Kimi K2 on RTX PRO 6000 Blackwell

SM120 Deployment Benchmark Report

January 29, 2026

Executive Summary

Successfully deployed Kimi K2 models on 8x RTX PRO 6000 Blackwell (SM120) after developing a custom patch for the Triton attention extend kernel to fix shared memory exhaustion issues.

5,816	985	768 GB
Peak tok/s (K2-Thinking)	Peak tok/s (K2.5 INT4)	Total VRAM (8x96GB)

180K
Max KV Cache Tokens

Hardware Configuration

GPU Specifications

GPUs	8x NVIDIA RTX PRO 6000 Blackwell
Architecture	SM120 (Blackwell)
VRAM per GPU	96 GB GDDR6X
Total VRAM	768 GB
Shared Memory	~100 KB per SM
Interconnect	PCIe only (no NVLink)

GPU Topology

PIX pairs	GPU 0↔1, 2↔3, 4↔5, 6↔7
NUMA nodes	2 (4 GPUs each)
CPU	Dual-socket system
System RAM	~1.5 TB

Note: SM120 (consumer Blackwell) has only ~100KB shared memory vs 228KB on SM100 (datacenter Blackwell).

Models Tested

MODEL	QUANTIZATION	SIZE	EXPERTS	STATUS
Kimi K2.5	INT4 (Marlin)	~557 GB	384 experts, 8+1 active	WORKING
Kimi K2-Thinking	NVFP4 (cuDNN FP4)	~555 GB	384 experts, 8+1 active	WORKING

The Problem & Solution

The Problem

Triton attention extend kernel crashed with:

```
triton.runtime.errors.OutOfResources: out of resource: shared memory, Required: 106496, Hardware limit: 101376
```

The kernel was compiled for Hopper (SM90) with large block sizes that exceeded SM120's shared memory limit.

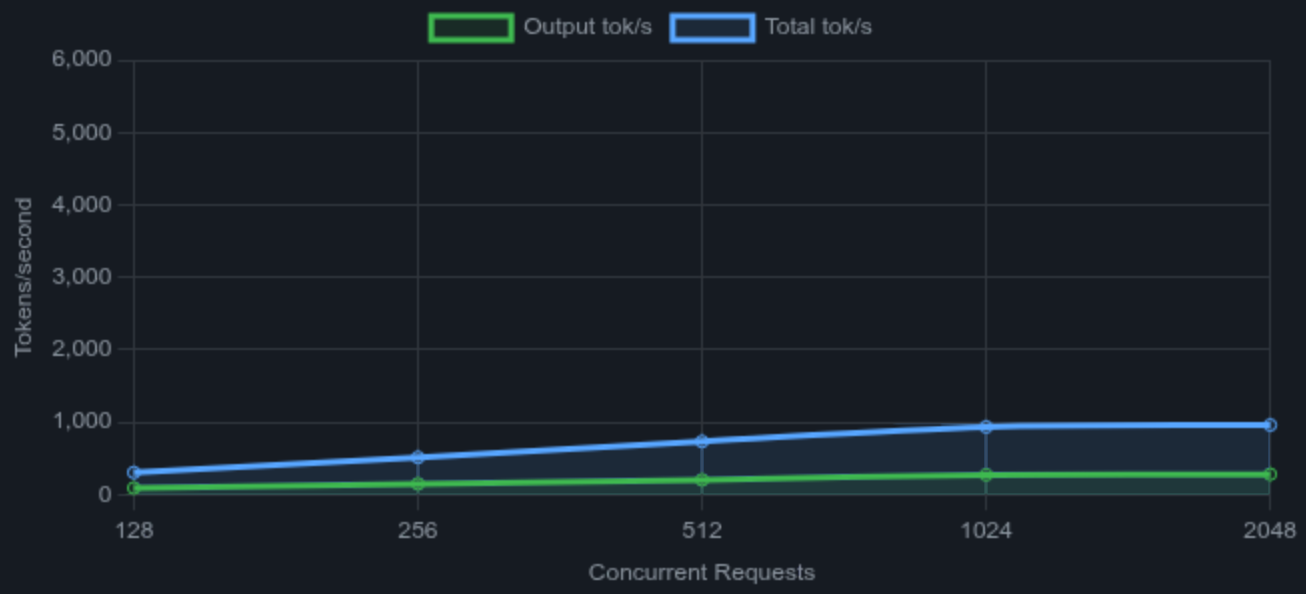
The Solution

Added SM120-specific case in `extend_attention.py`:

