GUARD: SCALABLE STRAGGLER DETECTION AND NODE HEALTH MANAGEMENT

FOR LARGE-SCALE TRAINING

Guanliang Liu, Abhinandan Patni, Congzhu Lin, Zoe Zeng, Jack Wittmayer, Josh Wu , Ashvin Nihalani, Binxuan Huang, Yinghong Liu, Rory Na, Anthony Ko , Alexander Zhipa, Cong Cheng, Mi Sun, Vijay Rajakumar, Rejith George Joseph, Parthasarathy Govindarajen

MLSys, 2026, Bellevue, WA

amazon

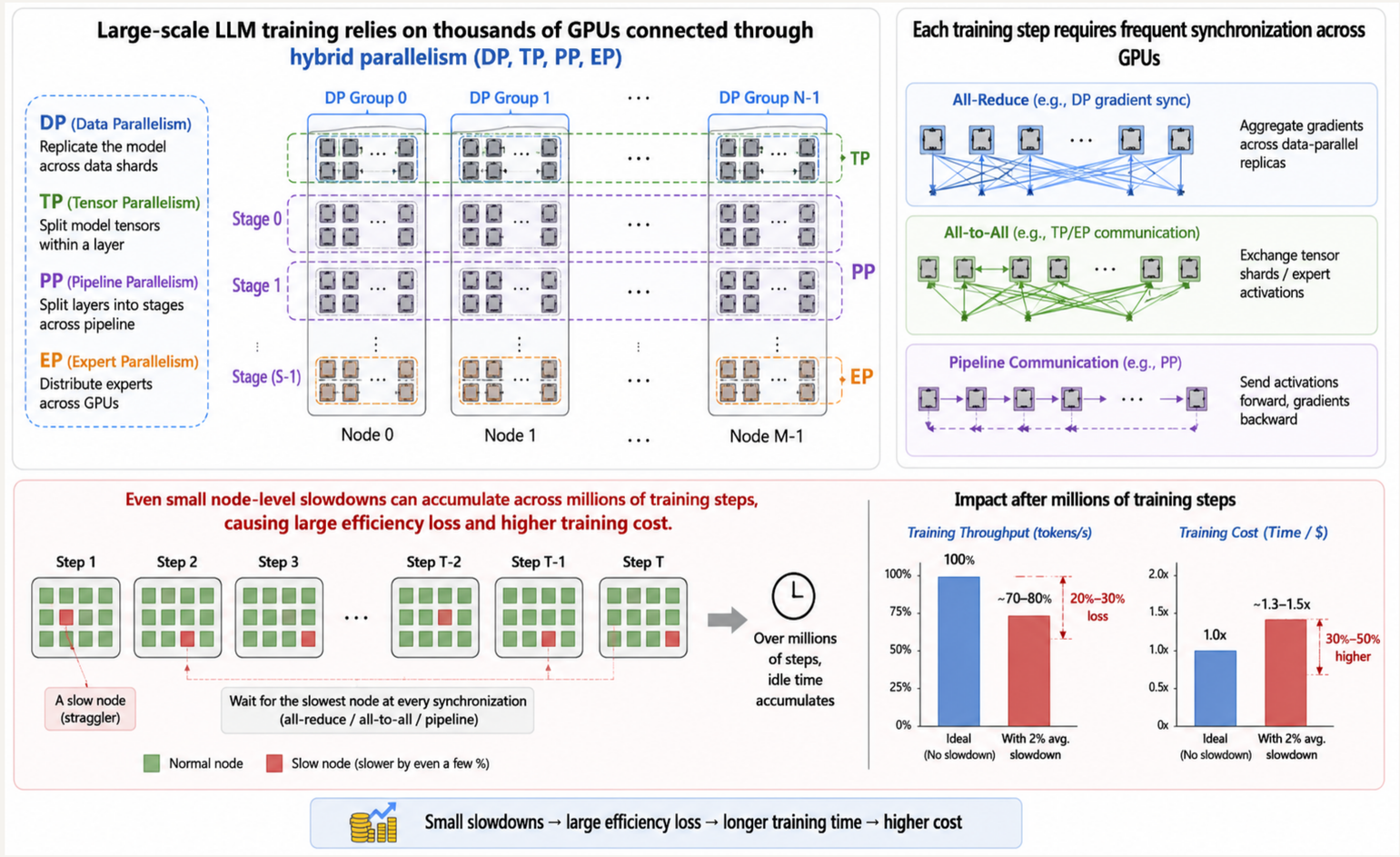
Agenda

- **Background, Motivation & Problem Statement**
  Why grey nodes and stragglers matter in large-scale LLM training.
- **Training Slowdown Analysis**
  CPU configuration, communication degradation, and GPU performance bottlenecks.
- **GUARD System Overview**
  A closed-loop framework combining online monitoring and offline node sweep.
- **Online Monitoring: Metrics & Detection Strategy**
  Key health signals, peer-based anomaly detection, and tiered mitigation policy.
- **Offline Node Sweep & Node Health Management**
  Single-node validation, multi-node communication sweep, quarantine, and recovery.
- **Evaluation Results & Key Takeaways**
  Improvements in MFU, training step time, variance, and MTTF.

Background, Motivation & Problem Statement

- Large-scale LLM training uses hybrid parallelism (DP, TP, PP, and EP) across thousands of GPUs.
- Training throughput is often limited by the slowest participant in each synchronization group. In MoE and hybrid-parallel systems, dynamic workload and communication patterns make such performance imbalance difficult to detect.
