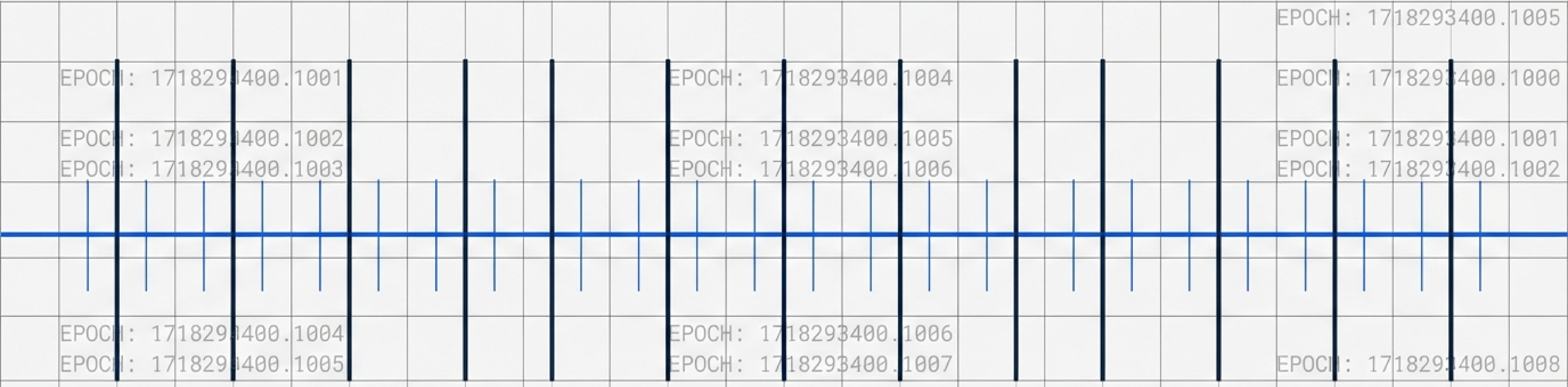
Mathematical Determinism at Exascale

Eradicating AI Network Incast and Recovering 15% GPU Yield via O(1) eBPF Predictive Pacing



The Compute Yield Crisis at Hyperscale

The Bottleneck

As AI models scale to tens of trillions of parameters, the reactive Ethernet protocols (TCP/IP, RoCEv2) have replaced raw compute as the primary scaling bottleneck.