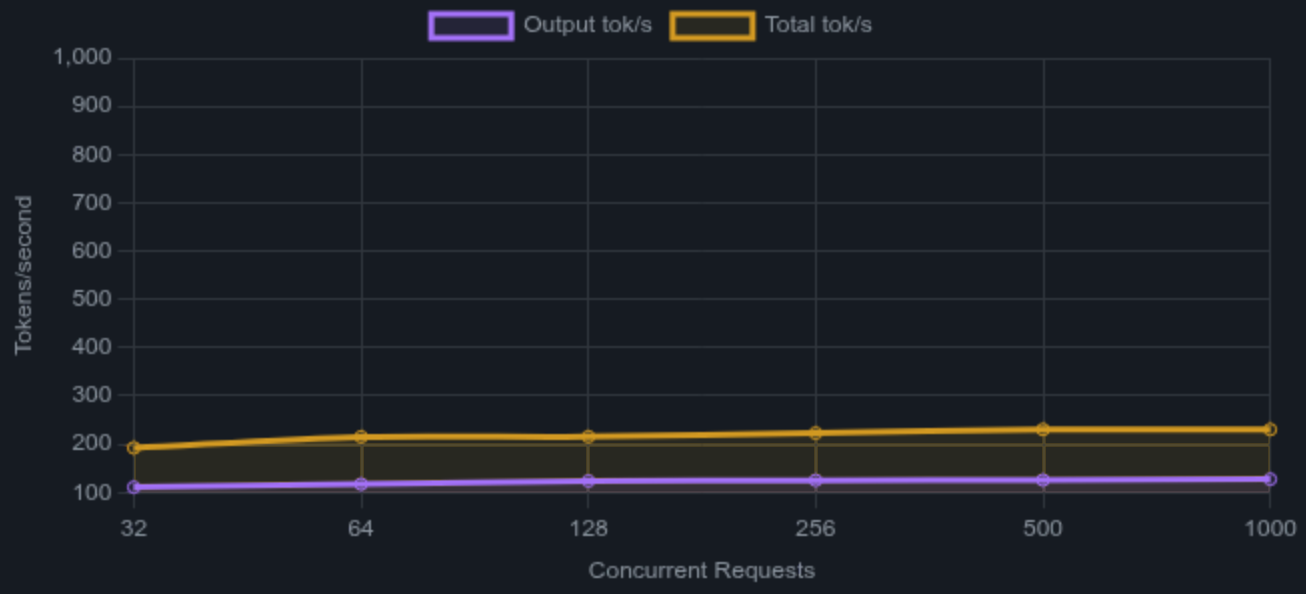
```
elif _is_cuda and CUDA_CAPABILITY[0] == 12: # SM120 Blackwell RTX if Lq <= 128: BLOCK_M, BLOCK_N = (64, 64) elif Lq <= 256: BLOCK_M, BLOCK_N = (32, 64) else: BLOCK_M, BLOCK_N = (32, 32)
```

Performance Benchmarks

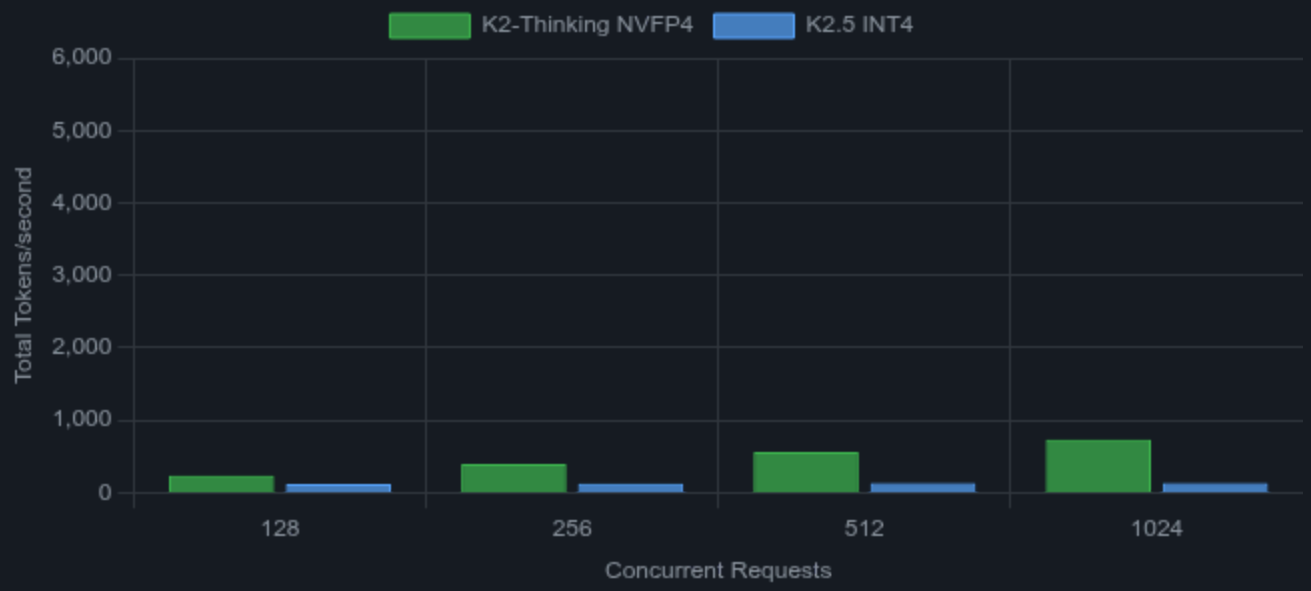
K2-Thinking NVFP4: Throughput vs Concurrency



