Current	Result	Size	Time	Cycles	GPU	Frequency	Process	Attributes
1	1/1+ geom_kernel_fp32	32 x 32 x 32	31.82 ms	35,936,003	0 - NVIDIA GeForce RTX 4060 Laptop GPU	1.62 GHz	[Device] geom_fp32	

Session | Details | Connection | Communication | Raw | Operation

Compare | Tools | Close | Export

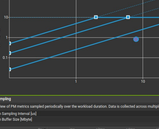
GPU Speed of Light Throughput

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 the Compute and Memory to clearly identify the highest contributor. High-level overview of the utilization for compute and memory resources of the GPU presented on a combined chart.

Compute SOL Throughput [%]	86.74
Memory Throughput [%]	86.74
Compute Throughput [inst/cycle]	1.73
L2 Cache Throughput	6.06
Memory Throughput [Gbyte/s]	541.19

High Throughput This workload is utilizing greater than 60.0% of the available compute or memory performance of the device. To further improve performance, work will likely need to be shifted from the most utilized to another unit. Start by analyzing workloads in the Compute Workload Analysis section.

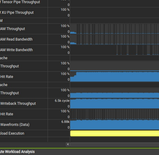
Roofline Analysis The ratio of peak float (FP32) to double (FP64) performance on this device is 64:1. The workload achieved 0% of this device's FP32 peak performance and 0% of its FP64 peak performance. See the Roofline topic for more details on roofline analysis.



PM Sampling

Timeline view of PM metrics compiled periodically over the workload duration. Data is collected across multiple passes. Use this section to understand how workload behavior changes over the runtime.

Maximum Sampling Interval [us]	8
Number of Passes [Multipass]	5.92



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 Active [inst/cycle]	30.25
Executed Ipc Elapsed [inst/cycle]	0.95
Issue Slots Busied [%]	23.76

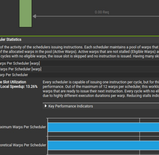
Low Utilization All compute pipelines are under-utilized. Either this workload is very small or it doesn't issue enough warps per scheduler. Check the Occupancy Section and Scheduler Statistics sections for further details.

Memory Workload Analysis

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 these units (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.

Memory Throughput [Gbyte/s]	2.79
L1/TEX Hit Rate [%]	33.33
L2 Hit Rate [%]	97.74
L2 Compression Input Sectors [sector/s]	0
L2 Compression Rate	0
L2 Compression Success Rate [%]	0

Shared Store Bank Conflicts Est. Speedup: 22.10% The memory access pattern for shared stores might not be optimal and causes an average of 1.5 bank conflicts per 128 bank accesses. This results in 9709714 bank conflicts, which represent 29.39% of the overall 33080800 wavefronts for shared stores. Check the Shared Memory section for uncoalesced shared stores.

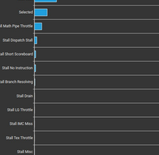


Scheduler Statistics

Analysis 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 status of the allocated warps to pick one eligible warp. 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 Warps). On average, this number tells how many of the available instruction dispatchers are being used.

Active Warps Per Scheduler [warp]	8.49
Eligible Warps Per Scheduler [warp]	1.18

Issue Slot Utilization Est. Local Speedup: 12.26% Every scheduler is capable of issuing one instruction per cycle, but for this workload each scheduler only issues an instruction every 4.3 cycles. This might leave hardware resources underutilized and may lead to lower overall performance. The utilization chart of the scheduler and instruction pipelines in the Compute Workload Analysis section may be helpful to find the reason for the issue slots being not fully utilized. Issue slots can be stalled for various reasons and profiling the warp stalls may be helpful to identify which stalls impose the most severe impact on the issue rate. Every cycle with no eligible warps results in no instruction being issued and the issue slot remains unused. To increase the number of eligible warps, avoid possible load imbalances due to highly different execution durations per warp. Reducing stalls indicated on the Warps Scheduler and Warp Stall Sampling sections can help, too.

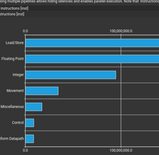


Warp State Statistics

Analysis of the activity of 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 latency needs to be hidden by other warps to achieve peak throughput. For each of the warp states, a chart is provided that shows the distribution of the number of warps in that state across all cycles. Stalls are not always impacting the overall performance or are they completely avoidable. Only focus on stall reasons if the scheduler fail to issue every cycle. When executing a kernel with mixed library and user code, these metrics show the combined values.

Warps Cycles Per Issued Instruction [cycle]	20.55
Warps Cycles Per Executed Instruction [cycle]	20.55

Stc Threads Stall Est. Speedup: 13.26% On average, each warp of this workload spends 21.1 cycles being stalled waiting for the MIO (memory input/output) instruction queue to be not full. This stall reason is high in cases of extreme utilization of the MIO pipelines, which include special math instructions, dynamic branches, as well as shared memory instructions. When caused by shared memory accesses, trying to use fewer but wider loads can reduce pipeline pressure. This stall type represents about 63.1% of the total average of 33.4 cycles between issuing two instructions.