15%

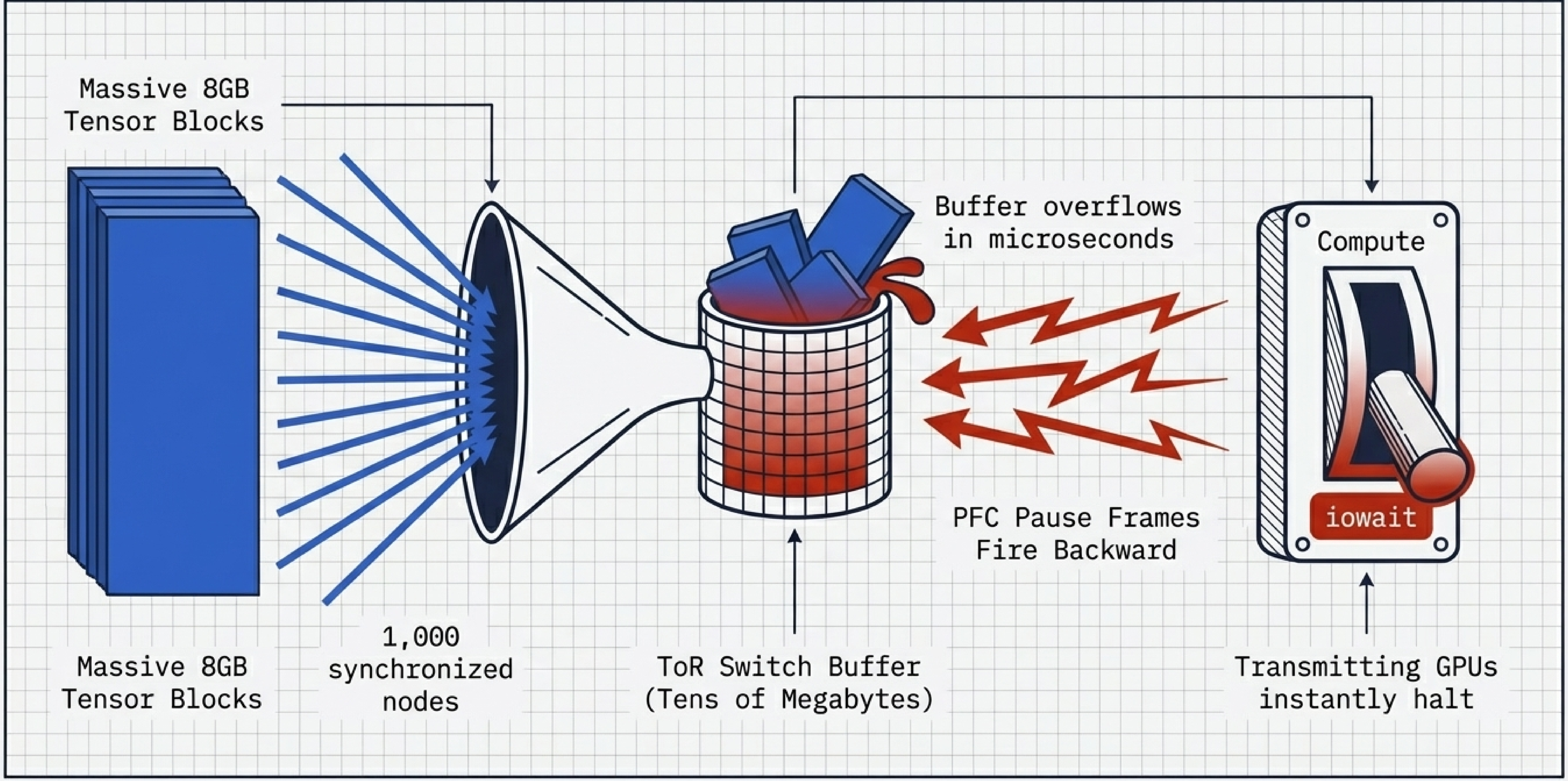
The Reality

In a 1,000-node GCP A3/A3-Mega deployment, a 15% network stall means 150 nodes are consuming megawatts of electricity but contributing zero mathematical progress.

The Penalty

Synchronized AI training phases cause network stalls that routinely degrade overall Time-to-Train by 15% or more.