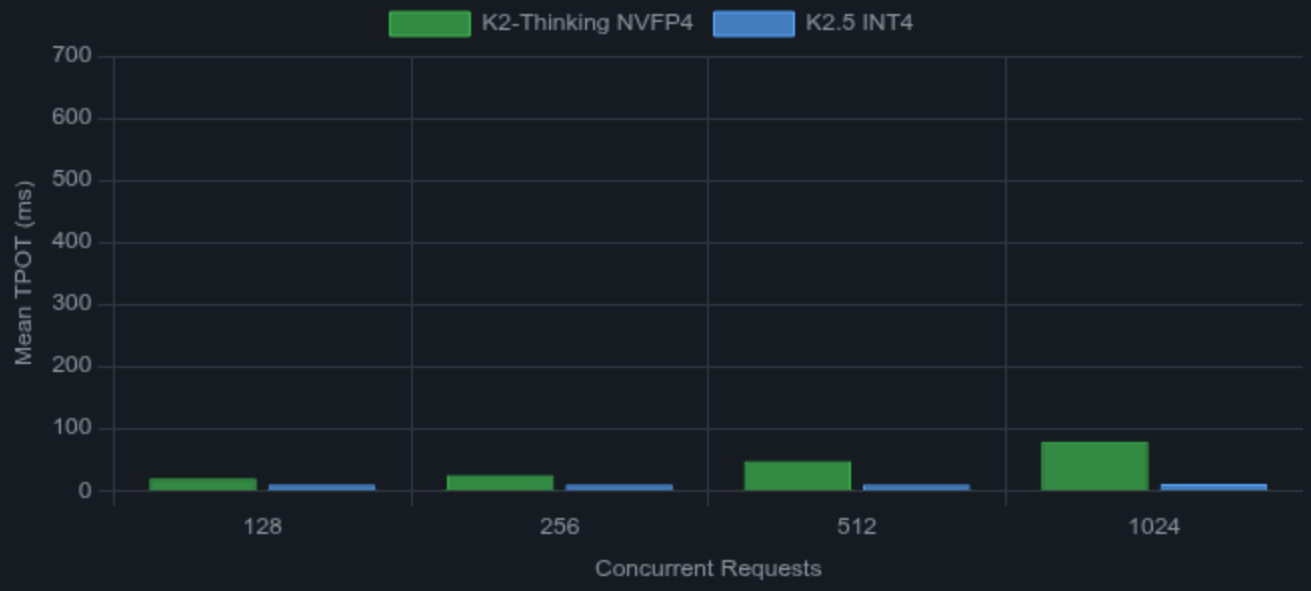
K2.5 INT4: Throughput vs Concurrency



Throughput Comparison (Total tok/s)



Latency Comparison (TPOT ms)



Detailed Benchmark Results

K2-Thinking NVFP4 (256 input, 100 output tokens)

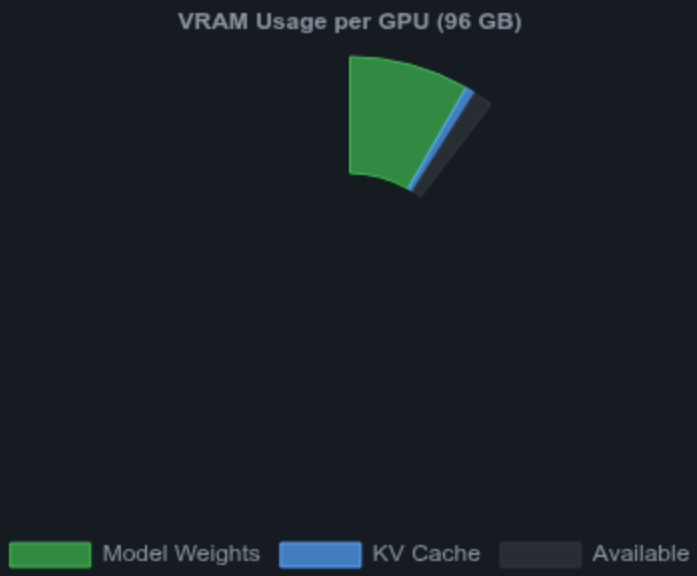
CONCURRENCY	OUTPUT TOK/S	TOTAL TOK/S	PEAK TOK/S	MEAN TTFT	MEAN TPOT
128	524	1,795	1,197	1,324 ms	160 ms
256	867	3,039	1,937	827 ms	207 ms
512	1,192	4,345	3,128	1,954 ms	401 ms
1024	1,610	5,655	4,941	3,560 ms	687 ms
2048	**1,650**	**5,816**	3,507	12,406 ms	576 ms

