Current	1790 - gemm\nc2dBlockKTiling	Size	Time	Cycles	GPU	SM Frequency	Process	Attributes
		(32, 32, 1)x(256, 1, 1)	35.95 ms	32,258,062	0 - NVIDIA GeForce RTX 3060 Laptop GPU	900.00 MHz	[24915] ncu_vectorize_gemm	

Summary | Details | Source | Context | Comments | Raw | Session — Compare, Rails, View, Export

GPU Speed Of Light Throughput

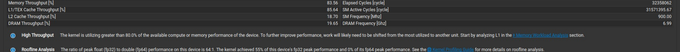
High-level overview of the throughput for compute and memory resources of the GPU. For each unit, the throughput reports the achieved percentage of utilization with respect to the theoretical maximum. Breakdowns show the throughput for each individual sub-metric of Compute and Memory to clearly identify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

Compute (SM) [%]	62.61	Duration [ms]	35.95
Memory Throughput [%]	85.56	Elapsed Cycles [cycle]	32258062
L1/TEX Cache Throughput [%]	89.44	SM Active Cycles [cycle]	31577395.67
L2 Cache Throughput [%]	18.70	SM Frequency [MHz]	900.00
DRAM Throughput [%]	19.68	DRAM Frequency [GHz]	6.99

High Throughput The kernel is utilizing greater than 80.0% 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 L1 in the Memory Workload Analysis section.

Roofline Analysis The ratio of peak float (fp32) to double (fp64) performance on this device is 64:1. The kernel achieved 55% of this device's fp32 peak performance and 0% of its fp64 peak performance. See the Roofline Chart for more details on roofline analysis.

GPU Throughput



Compute Throughput Breakdown

SM Issue Active [%]	62.61
SM Inst Executed [%]	62.61
SM Pipe Fma Cycles Active [%]	56.29
SM Pipe Fmaheavy Cycles Active [%]	55.89
SM Inst Executed Pipe Lsu [%]	33.76
SM Mio2rf Writeback Active [%]	22.50
SM Mio Inst Issued [%]	11.83
SM Pipe Alu Cycles Active [%]	3.71
SM Mio Pq Read Cycles Active [%]	3.51
SM Mio Pq Write Cycles Active [%]	3.51
SM Inst Executed Pipe Adu [%]	1.73
SM Inst Executed Pipe Uniform [%]	0.44
SM Inst Executed Pipe Cbu Pred On Any [%]	0.01
SM Pipe Tensor Cycles Active [%]	0
SM Pipe Fp64 Cycles Active [%]	0
SM Memory Throughput Internal Activity [%]	0
SM Inst Executed Pipe Xu [%]	0
SM Sysmem Throughput Internal Activity [%]	0
SM Inst Executed Pipe Ad [%]	0
SM Inst Executed Pipe Tex [%]	0
SM Inst Executed Pipe Qn [%]	0

Memory Throughput Breakdown

L1 Data Pipe Lsu Wavefronts [%]	83.56
L1 Lsu Writeback Active [%]	45.23
L1 Lsuin Requests [%]	33.76
DRAM Dram Sectors [%]	19.65
DRAM Cycles Active [%]	19.55
L2 T Sectors [%]	18.70
L2 Lts2dsr Cycles Active [%]	18.18
L1 Data Bank Reads [%]	16.47
L1 M Xbar2Tmu Read Sectors [%]	14.05
GPU: Compute Memory Access Throughput Internal Activity [%]	11.38
L2 T Tag Requests [%]	11.38
L2 Xbar2Dp Cycles Active [%]	10.77
L2 S Sectors Fill Device [%]	9.55
L1 M L1tag2dsr Req Cycles Active [%]	8.32
L2 D Sectors [%]	7.13
L1 Data Bank Writes [%]	3.51
L2 Lts2xbar Req Cycles Active [%]	3.51
L1 Tex Cycles Active [%]	0.88
L1 Tex Writeback Active [%]	0
L2 Xbarmux Input Cycles Active [%]	0
L1 T Wavefronts [%]	0
L1 Data Pipe Tex Wavefronts [%]	0
GPU: Compute Memory Request Throughput Internal Activity [%]	0