The Physics of Many-to-One Incast



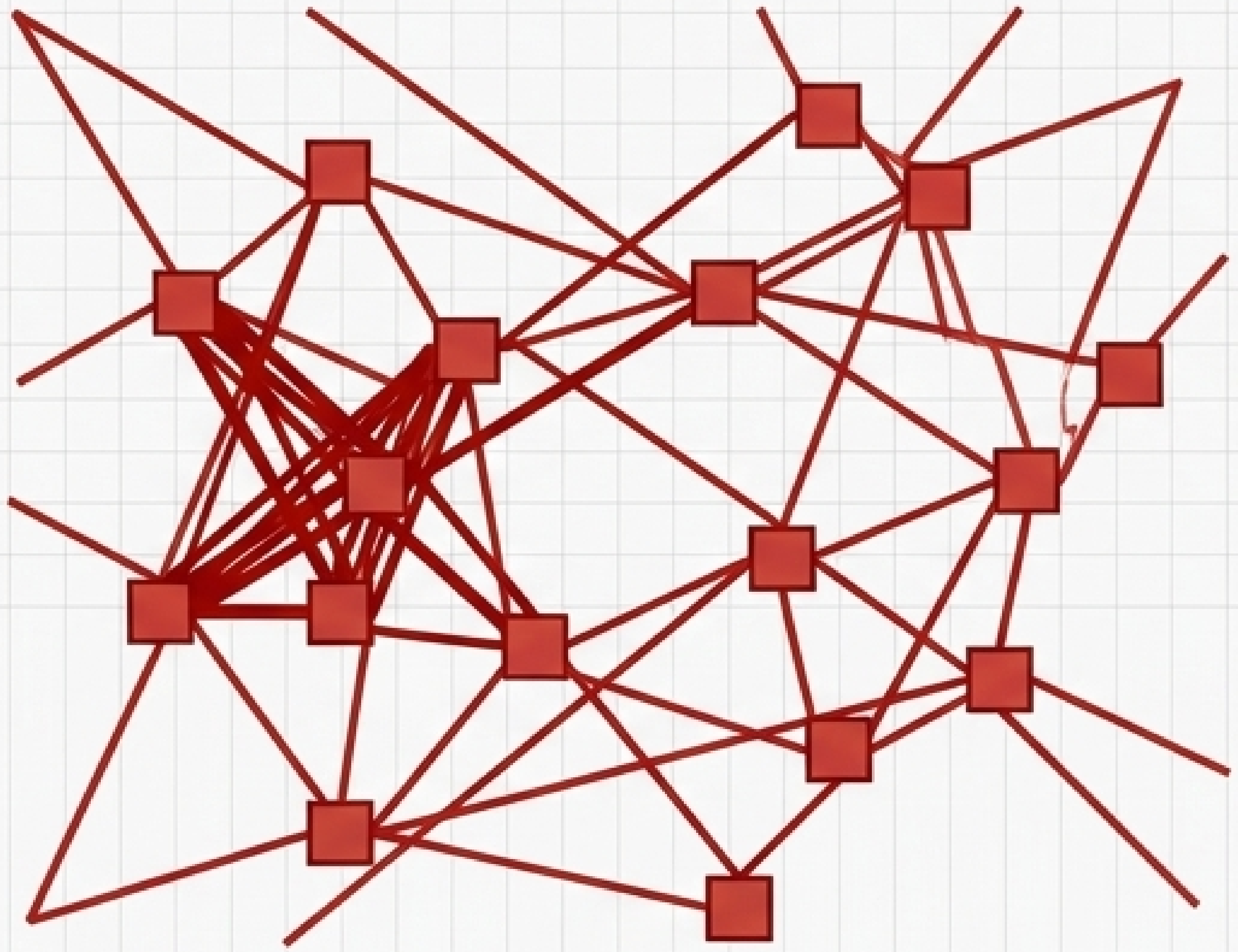
The Fallacy of Reactive Hardware

	Deep Buffer Switches	InfiniBand	Standard RoCEv2	PTCP (Software Pacing)
Mechanism	Reactive memory caching	Reactive credit-based	Reactive blind transmission	**Predictive Software Pacing**
Efficacy at Exascale	Delays the drop marginally	Struggles with adaptive routing latency	**Catastrophic buffer overflows**	Mathematically deterministic
Cost / CapEx	Extreme power/CapEx	Severe premium	Commodity	Zero hardware CapEx
Vendor Lock-in	High	Absolute	None	None

Reactive hardware protocols transmit blindly at line rate and react only after congestion occurs. The solution requires a fundamental shift to prediction.