Instruction Statistics

Analysis of the executed low-level assembly instructions (SASS). The instruction mix provides insight into the type and frequency of the executed instructions. A narrow mix of instruction types implies a dependency on few instruction pipelines, while others remain idle. Different instruction types utilizes finishing latencies and enables parallel execution. Note that "Instructions Executed" counts all executed instructions, and "Instructions Issued" counts only the first issue of each instruction. Cycles are spent in system calls.

Average Executed Instructions / Scheduler [inst]	815,059.903
Issued Instructions [inst]	815,056.154

NVLink Topology

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

NVLink Topology

The system does not have any NVLink connections.

NVLink Tables

Detailed tables with properties for each NVLink.

Logical NVLink Properties

The system does not have any NVLink connections.

Nvlink Infinity

Non-uniform memory access (NUMA) affinities based on compute and memory distances for all GPUs.

NUMA ID Table

GPU ID	GPU Name	NUMA ID by GPU Affinity	CPU Affinity	NUMA ID by Memory Affinity
0	NVIDIA GeForce RTX 4060 Laptop GPU	0-15	0	0

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.

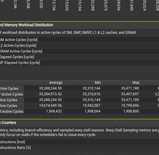
Block Size	1,024	Dynamic Shared Memory Per Block [Byte/block]	0
Threads [thread]	4,194,304	Driver Shared Memory Per Block [Byte/block]	1,152
Waves Per SM	175.67	Shared Memory Configuration Size [Kbyte]	16,384
Uses Green Context	0	Stack Size	1,024
Cluster Size	0	# TPCs	17
Enabled TPC IDs	all		

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
Theoretical Active Warps per SM [warp]	32
Achieved Occupancy [%]	66.64
Achieved Active Warps Per SM [warp]	31.99

Theoretical Occupancy Est. Speedup: 16.66% The 8.00 theoretical warps per scheduler this kernel can issue according to its occupancy are below the hardware maximum of 12. This kernel's theoretical occupancy (66.7%) is limited by the number of required registers, and the number of warps within each block.



GPU and Memory Workload Distribution

Analysis of workload distribution in active cycles of SM, DRAM, SMSP (TEX), L1 & L2 caches, and DRAM

Average Active Cycles [cycle]	35,280,244.70	Average L1 Active Cycles [cycle]	35,289,244.39	
Average L2 Active Cycles [cycle]	19,673,549.56	Average SMSP Active Cycles [cycle]	35,344,975.42	
Average DRAM Active Cycles [cycle]	1,908,452	Total Elapsed Cycles [cycle]	852,216,496	
Total L1 Elapsed Cycles [cycle]	852,216,496	Total L2 Elapsed Cycles [cycle]	852,216,496	
Total DRAM Elapsed Cycles [cycle]	3,408,897,984	Total DRAM Elapsed Cycles [cycle]	797,798,976	

Workload Distribution

	Average	Min	Max	Total
SM Active Cycles	35,290,244.70	35,215,144	35,471,188	849,317,888
SMSP Active Cycles	35,344,975.42	35,252,676	35,467,607	3,534,497,542
L1 Active Cycles	35,290,244.70	35,215,144	35,471,188	849,317,888
L2 Active Cycles	19,673,549.56	19,561,087	19,799,695	314,792,793
DRAM Active Cycles	1,908,452	1,906,064	1,908,800	7,633,729

Source Counters

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.

Branch Instructions [inst]	10,524	Branch Efficiency [%]	100
Branch Instructions Ratio [%]	0.01	Avg. Divergent Branches [branches]	0

Warp Stall Sampling (All Samples)

Location	Value [%]
0x7f8c1f7c2a00 in gemm_kernel_fp32	10,524
0x7f8c1f7c2a10 in gemm_kernel_fp32	20,058
0x7f8c1f7c2a20 in gemm_kernel_fp32	20,139
0x7f8c1f7c2a30 in gemm_kernel_fp32	10,524
0x7f8c1f7c2a40 in gemm_kernel_fp32	10,524

Most Instructions Executed

Location	Value
0x7f8c1f7c2a00 in gemm_kernel_fp32	8,388,608
0x7f8c1f7c2a10 in gemm_kernel_fp32	8,388,608
0x7f8c1f7c2a20 in gemm_kernel_fp32	8,388,608
0x7f8c1f7c2a30 in gemm_kernel_fp32	8,388,608
0x7f8c1f7c2a40 in gemm_kernel_fp32	8,388,608

Follow the rules subject to get guidance on how to navigate through the report and quickly discover performance bottlenecks in this kernel.
You could also disable Source Counters to focus on selected performance aspects and make profiling faster.