Floating Point Operations Roofline

PM Sampling

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

Maximum Sampling Interval [us]	256	# Pass Groups	2
Maximum Buffer Size [MBytes]	16.38	Dropped Samples [sample]	2

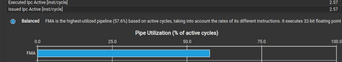
Compute Workload Analysis

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.

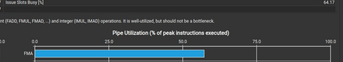
Executed Ipc Elapsed [inst/cycle]	2.99	SM Busy [%]	64.17
Executed Ipc Active [inst/cycle]	2.97	Issue Slots Busy [%]	64.17
Issued Ipc Active [inst/cycle]	2.97		

Balanced FMA is the highest-utilized pipeline (57.6%) based on active cycles, taking into account the rates of its different instructions. It executes 32-bit floating point (FADD, FMUL, FMAD, ...) and integer (IMUL, IMAD) operations. It is well-utilized, but should not be a bottleneck.

Pipe Utilization (% of active cycles)



Pipe Utilization (% of peak instructions executed)



Memory Workload Analysis

Memory Chart

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 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 [GBytes/s]	65.65	Mem Busy [%]	83.56
L1/TEX Hit Rate [%]	4.92	Max Bandwidth [%]	45.23
L2 Hit Rate [%]	48.92	Mem Pipes Busy [%]	33.76
L2 Compression Success Rate [%]	0	L2 Compression Ratio	0

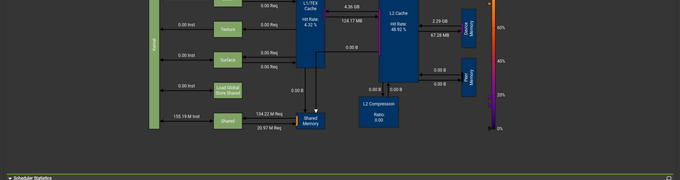
L1TEX Global Load Access Pattern Est. Speedup: 1.27%
The memory access pattern for global loads from L1TEX might not be optimal. On average, only 31.5 of the 32 bytes transmitted per sector are utilized by each thread. This could possibly be caused by a stride between threads. Check the Kernel Counters section for uncoalesced global loads.

L1TEX Global Store Access Pattern Est. Speedup: 41.78%
The memory access pattern for global stores to L1TEX might not be optimal. On average, only 16.0 of the 32 bytes transmitted per sector are utilized by each thread. This could possibly be caused by a stride between threads. Check the Kernel Counters section for uncoalesced global stores.

Shared Load Bank Conflicts Est. Speedup: 94.26%
The memory access pattern for shared loads might not be optimal and causes on average a 5.0 - way bank conflict across all 134217728 shared load requests.This results in 268435456 bank conflicts, which represent 40.00% of the overall 671092564 wavefronts for shared loads. Check the Kernel Counters section for uncoalesced shared loads.

Shared Store Bank Conflicts Est. Speedup: 28.93%
The memory access pattern for shared stores might not be optimal and causes on average a 2.4 - way bank conflict across all 20971520 shared store requests.This results in 16777216 bank conflicts, which represent 33.33% of the overall 50331648 wavefronts for shared stores. Check the Kernel Counters section for uncoalesced shared stores.