- Even minor node-level slowdowns can accumulate over millions of training steps, leading to significant efficiency loss and increased training cost.

This motivates the need to identify hidden performance-degraded yet functional nodes, known as grey nodes.



Grey Nodes: Grey Nodes: Functional but Performance-Degraded

- Grey nodes pass standard functional checks but deliver degraded runtime performance.
- They usually do not crash the job, so the slowdown can silently persist for multi-week training runs.
- Common root causes include thermal throttling, degraded NICs, CPU bottlenecks, NVLink instability, and power limits.
- Traditional NCCL tests and GPU burn-in checks mainly validate correctness, not sustained workload performance.
- The impact is amplified in synchronization-heavy workloads because the slowest participant gates progress.
- The core problem is to detect performance-degraded nodes early without interrupting healthy training jobs.

Training Slowdown Analysis

- Training slowdowns often appear as persistent step-time inflation or intermittent step-time spikes. In distributed training, small node-level degradations are amplified by repeated synchronization barriers.
- CPU-side bottlenecks can delay data loading, checkpointing, and communication coordination.
- Communication bottlenecks can be hidden by NCCL rerouting, which preserves correctness but reduces bandwidth.
- GPU-side degradation can reduce sustained compute throughput without triggering explicit hardware failures.

Guard turns grey-node handling from reactive debugging into proactive cluster management