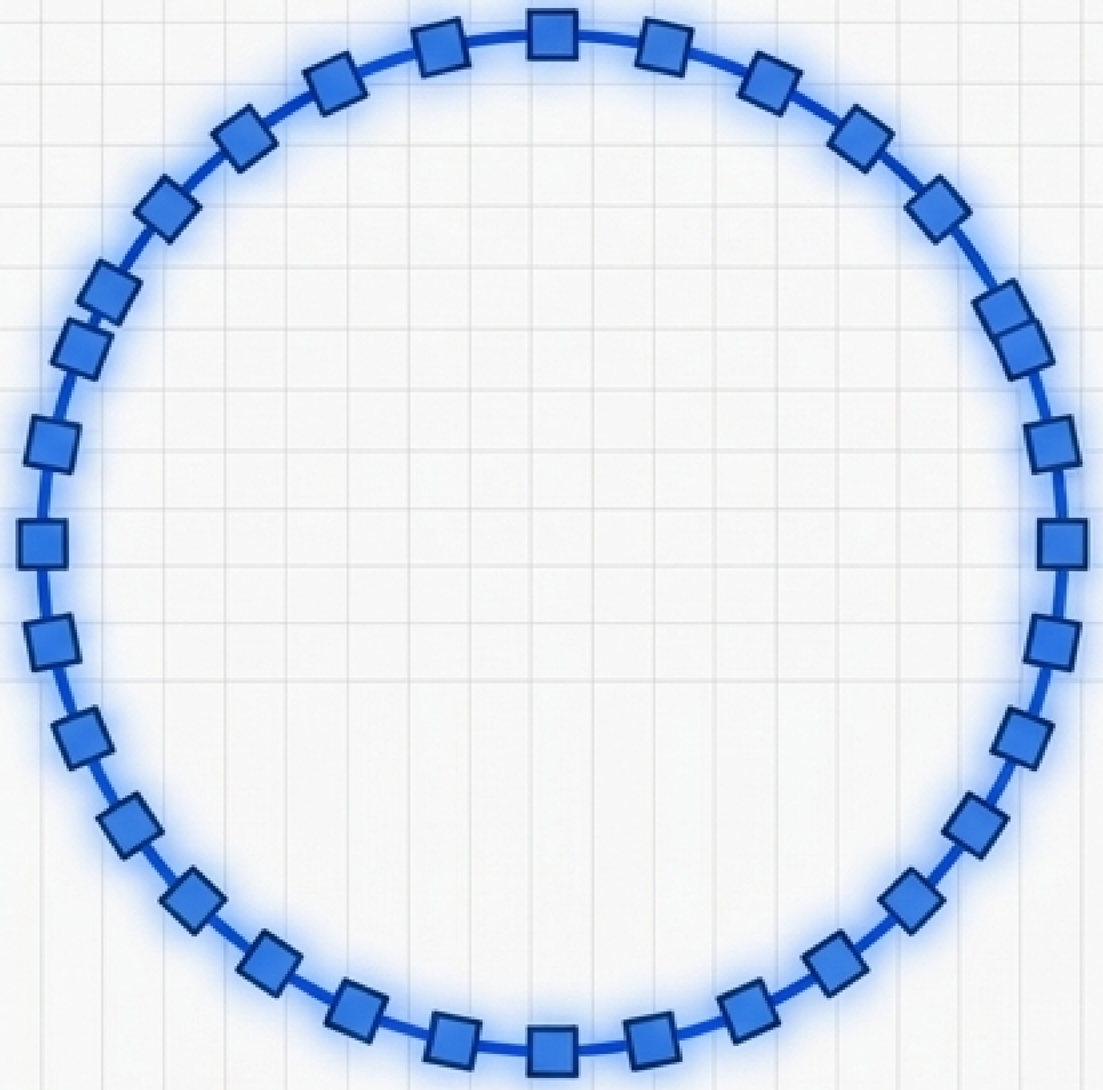
Software-Defined Determinism

The Past



Reactive Buffering

The Future

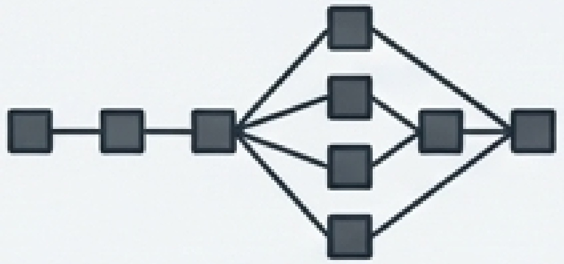


Predictive Host-Level Pacing

Network determinism is no longer a hardware problem; it is a software engineering discipline. Tensor Networks' Predictive Tensor Control Plane (PTCP) bypasses reactive networking entirely by operating on feed-forward mathematical determiniism directly within the Linux kernel.

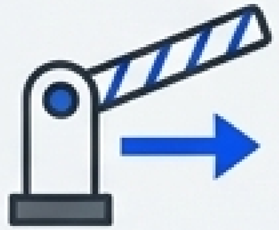
The PTCP Architecture Stack

Application Layer



PTCP Orchestrator parses the AI execution graph to predict exactly when GPUs will fire their tensors, mapping cluster topology and line rate.

