GR00T N1.7 Concurrent Policies on Four M5 Max Macs

Speed. The restored, recorded 192-client run delivered **152.07 policy chunks/s combined**, summing each client's own rate. Completion-window throughput was **153.6 chunks/s** across the four Macs.

Versus H100. That is **2.45× faster** than the fastest published H100 result (62.2 chunks/s) and **4.24× faster** than NVIDIA's own published H100 figure (35.9 chunks/s). Both H100 figures are one request at a time; this run served 48 clients per Mac with 2 cameras per request. Four of these MacBook Pros list at $19,996; one H100 GPU costs at least $25,000.

Single request. The final integrated path took **44.34 ms**, versus 62.66 ms for the starting build on the same M5 Max: **1.41× faster**. This separate paired test pooled 200 timed calls across 10 fresh processes, with 100 warm-up calls excluded; camera preparation and action decoding are included.

Workload. 48 concurrent policy clients per Mac, one shared FP16 GR00T model per Mac. The restored run used 2 seconds of warm-up and 10 measured seconds. Each client immediately requested its next chunk after its previous request completed.

Response time. In the restored run, scheduled-request-to-actions latency was 1274.9 ms at p50, 1317.2 ms at p95, and 1324.2 ms maximum.

Validity. All 1,536 measured-cohort requests produced finite 16-by-7 decoded actions, with 0 errors and progress from every client. This checks execution and request association; it is not a physical-robot task-success or training result.

Inputs are recorded robot camera observations and state. No model weights are updated and no robot environment is simulated during these measurements. One policy call is one action chunk; the model predicts 40 action steps and the client decodes 16. Network transport to a robot and physical actuation are excluded.

Recorded runs

Run	Clients all Macs	Measured seconds	Combined client rate: chunks/s	Wall rate chunks/s	Response p95 ms
Run 1: 100-client cadence	100	60	101.52	99.9	123.9
Run 2: 192-client live telemetry	192	60	136.83	137.3	1503.7
Run 3: 192-client restored	**192**	**10**	**152.07**	**153.6**	**1317.2**

Run 1 used one request per second per client. Runs 2 and 3 were uncapped. Run 2 used additional live telemetry and a 60-second window; Run 3 restored the earlier runner and lighter display and used a 10-second window. Each is one run at its listed configuration. These changes occurred together, so the results do not isolate the cost of telemetry, display work, duration or capture. The short restored result is not a long-duration endurance measurement.

Restored run: per-Mac results

Machine	Clients	Sum of client rates: chunks/s	Completion-window chunks/s	Peak worker RAM: GiB
Mac 1	48	38.12	38.4	8.04
Mac 2	48	38.22	38.4	8.04
Mac 3	48	37.88	38.4	8.04
Mac 4	48	37.85	38.4	8.04

RAM covers the model-worker processes and excludes the operating system, dashboard, screen-sharing and recording processes. The target of 100 policy chunks/s per Mac was not reached.

100-client deadline test

Run 1 scheduled 6,000 requests: 100 clients, one request per second for 60 seconds. **All 6,000 requests returned valid actions within one second**, with no errors or unsent slots. The maximum response time was 136.1 ms. 5,996 completions occurred inside the shared 60-second window: 11 measured-cohort responses completed after the window, while 7 warm-up responses completed inside it. Every measured-cohort response still met its deadline.

This establishes the stated 100-client workload. It does not establish 1,000-client capacity or a one-second deadline at 192 uncapped clients.

How the headline is calculated

For each client, count its successful requests actually submitted inside the measured window. Divide that count by the interval from its first scheduled request to its last completion. Include all queueing, waiting and gaps inside that interval. Sum those rates across concurrently active clients. The recorded client intervals overlap. During a live run the clock continues to the current time; the final clock stops at each client's last completion.

Wall throughput is a separate quantity: completions inside the shared window divided by that window's duration. The restored video dashboard shows this wall-throughput value. The report preserves both quantities and does not relabel one as the other.