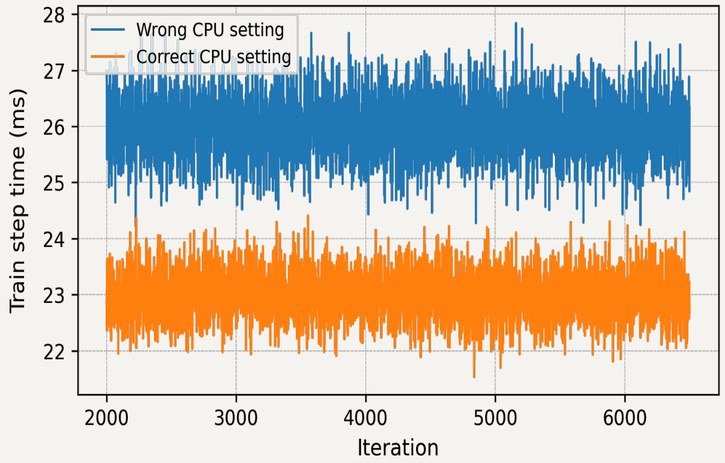


Fig. 1 Different CPU settings' speed

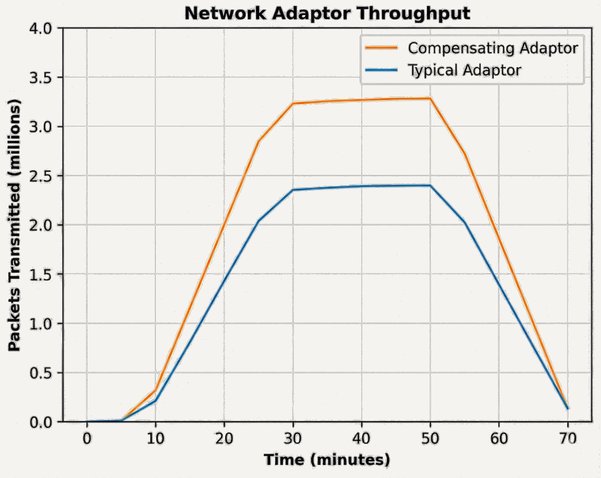


Fig. 2 Abnormal network throughput

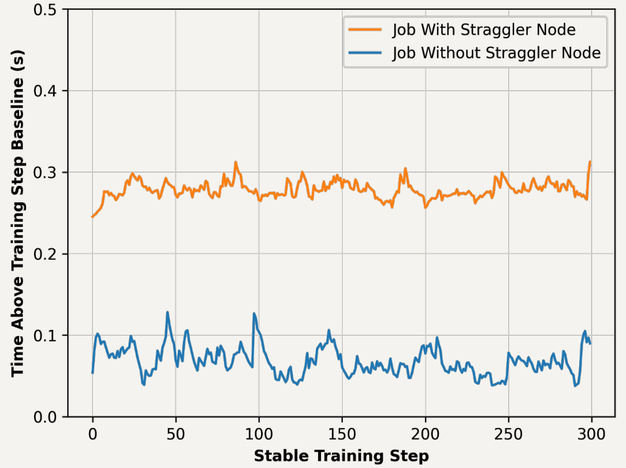


Fig.3 Train slowdown due to network

GUARD System Overview

- GUARD is a closed-loop system for scalable straggler detection and node health management.
- The online monitoring layer observes hardware, network, and training performance signals during real jobs. Suspicious nodes are removed from the healthy node pool and sent to offline validation.
- The offline node sweep reproduces realistic compute and communication patterns to confirm node health. After repair and validation, healthy nodes are returned to the good node pool; failed nodes remain quarantined or are replaced.