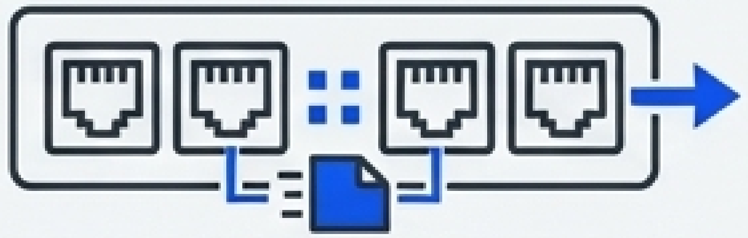
Host OS (Linux Kernel)



`Linux Traffic Control (clsact)`

PTCP pushes the Earliest Departure Time (EDT) mathematical envelope into a pinned eBPF map.

Hardware



Outbound socket buffers (skb) intercepted. Packets are mathematically paced before hitting the wire, bypassing the reactive hypervisor.

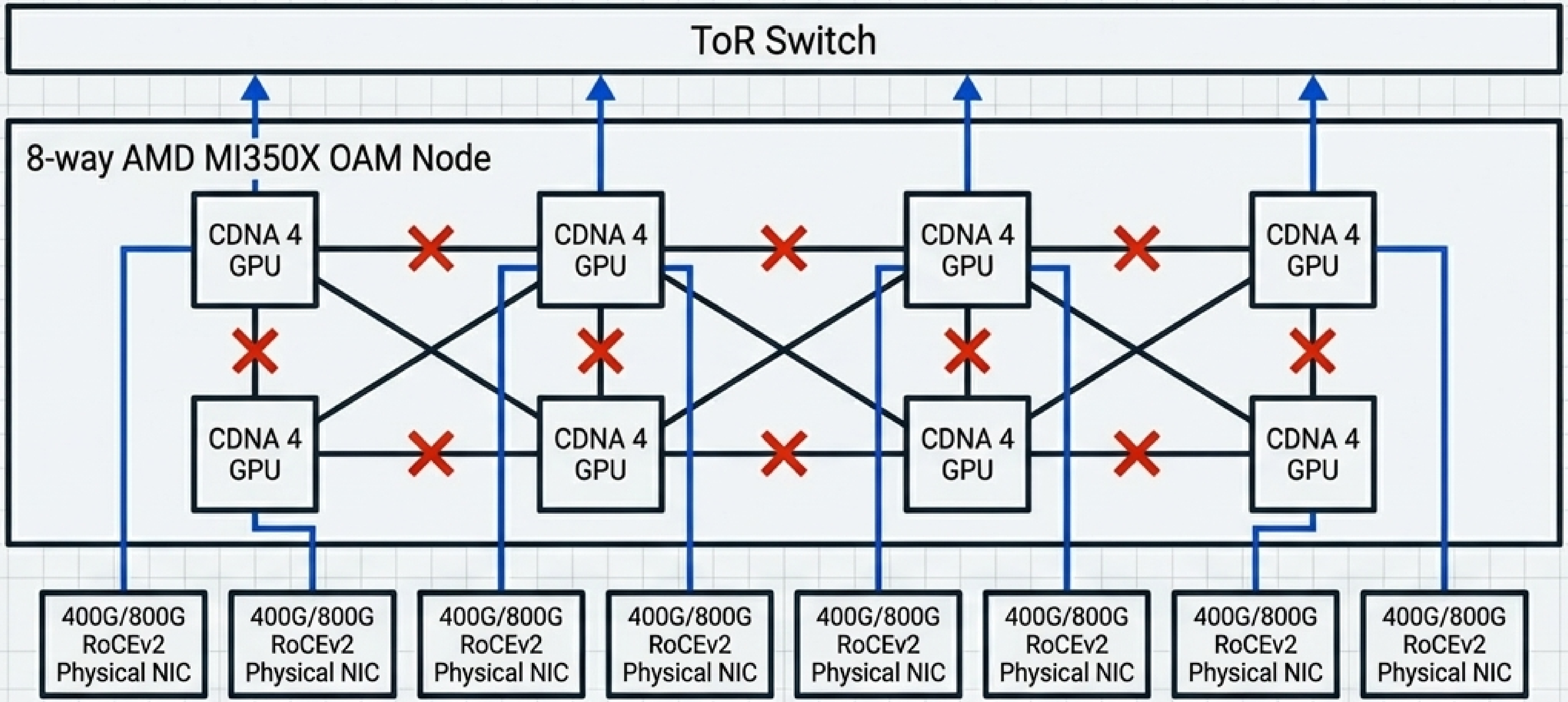
O(1) eBPF Predictive Pacing

$$\Delta t = \frac{MTU \times N}{C}$$

Δt : Exact nanosecond inter-packet gap
MTU : Payload Size
N : Number of concurrent transmitting nodes
C : Port line rate

Instead of allowing the NIC to transmit at an unconstrained 400 Gbps or 800 Gbps, PTCP applies an absolute Earliest Departure Time (EDT) to each packet. The tensors arrive at the ToR switch perfectly interleaved. The switch buffer never fills.