Memory Chart

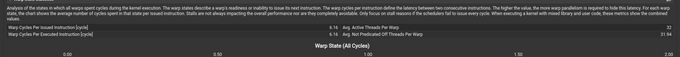


Scheduler Statistics

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 ins- tructions (Issued Warps). 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 [warp]	3.95	No Eligible [%]	35.82
Eligible Warps Per Scheduler [warp]	1.64	One or More Eligible [%]	64.18
Issued Warp Per Scheduler	0.64		

Warps Per Scheduler

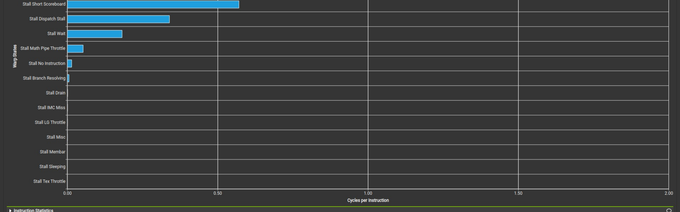


Warp State Statistics

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 [cycle]	6.16	Avg. Active Threads Per Warp	32
Warp Cycles Per Executed Instruction [cycle]	6.16	Avg. Not Predicated Off Threads Per Warp	31.94

Warp State (All Cycles)



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]	243099239	Avg. Executed Instructions Per Scheduler [inst]	202582691.87
Issued Instructions [inst]	243106427	Avg. Issued Instructions Per Scheduler [inst]	202588368.93

NVLink Topology

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

NVLink Tables

Detailed tables with properties for each NVLink.

NUMA Affinity

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

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	1024	Function Cache Configuration	CachePreferNone
Registers Per Thread [register/thread]	116	Static Shared Memory Per Block [Kbyte/block]	8.19
Block Size	256	Dynamic Shared Memory Per Block [bytes/block]	0
Threads [thread]	262144	Driver Shared Memory Per Block [Kbyte/block]	1.02
Waves Per SM	11.07	Shared Memory Configuration Size [Kbyte]	65.54
Uses Green Context	0	# SMs [SM]	30

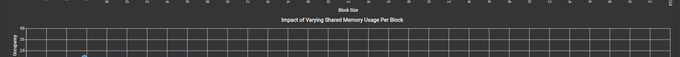
Occupancy

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.

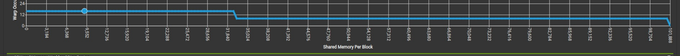
Theoretical Occupancy [%]	33.33	Block Limit Registers [block]	2
Theoretical Active Warps per SM [warp]	32	Block Limit Shared Mem [block]	7
Achieved Occupancy [%]	32.94	Block Limit Warps [block]	4
Achieved Active Warps Per SM [warp]	15.81	Block Limit SM [block]	16

Theoretical Occupancy The 4.00 theoretical warps per scheduler this kernel can issue according to its occupancy are below the hardware maximum of 12. This kernel's theoretical occupancy (33.3%) is limited by the number of required registers.

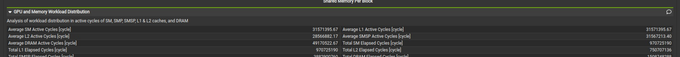
Impact of Varying Register Count Per Thread



Impact of Varying Block Size



Impact of Varying Shared Memory Usage Per Block



SPU and Memory Workload Distribution

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

Average SM Active Cycles [cycle]	31577395.67	Average L1 Active Cycles [cycle]	31577395.67
Average L2 Active Cycles [cycle]	28968861.17	Average SMSP Active Cycles [cycle]	31567233.40
Average DRAM Active Cycles [cycle]	6917202.67	Total SM Elapsed Cycles [cycle]	970725190
Total L1 Elapsed Cycles [cycle]	970725190	Total L2 Elapsed Cycles [cycle]	790707136
Total DRAM Elapsed Cycles [cycle]	380250760	Total DRAM Elapsed Cycles [cycle]	1009748288

Workload Distribution

	Average	Min	Max	Sum
SM Active Cycles	31577395.67	31447611	32358588	947147870
SMSP Active Cycles	31567233.40	31442339	32356437	378806808
L1 Active Cycles	31577395.67	31447611	32358588	947147870
L2 Active Cycles	28968861.17	28437838	29591470	685605172
DRAM Active Cycles	6917202.67	4914120	8973128	209027136

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. Only focus on stalls if the schedulers fail to issue every cycle.