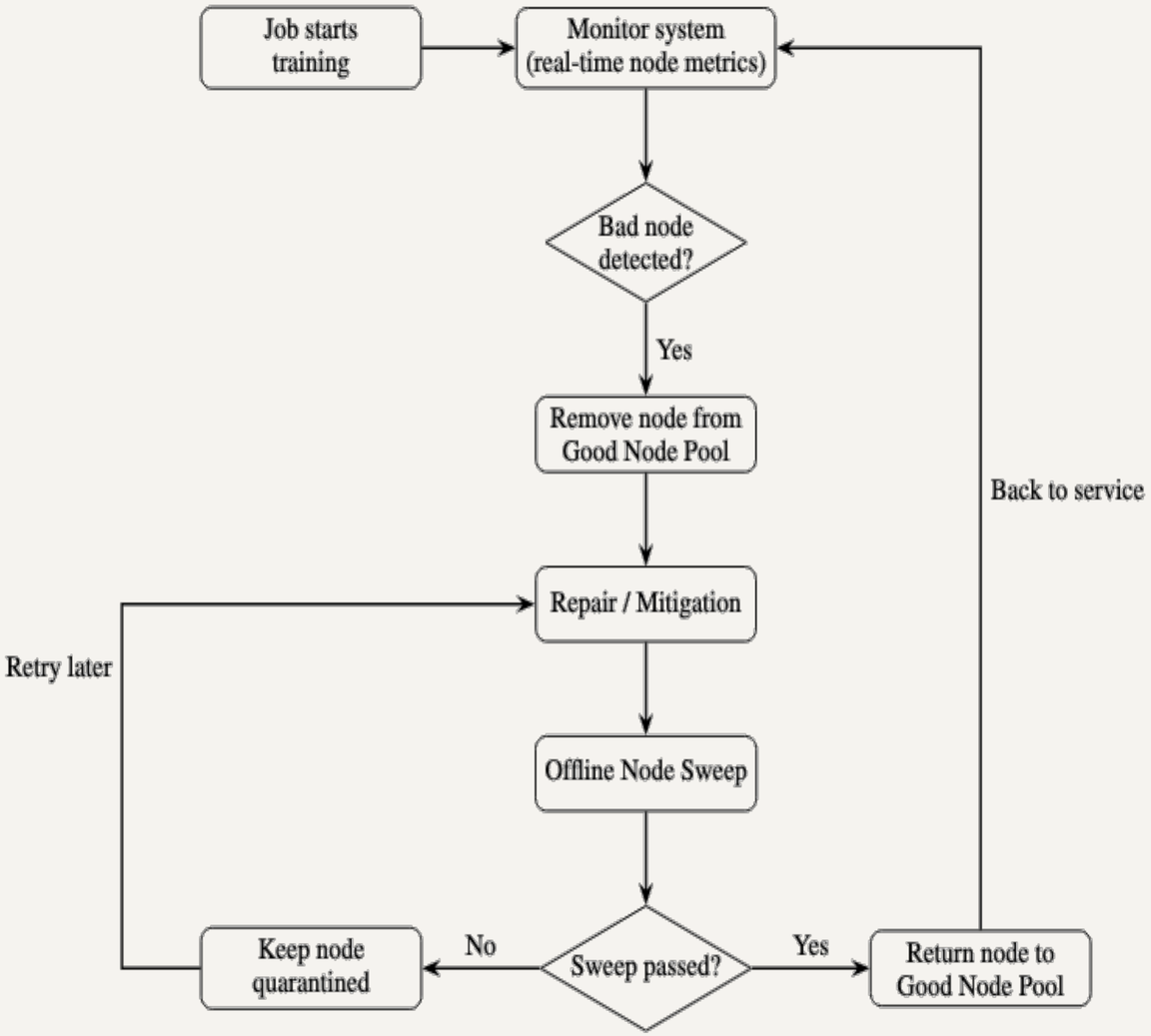
