Power and Thermal Management

for GPU-Based LLM Inference

February 2026

Table of Contents

1. Introduction

As LLM inference workloads scale to occupy entire racks of GPUs, power consumption and heat dissipation become first-order engineering constraints. A single high-end data center GPU can draw 300–700 W under load, and a system of many smaller GPUs may collectively draw similar or greater power while also presenting more complex thermal challenges due to density and airflow considerations. Understanding where power is consumed, how it translates to heat, and what levers are available to manage both is essential for building sustainable and cost-effective inference infrastructure.

This report covers the physics and engineering of power and thermal management for GPU inference systems, from chip-level power domains through system-level cooling strategies, with a focus on practical techniques for multi-GPU small-chip deployments.

2. GPU Power Anatomy

2.1 Where Power Goes

GPU power consumption is distributed across several domains. The compute cores (CUDA cores, tensor cores) consume power proportional to their utilization and clock frequency. The memory subsystem—both the GDDR6/HBM modules and the memory controllers—draws significant power during the bandwidth-intensive decode phase. The interconnect fabric (NVLink, PCIe PHYs) adds a fixed base load plus a data-dependent component. Finally, voltage regulators, fans, and auxiliary circuits contribute a baseline that is present even at idle.

For LLM inference specifically, the decode phase tends to be memory-bandwidth-bound, meaning memory power dominates over compute power. The prefill phase is compute-bound and will draw peak power from the compute cores. This phase-dependent power profile creates variable thermal loads that cooling systems must accommodate.

2.2 Power Scaling with Chip Count

A system of many small GPUs does not simply consume more total power than fewer large GPUs—the relationship is more nuanced. Each additional chip adds its own baseline idle power, voltage regulator losses, and interconnect power. However, smaller chips often operate at lower voltages and frequencies, and their per-chip power draw may be 75–150 W compared to 300–700 W for a large accelerator. The total system power depends on how many chips are needed, but the per-chip thermal density is typically lower, which can simplify cooling at the chip level while complicating it at the system level due to the sheer number of heat sources.

3. Thermal Fundamentals

3.1 Heat Generation and Dissipation

Every watt of electrical power consumed by a GPU is ultimately converted to heat. The thermal design power (TDP) or total graphics power (TGP) rating indicates the sustained heat output the

cooling system must handle. For a rack containing 16 small GPUs at 150 W each, the total thermal load is 2.4 kW from GPUs alone—plus CPU, memory, storage, networking, and power supply losses, which can bring the total to 4–6 kW per server.

3.2 Thermal Throttling

When a GPU's junction temperature exceeds its thermal limit (typically 83–95°C depending on the chip), the firmware reduces clock frequencies to prevent damage. This thermal throttling directly impacts inference latency and throughput. In a distributed inference setup, throttling on even one GPU can bottleneck the entire pipeline, since all chips must synchronize at each layer boundary. Maintaining temperatures well below throttle thresholds is therefore critical for consistent performance.

3.3 Ambient Temperature and Altitude

Cooling effectiveness depends on the temperature differential between the GPU and the ambient air (or coolant). Data centers typically maintain inlet air at 18–27°C. At higher ambient temperatures, fans must spin faster (consuming more power themselves) or liquid cooling capacity must increase. At higher altitudes, air density decreases, reducing the heat-carrying capacity of air cooling by approximately 3–4% per 300 meters above sea level.

4. Cooling Strategies

4.1 Air Cooling

Air cooling remains the most common approach for small-GPU systems, particularly those based on consumer or workstation-grade cards. Effective air cooling for multi-GPU systems requires careful attention to airflow path design. Hot-aisle/cold-aisle containment prevents recirculation of exhaust air. GPUs should be arranged so that each card receives fresh cool air rather than preheated exhaust from an adjacent card. Blower-style coolers that exhaust heat directly out of the chassis are preferable to open-air coolers in dense multi-GPU configurations, as the latter can create internal hot spots.

4.2 Liquid Cooling

For dense deployments, direct-to-chip liquid cooling (using cold plates attached to the GPU die) offers 2–3x better thermal conductivity than air. This enables higher sustained power without throttling and allows for much denser packing of GPUs. Liquid cooling systems typically use a closed loop of water or water-glycol mixture circulated through a facility's chilled water system or a dedicated coolant distribution unit (CDU). The upfront cost is higher, but operational savings from reduced fan power and improved GPU utilization often justify it.