Empirical Validation: The AMD MI350X Constraint Setup



To simulate a 1,000-node GCP deployment locally, we mathematically force the GPUs to ignore internal xGMI/SHM and route all gradient tensors through the physical Ethernet RoCEv2 fabric while injecting background Incast congestion.

Phase 1: Manipulating Local Physics

```
# Force RCCL to use the RoCEv2 network interfaces, abandoning internal xGMI
export NCCL_P2P_DISABLE=1
export NCCL_SHM_DISABLE=1
export NCCL_ALGO=Ring
export NCCL_SOCKET_IFNAME=ens
```

To trigger the Many-to-One microburst on a single switch, a background iperf3 UDP flood targets GPU 0's NIC at the exact moment the benchmark runs.

Phase 2: Execution & The eBPF Actuator

The RCCL Stress Test

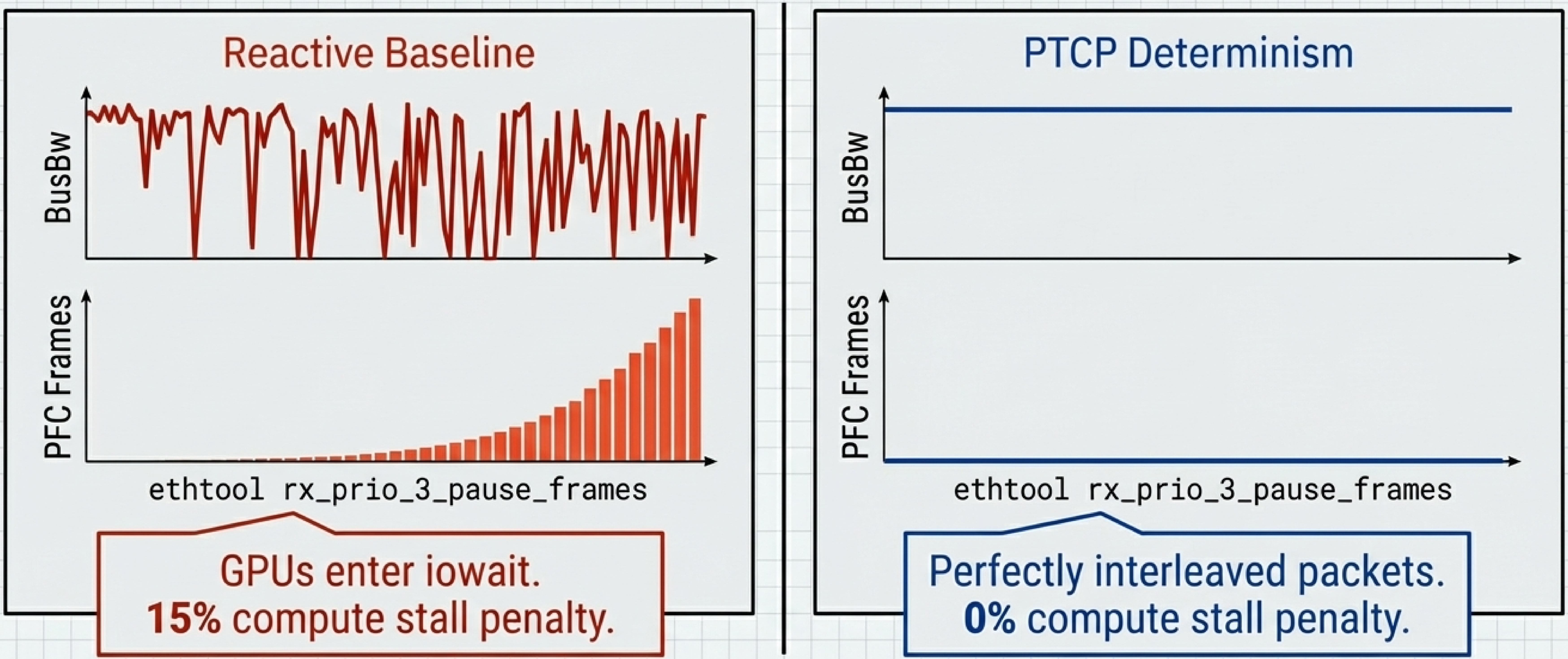
```
mpirun -np 8 \
  "-x NCCL_P2P_DISABLE=1 -x
NCCL_SHM_DISABLE=1 \
  "-x NCCL_SOCKET_IFNAME=ens \
  "./build/all_reduce_perf -b 8G -e
8G -f 2 -g 1 -w 10 -n 50"
```