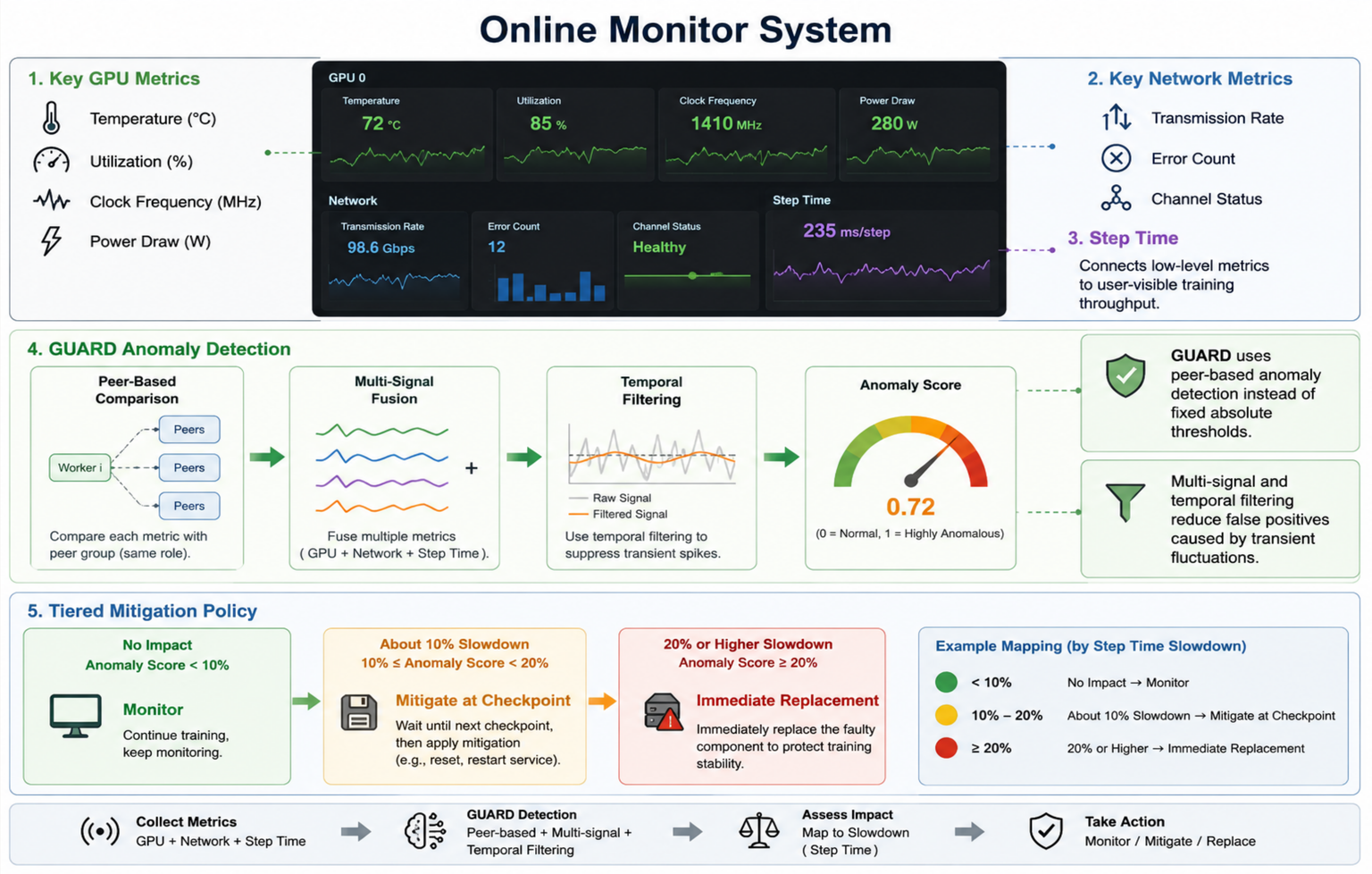