4.3 Immersion Cooling

Single-phase and two-phase immersion cooling submerge the entire server in a dielectric fluid. This eliminates fans entirely, reduces noise, and can handle thermal densities exceeding 100 kW per rack. While immersion is still relatively niche, it is increasingly adopted for AI inference

workloads where density and energy efficiency are paramount. The main drawbacks are the complexity of maintenance (hardware must be extracted from fluid for servicing) and the cost of the dielectric fluid itself.

5. Power Management Techniques

5.1 Power Capping and Frequency Scaling

NVIDIA GPUs support power capping via nvidia-smi, where a maximum power draw can be set below the card's rated TDP. For inference workloads, reducing power limits by 10–20% often has minimal impact on throughput (perhaps 5–10% reduction) because the decode phase is memory-bandwidth-bound and does not fully utilize the compute at maximum clocks. This nonlinear relationship between power and performance means that power capping is one of the most effective efficiency levers available.

Dynamic frequency scaling (DVFS) automatically adjusts GPU clock speeds based on workload. During the memory-bound decode phase, the GPU can downclock its compute cores without affecting token generation speed, saving power and reducing heat. During the compute-bound prefill phase, clocks ramp up to handle the parallel workload.

5.2 Quantization as a Power Strategy

Quantization reduces not only memory usage but also power consumption. Moving from FP16 to INT8 inference roughly halves the energy per operation for matrix multiplications, while INT4 quarters it. Fewer bits also mean less data moved from memory, reducing the power drawn by the memory subsystem. For a multi-GPU system where memory bandwidth is the bottleneck, quantization can reduce total system power by 30–50% while maintaining acceptable model quality.

5.3 Batch-Aware Power Management

Inference workloads are inherently bursty. Between requests or during low-traffic periods, GPUs may sit idle but still draw significant baseline power (30–50% of TDP). Strategies to address this include aggressive clock gating (reducing frequency to minimum during idle intervals), request batching with idle timeouts (consolidating work onto fewer GPUs and powering down unused ones), and workload-aware scheduling that routes requests to already-active GPUs before waking idle ones.

6. System-Level Considerations

6.1 Power Delivery Infrastructure

A multi-GPU inference system's power requirements must be considered end-to-end. Power supply units (PSUs) have their own efficiency curves—typically 90–96% efficient under 50–80% load, dropping at very low or very high utilization. Oversizing the PSU wastes money; undersizing causes shutdowns under peak load. For a system with 8 GPUs at 150 W each, the

GPU load alone is 1.2 kW, and the total system draw (including CPU, RAM, storage, fans) will be 1.8–2.2 kW, requiring a PSU rated for at least 2.4 kW with headroom.

6.2 Monitoring and Observability

Effective power and thermal management requires continuous monitoring. NVIDIA's DCGM (Data Center GPU Manager) and nvidia-smi provide per-GPU telemetry including power draw, temperature, clock speeds, memory utilization, and throttle events. In a distributed inference system, this telemetry should be aggregated (e.g., into Prometheus/Grafana) and correlated with inference metrics (latency, throughput) to identify thermal bottlenecks. Alerts on approaching throttle temperatures or power limits enable proactive intervention.

6.3 Energy Efficiency Metrics

The standard metric for inference efficiency is tokens per joule (or equivalently, tokens per watt-second). This captures both the throughput and the power cost. For comparing systems, tokens per dollar (incorporating hardware amortization and electricity cost) provides a more complete picture. A system of small GPUs running quantized models with power capping can often achieve competitive tokens-per-joule ratios compared to large-GPU systems, particularly for decode-dominated workloads.

Configuration	Typical GPU Power	System Power	Tokens/Sec	Tokens/Joule
1x H100 (FP16)	700 W	~1000 W	~80	~0.08
8x RTX 4090 (INT4)	8 x 300 W	~3000 W	~120	~0.04
8x RTX 4090 (INT4, capped)	8 x 200 W	~2100 W	~110	~0.05

Table: Approximate comparison for 70B model decode throughput. Values are illustrative and workload-dependent.

7. Conclusion

Power and thermal management are not afterthoughts—they are fundamental constraints that shape hardware selection, system architecture, and operational cost. For distributed inference on small GPUs, the key takeaways are: apply power capping aggressively (the performance-power curve is nonlinear and favorable), use quantization as both a memory and energy optimization, ensure adequate cooling with attention to airflow in dense multi-card configurations, and monitor continuously to catch thermal throttling before it impacts user-facing latency. As inference workloads scale, the practitioners who treat power as a first-class design variable will build the most cost-effective and sustainable systems.