CUDA getUnrestricted Kernel Execution Time Prediction Report

This report applies the CUDA kernel performance prediction template to the kernel: getUnrestricted, launched with Grid = (5,5) and Block = (32,32).

We estimate the expected per-launch execution time on five NVIDIA GPUs:
• GeForce GTX TITAN Black
• GeForce GTX TITAN X
• NVIDIA TITAN V
• GeForce RTX 2080 Ti
• GeForce RTX 4070

1. Effective Parallelism

Launch configuration:
• Grid = (5,5) → 25 blocks
• Block = (32,32) → 1024 threads/block
• THREADSPERDIM = 16
• countx = 5, county = 5
• rows = 100, cols = 10

The kernel uses:
n = blockIdx.x * THREADSPERDIM + threadIdx.x;
m = blockIdx.y * THREADSPERDIM + threadIdx.y;

With gridDim.x = gridDim.y = 5 and THREADSPERDIM = 16, n and m each range from 0 to 95, but only indices with n < countx (5) and m < county (5) do work.

This occurs only in block (blockIdx.x = 0, blockIdx.y = 0) and threadIdx.x, threadIdx.y in 0..4.

Thus there are 5 × 5 = 25 active (m,n) pairs per launch; all other threads return immediately. Each active thread performs a full Gram–Schmidt QR solve on a 100×10 system (similar to getRestricted).

2. FLOP Count

As in getRestricted, for each (m,n) pair we approximate:
• Gram–Schmidt orthogonalization: ≈ 21,100 FLOPs
• R = Q■X, B = Q■Y, and back-substitution: ≈ 22,100 FLOPs

Total per-thread (per (m,n)) FLOPs:
F_thread ≈ 43,200 FLOPs

With 25 active threads:
F_total = 43,200 × 25 = 1,080,000 ≈ 1.08 × 10■ FLOPs.

3. Memory Traffic (Approximate)

Per active thread we reuse the getRestricted estimate:
B_thread ≈ 1.99 × 10■ bytes (loads/stores of X, Q, R, Y, B).

With 25 active threads:
B_total ≈ 1.99 × 10■ × 25 = 4.975 × 10■ bytes ≈ 4.98 MB per kernel launch.

This again makes the kernel primarily memory bound.

4. GPU Specifications Used

Approximate FP32 peak performance and memory bandwidth:

GPU	Peak FP32 (FLOPs/s)	Bandwidth (bytes/s)
GTX TITAN Black	5.12e12	3.36e11
GTX TITAN X	6.14e12	3.365e11
TITAN V	1.49e13	6.528e11
RTX 2080 Ti	1.345e13	6.16e11
RTX 4070	2.9e13	5.04e11

5. Time Estimates

Using:
• F_total ≈ 1.08 × 10■ FLOPs
• B_total ≈ 4.975 × 10■ bytes

We compute for each GPU:
t_compute = F_total / Peak_FP32
t_mem = B_total / Bandwidth
t_body = max(t_compute, t_mem)
t_total ≈ t_body + 5 µs (kernel launch overhead).

GPU	t_compute (µs)	t_mem (µs)	t_body (µs)	t_total (µs)
GTX TITAN Black	0.211	14.81	14.81	≈ 19.81
GTX TITAN X	0.176	14.78	14.78	≈ 19.78
TITAN V	0.072	7.62	7.62	≈ 12.62
RTX 2080 Ti	0.080	8.08	8.08	≈ 13.08
RTX 4070	0.037	9.87	9.87	≈ 14.87

6. Conclusion

getUnrestricted performs roughly 1.08M FLOPs and moves ~5MB of data per launch, spread over only 25 active (m,n) threads while many threads return immediately.

Given the relatively modest FLOP count and multi-megabyte global memory traffic, the kernel is clearly memory bound. The body time is on the order of 8–15 µs, and adding a ~5 µs CUDA kernel launch overhead yields total times:

• GTX TITAN Black / TITAN X: ≈ 19.8 µs
• TITAN V: ≈ 12.6 µs
• RTX 2080 Ti: ≈ 13.1 µs
• RTX 4070: ≈ 14.9 µs

These estimates are approximate but consistent with expectations for a QR-based kernel with limited parallel occupancy and moderate memory bandwidth usage.