Fig. 4 Guard system workflow

Online Monitoring Metrics

- GPU metrics: temperature, utilization, clock frequency, and power draw.
- Network metrics: transmission rate, error count, and channel status.
- Step time links low-level metrics to training throughput.
- GUARD uses peer-based anomaly detection instead of fixed thresholds.
- Multi-signal and temporal filtering reduce transient false positives.
- Tiered mitigation: monitor (<10%), checkpoint mitigation (~10%), immediate replacement (≥20%).



Node Management Workflow

Quarantine unhealthy nodes
Unhealthy nodes are marked as grey nodes and removed from normal scheduling.

Check GPU / network errors
The Guard monitors whether the node continues to emit GPU or network errors.

Apply recovery actions
If errors exist, the system first reboots the node and redeploys drivers.
If errors persist, it further reprovisions or redeploys the node.

Return or replace nodes
Recovered nodes are returned for the next node sweep, while persistently faulty nodes are terminated or replaced.

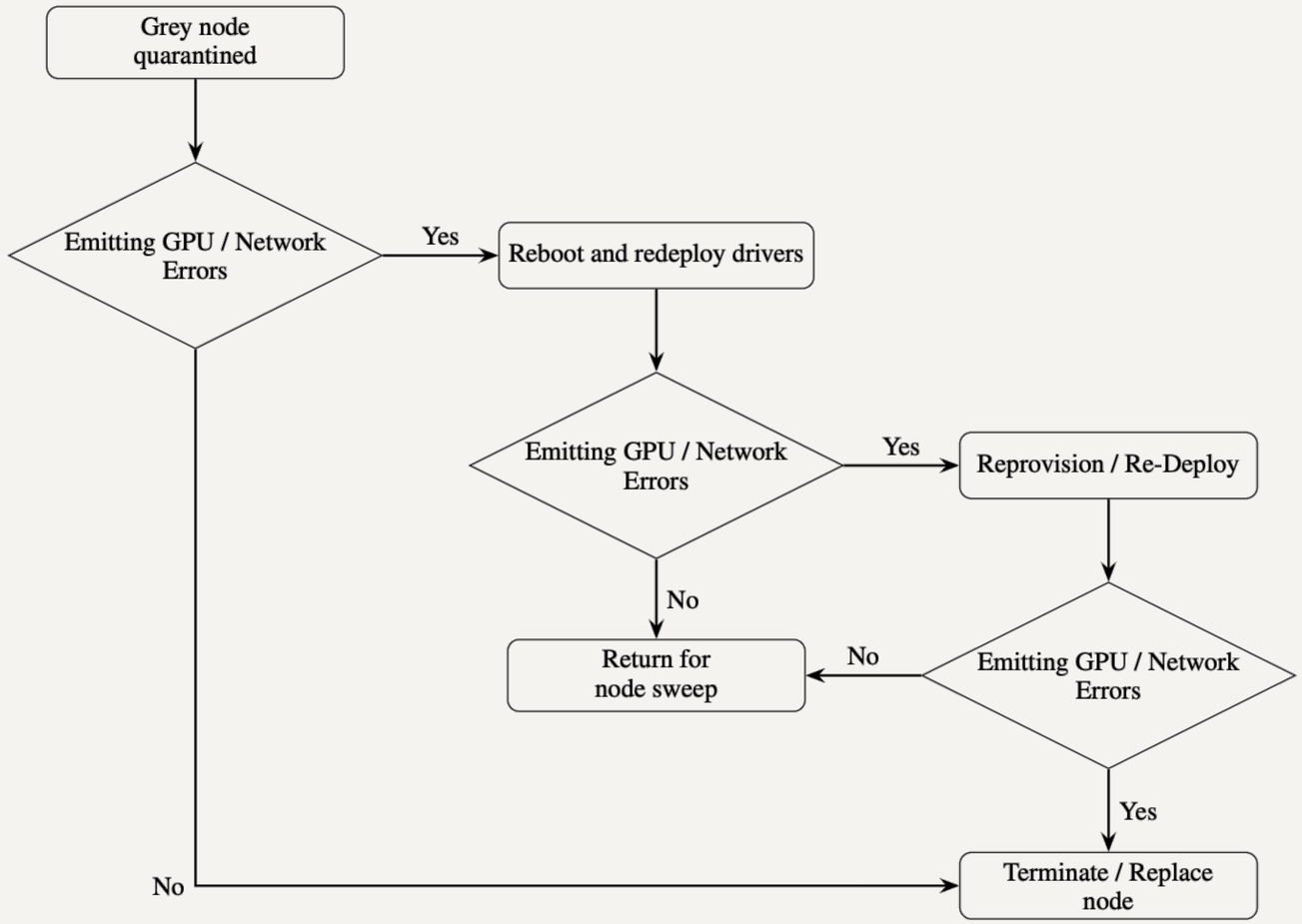


Fig. 5 Node management workflow

Offline Node Sweep

- Single-node sweep validates intra-node compute throughput and GPU-to-GPU communication symmetry. It exposes local issues such as thermal throttling, uneven GPU performance, and degraded NVLink paths.
- Multi-node sweep validates inter-node communication efficiency under controlled collective workloads. A 2-node sweep is often sufficient to detect degraded bandwidth, routing asymmetry, and faulty network paths.

