CLOCKWORK.IO

Unflappable Fabrics: Ending Link-Induced Chaos in GPU Clusters

Artificial intelligence (AI) is rapidly moving from specialized research to mainstream applications reshaping entire industries. Modern AI workloads demand unprecedented scale, performance, and reliability from GPU clusters. Yet as clusters grow to thousands of GPUs and tens of thousands of links, one bottleneck emerges again and again: network fragility. Network failures, link flaps, and network misconfigurations erode training efficiency, forcing job restarts and burning valuable GPU hours. This paper examines the true cost of network failures and introduces a new architectural approach - rooted in real-time telemetry and software-driven resilience - to ensure AI workloads keep moving, even when the network doesn't.

Network Failures Contribute to Poor GPU Cluster Utilization

AI training clusters represent a formidable infrastructure expenditure, and every lost hour amounts to tens of thousands of dollars. This is not a theoretical concern since actual FLOPs (Floating Point Operations per Second) utilization during large-scale AI training often ranges between 15% and 40%.

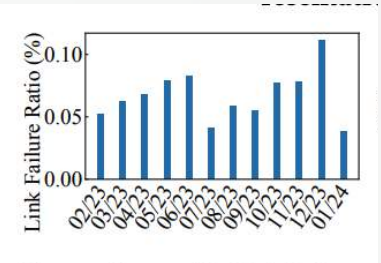
One of the dominant culprits is communication overhead, particularly in distributed training. Large AI models - especially transformer-based architectures - require frequent synchronization across GPUs for operations like gradient aggregation. As GPU counts grow into the thousands, any delays in synchronization due to network bottlenecks cause thousands of GPUs to be idle, waiting for the stragglers. Three network pathologies consistently outweigh all others in the amount of GPU time they waste:

1. **Cross-job congestion and contention.** Multi-tenant clusters run dozens of jobs whose "elephant" collective flows collide on the same links; incast hotspots stall every rank that has to enter the next All-Reduce.
2. **Network failures.** Short-lived link flaps, optic resets, or bursty bit-error episodes act like micro-outages, often causing job crashes and restarts.
3. **Misconfigurations and bugs, including topology & placement mismatch.** Incorrect ECN/PFC thresholds, buggy switch images, or schedulers that scatter jobs across extra spine hops are examples of misconfigurations that chip away at utilization.

The rest of this document focuses on transient network failures, their impact on overall cluster GPU utilization, and the Clockwork FleetIQ platform approach to mitigating the adverse impact of such transient network failures.

Characterizing Network Failures

Modern AI training clusters push networking to extreme scale, and the raw component count alone makes "zero-fault" operation statistically impossible. For instance, NVIDIA's DGX SuperPOD reference architecture shows that a cluster of 8,192 GPUs needs 256 leaf switches, 256 spine switches, ~25,000 links and ~50 k optical transceiver modules. Alibaba reported[1] that nearly 60% of their large-scale training jobs experience slowness in production. Meta[2] reported 466 job restarts during their 54-day Llama 3 training, averaging 8.6 times per day.



Network-related disruptions in large-scale GPU clusters fall broadly into **three categories**: hardware failures, disruptive link flaps and self-healing link flaps. While each of these manifests differently, only the former two - hardware failures and disruptive flaps - result in job restarts.

1. **Hardware failures** stem from physical component degradation — such as failed optical transceivers or switch crashes. Based on Telcordia SR-332 specifications and production cluster data from Alibaba, NIC-to-leaf switch links fail at a rate of 0.68% per year, while ToR switches experience critical failures at 0.61% per year. In practical terms, this equates to **approximately one hard link failure per week** in a 1,000-GPU cluster, each of which can disrupt in-progress training jobs.
2. **Disruptive link flaps** are short-lived disruptions where a link temporarily drops and is unable to recover or resolve the loss of connectivity automatically. Experience from hyperscalers like Meta and Microsoft shows that a cluster of 8,000 GPUs may experience 5–20 link flaps per day and **~15% of all flaps are disruptive**, requiring manual intervention - such as cleaning or reseating optics - and can persist for tens of minutes. During this time, collective operations stall, causing timeouts and ultimately forcing the job to restart from its last checkpoint. **In a 1,000-GPU cluster, these disruptive flaps lead to 1–4 job-impacting events per week**.

Number of GPUs	Job restarts / year	Mean time to failure
1000 GPUs	100 - 250	35-87 hours
5000 GPUs	500 - 1250	7 - 18 hours
10,000 GPUs	1000 - 2500	3.5 - 9 hours
50,000 GPUs	5000 - 12,500	42 - 105 minutes

3. **Self-healing link flaps** are short-lived disruptions where a link temporarily drops and then recovers automatically, often within a few seconds. These are typically caused by transient bit errors or marginal signal integrity, and roughly 85% of link flap events self-resolve within the timeout limits of collective communication libraries like NCCL, meaning jobs continue unaffected. Short flaps don't crash a job because the NIC retransmits packets up to 7 times after the IB_TIMEOUT elapses, allowing the link to self-heal.