K2.5 INT4 (256 input, 100 output tokens)

CONCURRENCY	OUTPUT TOK/S	TOTAL TOK/S	PEAK TOK/S	MEAN TTFT	MEAN TPOT
32	171	712	323	363 ms	96 ms
64	214	864	357	2,026 ms	84 ms
128	258	884	423	8,330 ms	85 ms
256	267	935	449	19,010 ms	84 ms
500	270	985	439	38,647 ms	88 ms
1000	**281**	**985**	419	85,590 ms	91 ms

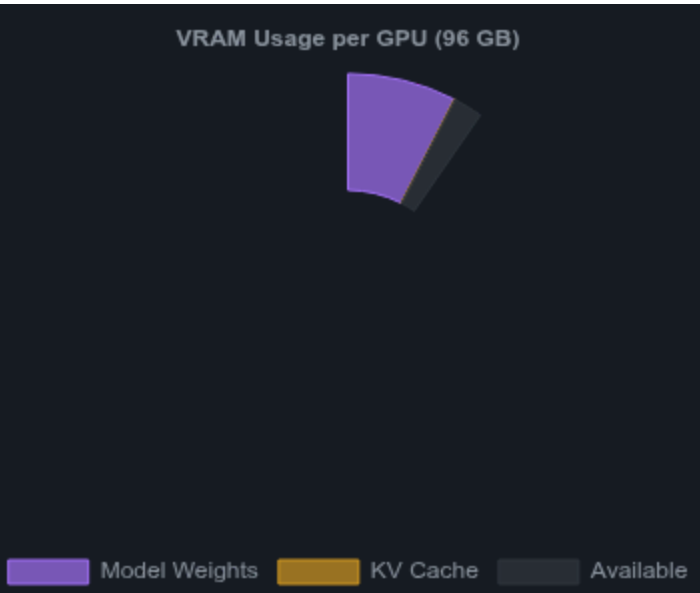
Memory Analysis

K2-Thinking NVFP4



Model Weights	73.87 GB/GPU
KV Cache	~6 GB/GPU
KV Cache Capacity	**180,191 tokens**
Available after load	12.24 GB/GPU

K2.5 INT4



Model Weights	72.33 GB/GPU
KV Cache	~0.3 GB/GPU
KV Cache Capacity	**4,578 tokens**
Available after load	19.71 GB/GPU

K2.5 INT4 has BF16 attention weights (not quantized), consuming more memory and leaving less for KV cache.

Key Findings

Why K2-Thinking Outperforms K2.5

- **40x larger KV cache** (180K vs 4.5K tokens)
- NVFP4 quantization is more memory-efficient
- cuDNN FP4 GEMM works well on SM120

- Can sustain high concurrency without queuing

SM120 vs SM100 Differences

- **Shared memory:** 100KB (SM120) vs 228KB (SM100)
- Consumer Blackwell requires kernel modifications
- TMA block layouts don't work on SM120
- Persistent kernels must be disabled

Deployment Attempts Timeline

ATTEMPT	MODEL	CONFIGURATION	RESULT
1	K2.5 INT4	SGLang Main	FAILED - Shared memory (106KB > 101KB)
2	K2.5 INT4	SGLang PR #16975	FAILED - Model class incompatibility
3	K2-Thinking NVFP4	SGLang Main	PARTIAL - Short prompts only
4	K2-Thinking NVFP4	SGLang PR #16975	PARTIAL - PR fixes don't cover attention
5	**K2-Thinking NVFP4**	**Main + SM120 Patch**	SUCCESS - 5,816 tok/s peak

ATTEMPT	MODEL	CONFIGURATION	RESULT
6	**K2.5 INT4**	**Main + SM120 Patch**	SUCCESS - 985 tok/s peak

Recommendations

For Production Use

- **Use K2-Thinking NVFP4** for highest throughput
- Apply the `extend_attention.py` SM120 patch
- Target 512-1024 concurrent requests for optimal throughput
- Monitor KV cache utilization

For Upstream Contribution

- Submit SM120 attention fix to SGLang
- Combine with PR #16975 for MoE fixes
- Add SM120 detection to kernel autotuning
- Generate optimized MoE configs for E=384

Generated: January 29, 2026 | Hardware: 8x NVIDIA RTX PRO 6000 Blackwell (SM120)

Patch: `artifacts/patches/sm120-extend-attention.patch`