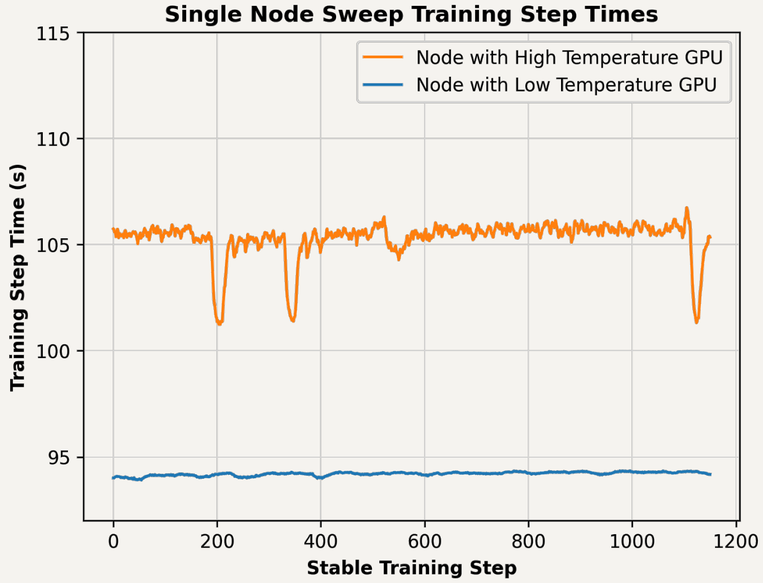


Fig. 6 Single node sweep result

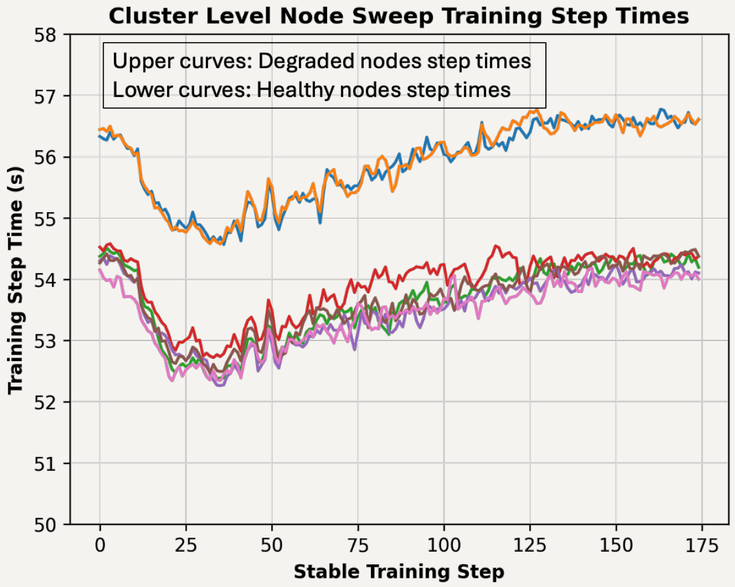


Fig.7 Multi-node node sweep result

Experimental Results

- GUARD improves mean FLOPs utilization by up to 1.7× on production-scale training workloads. Average training step time is reduced from 17 seconds to 10 seconds after applying health monitoring and node selection.
- Run-to-run variance is reduced from 20% to 1%, improving performance predictability.
- Key takeaway: combining online monitoring with offline validation turns grey-node handling from reactive debugging into proactive operations.

Fig. 9 Train step time result

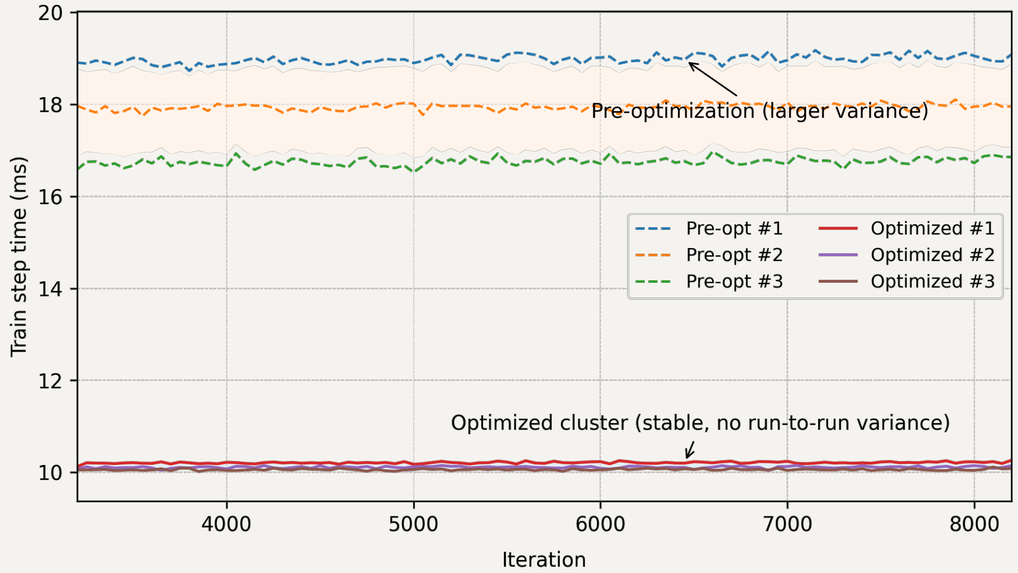


Fig. 10 Run-to-run variance result

Experimental Results

- Ablation results show MFU improves from 5% with NCCL/burn-in only to 17% with enhanced sweep and online monitoring.
- Average MTTF improves to 16.7 hours, while average human intervention interval decreases to 0.5 hours.

Table 1. Ablation Result

Method	Avg. MTTF (h)	Avg. Human Interval (h)	Avg. MFU	Detects HW Degradation
NCCL / Burnin Tests Only	6.6	5.6	5%	No
NCCL / Burnin + Node Sweep	8.1	2.0	10%	No
NCCL / Burnin + Online Monitoring + Node Sweep	9.2	1.2	14%	Yes
NCCL / Burnin + Online Monitoring + Enhanced Node Sweep	**16.7**	**0.5**	**17%**	Yes

Conclusion

- Large-scale LLM training requires strong node-level performance consistency.
- Grey nodes silently degrade throughput while passing standard functional checks.
- GUARD detects, quarantines, and recovers performance-degraded nodes automatically.
- GUARD improves MFU, reduces MTTR, and increases MTTF.
- This leads to more reliable and efficient large-scale training.

Thank You