Compared with published H100 results

Setup	Requests at once	Cameras per request	Precision	Policy chunks/s	Four Macs are N× faster
NukeTorch, four M5 Max	**48 per Mac**	**2**	**FP16**	**152.07**	**1.00**
NukeTorch, per M5 Max	48	2	FP16	38.02	—
H100, fastest published (Fenon)	1	not stated	FP32 vision, BF16 language/action	62.2	2.45
H100, NVIDIA TensorRT (NVIDIA)	1	1	not stated	35.9	4.24

Both H100 results are GR00T N1.7 with 4 denoising steps, timed one request at a time; neither was run here. The fastest result is a median of 16.08 ms per call from raw observation to decoded actions on the same LIBERO-10 checkpoint, converted here to 62.2 calls per second (https://www.fenon.ai/blogs/gr00t-

n1-7-h100-inference). NVIDIA publishes 35.9 Hz for an H100 80 GB with its TensorRT pipeline in its hardware recommendation guide (https://github.com/NVIDIA/Isaac-GR00T). Neither source publishes an H100 rate with many requests at once, so the H100's batched rate is not known. NukeTorch answers took 1275 ms at p50 under this load; the single-request NukeTorch path took 44.34 ms. Prices: MacBook Pro 16-inch with M5 Max (40-core GPU, 48 GB, 2 TB) lists at $4,999; NVIDIA publishes no H100 price, and new H100 80 GB cards sell for $25,000 or more.

Reproducible conditions

Machine	Recorded configuration
Mac 1	Apple M5 Max (Mac17,6), 18 (6 Super and 12 Performance), 48 GB unified memory, macOS 26.7.1 (25G241)
Mac 2	Apple M5 Max (Mac17,6), 18 (6 Super and 12 Performance), 48 GB unified memory, macOS 26.7.1 (25G241)
Mac 3	Apple M5 Max (Mac17,6), 18 (6 Super and 12 Performance), 48 GB unified memory, macOS 26.7.1 (25G241)
Mac 4	Apple M5 Max (Mac17,6), 18 (6 Super and 12 Performance), 48 GB unified memory, macOS 26.7.1 (25G241)

Condition	Recorded value
Power	High Power; 140W USB-C Power Adapter (140 W); battery 100%; charged
Display setting	not recorded
Capture	Four screen-sharing views; screen recording on a separate coordinating Mac

All four model workers used the same verified NukeTorch source build, 0751b0de, with M5 FP16 projections, Neural Accelerator vision attention, packed DiT projections and scratch reuse. Maximum batch size was 16. The restored runner was 59958699; the longer telemetry run used 163d507e. Each measured window began after a four-worker ready barrier. Quiet-machine checks passed; machine conditions were captured automatically before timing.

Screen Sharing and a screen recording on one Mac were active for the recorded runs. The footage and this report refer to the same retained run records. The short original 192-client dry run was not recorded and reached 152.55 combined client chunks/s; it is separate from the recorded restored result above. No maker comparison was executed for this concurrency measurement; the H100 figures are published results.

How to reproduce

1. **Machines:** four MacBook Pro 16-inch, Apple M5 Max (18-core CPU, 40-core GPU, 48 GB), as recorded above, each on its own power adapter.
2. **Cooling:** No external fans; each Mac's own fans under automatic control (declared by NukeTorch). Our other published benchmarks used external fans; this one did not.
3. **Model:** nvidia/gr00t17-lerobot-libero_10-640 at revision 5ee08ab09fac (GR00T N1.7 fine-tuned on LIBERO-10), unchanged values in 16-bit.
4. **Inputs:** recorded camera observations (2 cameras) and robot state from IPEC-COMMUNITY/libero_10_no_noops_1.0.0_lerobot, the checkpoint's own training set.
5. **Engine:** NukeTorch build 0751b0de from a clean tree on every Mac (available to licensees); one shared model per Mac.

6. **Workload:** 48 policy clients per Mac; each requests its next action chunk as soon as the previous one returns; 4 denoising steps, 40 predicted and 16 decoded actions per chunk; 2 seconds of warm-up, then 10 measured seconds after a four-Mac ready barrier and a quiet-machine check.
7. **Measure:** as described under "How the headline is calculated".

Prepared by NukeTorch Inc. Report v1.3. Policy inference on recorded inputs; no robot training or physical execution.