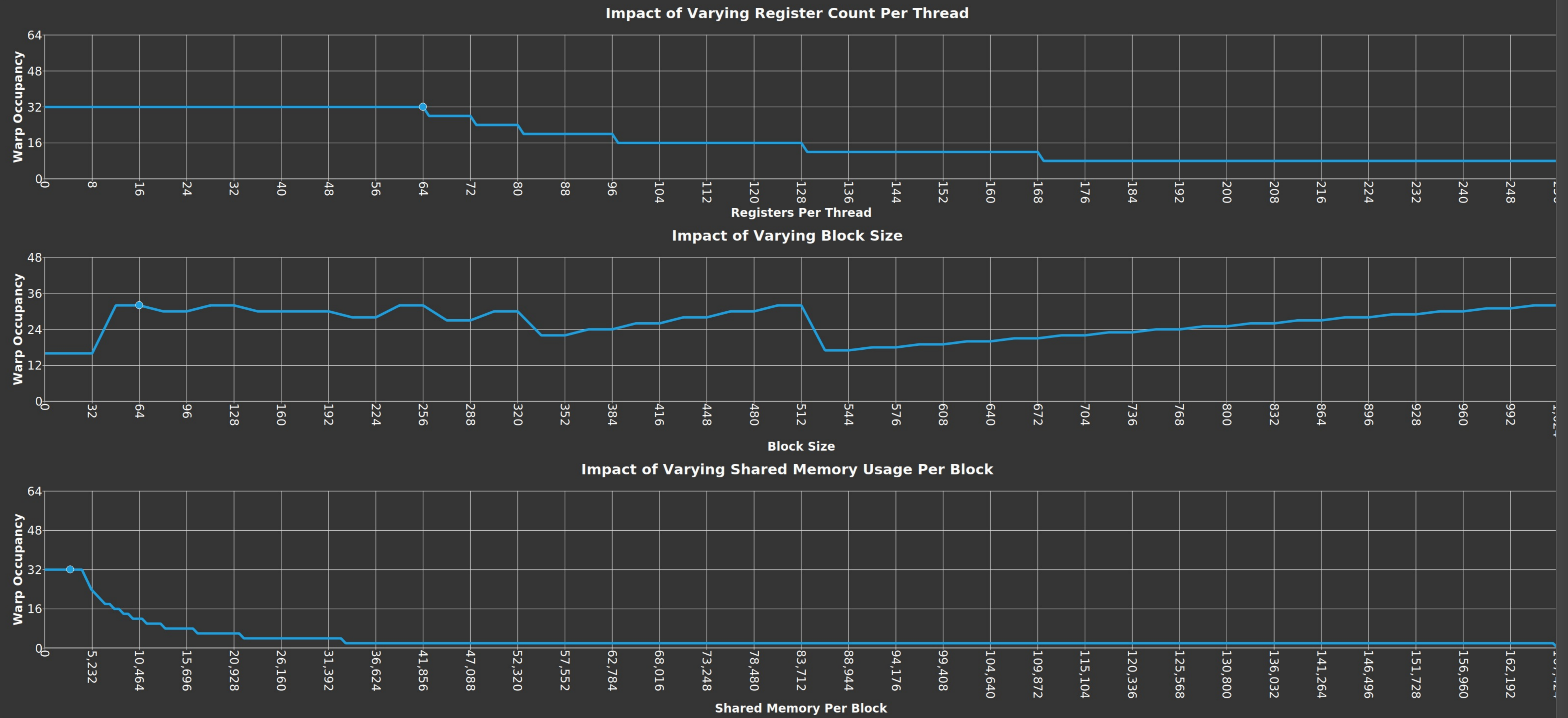
Summary of the configuration used to launch the kernel. The launch configuration defines the size of the kernel grid, the division of the grid into blocks, and the GPU resources needed to execute the kernel. Choosing an efficient launch configuration maximizes device utilization.

Grid Size	256	Registers Per Thread [register/thread]	6
Block Size	64	Static Shared Memory Per Block [Kbyte/block]	1.7
Threads [thread]	16,384	Dynamic Shared Memory Per Block [byte/block]	
Waves Per SM	4	Driver Shared Memory Per Block [Kbyte/block]	1.0
Function Cache Configuration	cudaFuncCachePreferNone	Shared Memory Configuration Size [Kbyte]	65.5

Occupancy

Occupancy is the ratio of the number of active warps per multiprocessor to the maximum number of possible active warps. Another way to view occupancy is the percentage of the hardware's ability to process warps that is actively in use. Higher occupancy does not always result in higher performance, however, low occupancy always reduces the ability to hide latencies, resulting in overall performance degradation. Large discrepancies between the theoretical and the achieved occupancy during execution typically indicates highly imbalanced workloads.

Theoretical Occupancy [%]	66.67	Block Limit Registers [block]	1
Theoretical Active Warps per SM [warp/cycle]	32	Block Limit Shared Mem [block]	2
Block Limit Warps [block]	24	Block Limit SM [block]	1



Source Counters

All

Source metrics, including branch efficiency and sampled warp stall reasons. Warp Stall Sampling metrics are periodically sampled over the kernel runtime. They indicate when warps were stalled and couldn't be scheduled. See the documentation for a description of all stall reasons. Only focus on stalls if the schedulers fail to issue every cycle.

Branch Instructions [inst]	2,503,740	Branch Instructions Ratio [%]	0.1
----------------------------	-----------	-------------------------------	-----