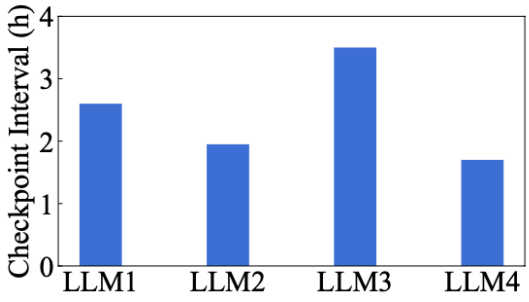
When viewed together, **hardware failures and disruptive flaps form the dominant source of job restarts** in large-scale AI training clusters. They contribute to an **aggregate annual rate of 0.1 to 0.25 disruptive outages per GPU**, with each event potentially causing the loss of hundreds to thousands of GPU-hours.

Translating Disruptive Network Failures and Flaps Into Lost GPU Hours

When a disruptive link failure causes an outage that lasts longer than the time-out limits of collective communication libraries, the training job needs to be restarted. The job must be restarted from the most recent consistent saved state, which is the last checkpoint. Consequently, the time spent training across all the involved GPUs since that last checkpoint, along with the time taken to recover, represents the GPU hours lost.

The estimation of GPU hours lost due to a job restart involves several interconnected parameters. These primarily include the frequency at which checkpoints are saved, the time required to recover from the most recent checkpoint, and the total number of GPUs actively participating in the training job.

- **Checkpointing frequency of 2-4 hours results in a loss of 1-2 hours per disruptive link failure**. The choice of checkpointing frequency involves a critical trade-off. More frequent checkpointing reduces the amount of computation that needs to be redone if a failure occurs, but it also introduces overhead because of the time taken to save the model's state to storage, potentially stalling the GPU computation. In their paper[3], Alibaba benchmarks typical checkpointing intervals for four production LLMs of between 2 and 4 hours, which seems to be a good representative checkpointing frequency. Given this checkpointing frequency, a disruptive link failure causes a loss of 1-2 hours (mid-point of the checkpoint interval) across all the GPUs allocated to the job.



- **Recovery time adds an additional 10-30 minutes of lost time per disruptive link failure**. Recovery from checkpoint involves retrieving the checkpoint data from the storage system and then loading the model parameters and optimizer states back onto the GPUs to resume the training process. While this also depends on the size of the model, the performance of the storage system and the network bandwidth, recovery time has been benchmarked in numerous studies to range between 10 and 30 minutes, when all the GPUs allocated to the job are unproductive.
- **Individual jobs have a wide range of GPUs allocated, that are idled by disruptive link failures.** GPU hours lost due to a restart is directly proportional to the number of GPUs that were allocated to the interrupted job, which is frequently less than the total number of GPUs available in the cluster, as multiple training jobs run concurrently. The number of GPUs allocated per job is a wide range - dozens of GPUs for fine-tuning and recommendation systems to several hundreds of GPUs for autonomous vehicle training or protein folding in drug discovery to thousands of GPUs in large language models. To estimate the impact of disruptive link failures in GPU clusters of hundreds to thousands of GPUs, we assume a range of 256-1000 GPUs allocated per impacted job.

	GPUs impacted	Checkpoint loss	Recovery time	GPU hours lost
Job 1	256	1 hour	10 mins	**299 hours**
Job 2	256	2 hours	30 mins	**640 hours**
Job 3	512	1 hour	10 mins	**597 hours**
Job 4	512	2 hours	30 mins	**1280 hours**
Job 5	1024	1 hour	10 mins	**1195 hours**
Job 6	1024	2 hours	30 mins	**2560 hours**

Putting the data together, the cumulative GPU hours lost due to each disruptive link failure ranges between ~500 GPU hours to ~1500 GPU hours. This depends on the three main factors described in this section - checkpointing frequency, time to restore from checkpoints and number of GPUs allocated to training jobs.

Business Impact Of Disruptive Network Failures

In clusters of 1,000 GPUs or more, network disruptions leading to job restarts can have significant and measurable business impact - in wasted compute cycles, delayed training schedules and lost engineering productivity.

1. **Direct financial impact of lost GPU hours is $250,000 to $750,000 annually in a 1000 GPU cluster**. Network-induced job failures typically force AI workloads to restart from the last checkpoint, discarding all progress made since that point. As outlined in the previous section, a 1,000-GPU cluster can experience 100 to 250 such events per year resulting in 500 to 1,500 GPU-hours lost per event, depending on the checkpointing frequency and recovery latency. When multiplied by an estimated $3 GPU-hour, the resulting financial loss from idle GPUs alone ranges from $250,000 to $750,000 annually. These figures represent tangible, invoice-visible costs directly tied to underutilized compute resources.
2. **Lost engineering productivity of ~$250,000 - 300,000 annually in a 1000 GPU cluster, and frustration!** Each incident of network disruption triggers a cascade of operational activity: incident triage, root cause analysis, corrective action (such as optic cleaning or replacement), and cluster revalidation. On average, each event consumes 2 to 4 hours of engineering time, impacting 4 - 6 people across infrastructure and data science teams. At 100 to 250 disruptive events per year in a 1000 GPU cluster, this translates to 2000 - 2400 staff-hours per year even using conservative estimates, and does not account for the added coordination overhead or emotional fatigue. Persistent failures introduce operational friction - delayed handoffs, blame cycles, and reduced morale - that further degrade team efficiency.
3. **Time to output (e.g., model updates)**. Disruptions also affect “time to output” - the time it takes to complete model training and move new models into production. Further, aggressive checkpointing, while reducing lost computation, can introduce I/O bottlenecks that directly impact the "wall clock time". delayed model deployment can translate into deferred feature releases, postponed product launches, and ultimately, reduced business agility. .