Deploying PTCP

```
ptcp-agent attach \
  "--interfaces ens21np0,ens22np0… \
   "--mode rocev2"
```

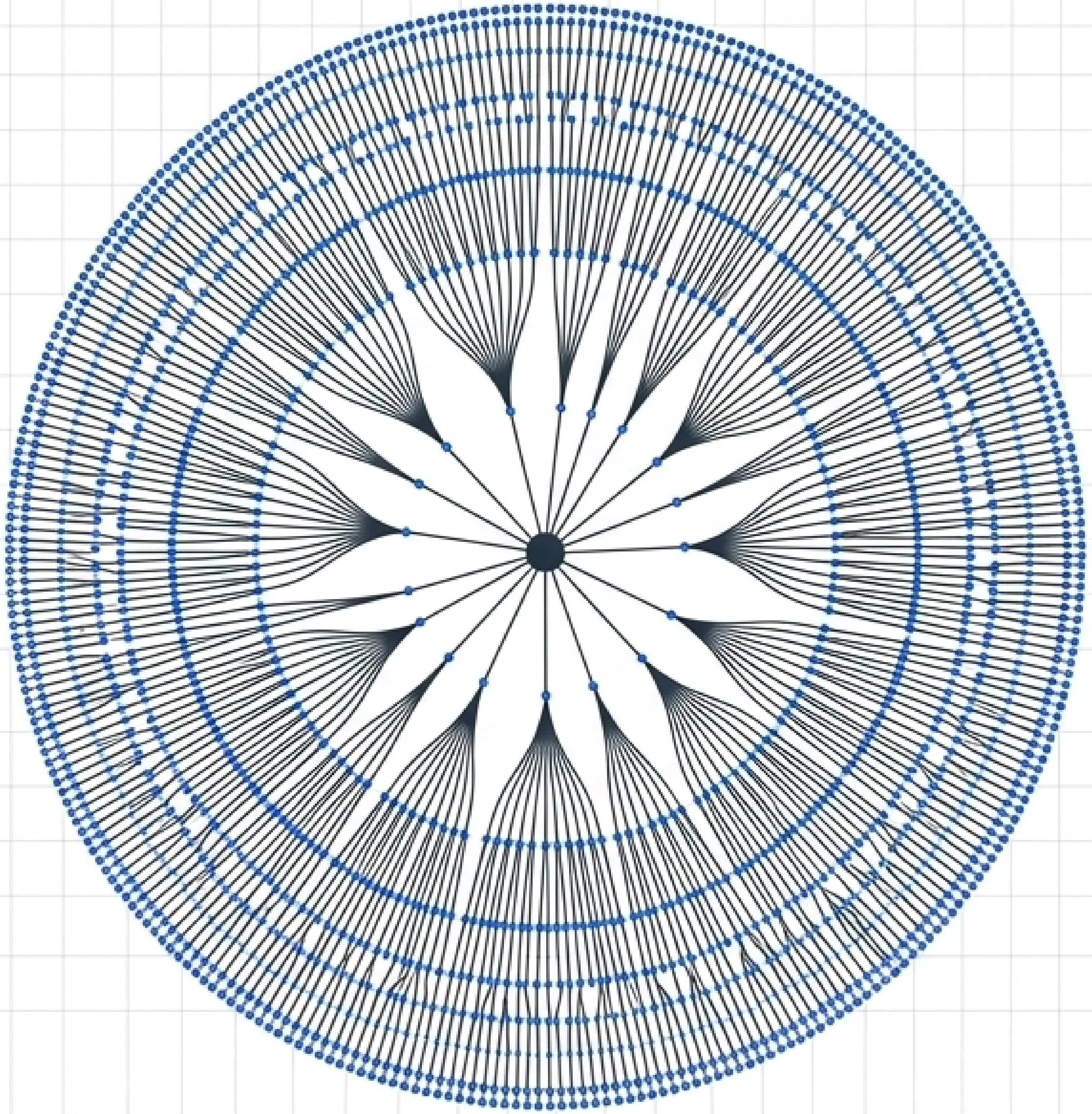
The test executes a massive 8GB tensor reduction across 8 GPUs for 50 iterations. PTCP's C-code is loaded into the kernel and attached to the Traffic Control (clsact) egress hooks of all 8 physical NICs.