Branch Instructions [inst]	8404992	Branch Efficiency [%]	100
Branch Instructions Ratio [%]	0.04	Avg. Divergent Branches	0

Uncoalesced Global Accesses Est. Speedup: 2.68%
This kernel has uncoalesced global accesses resulting in a total of 4194304 excessive sectors (3% of the total 142606336 sectors). Check the L2 Theoretical Sectors Global Excessive table for the primary source locations. The CUDA Programming Guide has additional information on reducing uncoalesced device memory accesses.

L2 Theoretical Sectors Global Excessive

Location	Value	Value [%]
L0_vectorize_gemm.cu:42 (0x000d0b0d0) in gemm\nc2dBlockKTiling	131,072	3
L0_vectorize_gemm.cu:42 (0x000d0b0d0) in gemm\nc2dBlockKTiling	131,072	3
L0_vectorize_gemm.cu:42 (0x000d0b0d0) in gemm\nc2dBlockKTiling	131,072	3
L0_vectorize_gemm.cu:42 (0x000d0b0d0) in gemm\nc2dBlockKTiling	131,072	3
L0_vectorize_gemm.cu:42 (0x000d0b0d0) in gemm\nc2dBlockKTiling	131,072	3

Uncoalesced Shared Accesses Est. Speedup: 38.57%
This kernel has uncoalesced shared accesses resulting in a total of 28521267 excessive wavefronts (46% of the total 721420288 wavefronts). Check the L1 Wavefronts Shared Excessive table for the primary source locations. The CUDA Optimization Guide has an example on optimizing shared memory accesses.

L1 Wavefronts Shared Excessive

Location	Value	Value [%]
L0_vectorize_gemm.cu:48 (0x000d0b0e0) in gemm\nc2dBlockKTiling	16,777,216	29
L0_vectorize_gemm.cu:48 (0x000d0b0e0) in gemm\nc2dBlockKTiling	16,777,216	29
L0_vectorize_gemm.cu:48 (0x000d0b0e0) in gemm\nc2dBlockKTiling	16,777,216	29
L0_vectorize_gemm.cu:48 (0x000d0b0e0) in gemm\nc2dBlockKTiling	16,777,216	29
L0_vectorize_gemm.cu:48 (0x000d0b0e0) in gemm\nc2dBlockKTiling	16,777,216	29

Warp Stall Sampling (All Samples)

Location	Value [%]
L0_vectorize_gemm.cu:47 (0x000d0b0d0) in gemm\nc2	145,135
L0_vectorize_gemm.cu:51 (0x000d0b0f0) in gemm\nc2	92,478
L0_vectorize_gemm.cu:48 (0x000d0b0e0) in gemm\nc2	59,000
L0_vectorize_gemm.cu:51 (0x000d0b0e0) in gemm\nc2	29,483
L0_vectorize_gemm.cu:51 (0x000d0b0d0) in gemm\nc2	5,856

Most Instructions Executed

Location	Value	Value [%]
L0_vectorize_gemm.cu:42 (0x000d0b0d0) in gemm\nc2	4,194,304	0
L0_vectorize_gemm.cu:42 (0x000d0b0d0) in gemm\nc2	4,194,304	0
L0_vectorize_gemm.cu:42 (0x000d0b0d0) in gemm\nc2	4,194,304	0
L0_vectorize_gemm.cu:42 (0x000d0b0d0) in gemm\nc2	4,194,304	0
L0_vectorize_gemm.cu:42 (0x000d0b0d0) in gemm\nc2	4,194,304	0

Follow the rules outputs to get guidance on how to navigate through the report and quickly discover performance bottlenecks in this kernel.
You could also disable rules that are not relevant to focus on selected performance aspects and make profiling faster.