Disruptive link flaps and network failures represent more than just transient technical glitches - they carry a heavy operational and financial burden. **In a 1,000-GPU cluster, the combined effect of wasted GPU-hours, lost engineering productivity, and delayed time-to-market can result in $500,000 to $1 million in direct financial impact**, even before accounting for long-term erosion in team performance and model development velocity.

Clockwork's Software-Driven Fabric Architecture Enables Resilient AI Workloads

Clockwork's mission is to "Accelerate AI with fast, functional fabrics". Clockwork’s Software-Driven Fabric (SDF) architecture leverages software instead of proprietary hardware to deliver resilience, determinism, and superior price-performance.

Two innovations form the foundation of Clockwork's FleetIQ platform:

1. **Clockwork’s Global Clocksync** aligns every node’s internal clock, across thousands of machines, to within nanoseconds, using a lightweight, peer-to-peer probe mesh and applying machine learning and graph optimization to achieve near-perfect synchronization. The result is tens of thousands of nodes - within a data center or across regions - operating on a unified nanoseconds-accurate timeline and serving as the foundation for ultra-dense telemetry fabric. Infrastructure teams can use Clockwork’s dashboards and APIs to monitor fabric health and job progress with great precision in real time.

2. **Dynamic Traffic Control (DTC)** manages network traffic in real-time purely through software. DTC automatically optimizes paths, mitigates congestion, and quickly adapts to network disruptions without manual intervention. Through integrations like NCCL plugins, it can steer queue pairs, reroute collective operations, or optimize message distribution across the fabric. Because DTC is software-based, it works seamlessly across diverse hardware setups - multi-vendor Ethernet networks, Infiniband networks or RoCE deployments, supporting compute clusters powered by NVIDIA, AMD, and other accelerators.

Building on these foundational building blocks, **Clockwork delivers workload resilience through Workload Failover - a key part of its FleetIQ platform.** Workload Failover is designed specifically to shield distributed training jobs from the impact of disruptive link failures. Workload Failover introduces a new class of failure-handling logic: one that is **job-aware, path-sensitive, and recovery-optimized**. When a path degrades beyond usable thresholds - whether due to optic resets, hardware failures or localized switch faults - Clockwork dynamically reassigns active queue pairs to alternate paths, maintaining the integrity of collective communication groups. Rather than restarting jobs when the fabric misbehaves, Workload Failover ensures forward progress continues, and when the network heals, jobs rebound automatically to full speed. This “graceful degradation” model transforms job reliability from a binary outcome (fail or succeed) into a spectrum where small faults trigger small slowdowns - not catastrophic resets.

A picture speaks a thousand words! The attached screenshot from a demo of Workload Failover illustrates the impact of a NIC failure without Workload Failover - the job dies until it is restarted. With Clockwork's Workload Failover deployed, the throughput of the job decreases by ~15% when the NIC fails as the traffic is rerouted to leverage healthy NICs - no job restart needed! The job subsequently fails back with no degradation in throughput as soon as the failed NIC is reconnected.

Clockwork's Vision

Disruptive network failures have become an expensive and pervasive threat to large-scale AI training, draining hundreds of thousands of GPU hours, eroding engineering productivity, and stalling time-to-insight. Yet this fragility is not inevitable. Clockwork’s software-driven fabric architecture - anchored by nanosecond-accurate clocksync, real-time traffic control and resilient workload failover - transforms the way GPU clusters handle failures. Instead of halting jobs, Clockwork enables clusters to absorb, adapt, and advance. As AI infrastructure scales toward ever-larger models and tighter timelines, resilience isn’t just a feature—it’s a foundation. Clockwork makes that foundation real, replacing failure-induced chaos with consistent, confident progress.

Sources:

1. Falcon: Pinpointing and Mitigating Stragglers for Large-Scale Hybrid-ParallelTraining, 2024
2. The Llama 3 Herd of Models, 2024
3. "Alibaba HPN: A Data Center Network for Large Language Model Training", ACM SIGCOMM '24
4. Gemini: Fast Failure Recovery in Distributed Training with In-Memory Checkpoints, 2023
5. Averages used: mid-point of events is 175 events per 1000GPU cluster; range of lost time is 500-1500 GPU hours; $3 = cost per GPU-hour
6. Estimated averages: ~1 - 2 infrastructure engineers, 3 - 4 data scientists lose 3 hours per event (troubleshooting, remedy, fleet preparation for restarts), with 100-250 events per year in a 1000-GPU cluster; average FTE cost to company of $250K

To request a demo, contact us at **hello@clockwork.io**