The Result: Eliminating the 15% Network Penalty



Despite identical hostile background congestion, the Linux FQ discipline combined with EDT pacing mathematically eradicates switch buffer overflows.

Scale-Out: The 100,000+ Node Reality

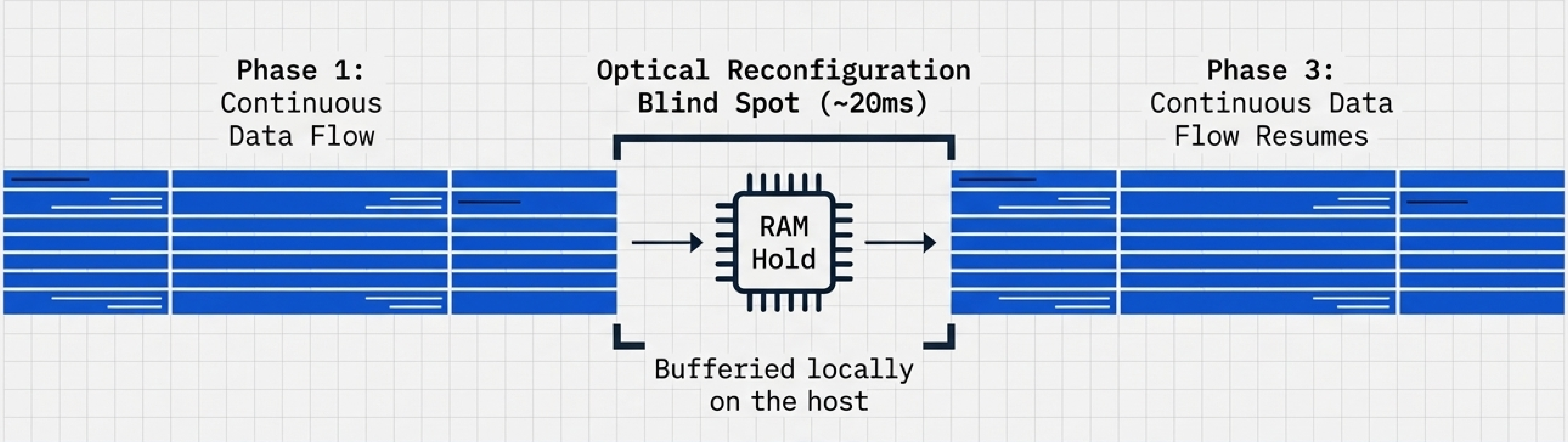


As hyperscalers build 100,000 to 300,000 GPU factories, microsecond-level Incast collisions increase exponentially.

Because PTCP's O(1) eBPF pacing is calculated independently at the edge (the host kernel), it scales infinitely without centralized controller latency.

Outperforms InfiniBand fabrics, which structurally struggle with adaptive routing latency at massive scale.

Scale-Up: Optical Circuit Switch (OCS) Integration



The Mechanism	PTCP Capability
To eliminate electronic spine switches, the industry is moving to OCS (e.g., Lumentum 3D MEMS). OCS physically moves mirrors, creating a ~20ms blind spot that destroys RoCEv2 packets.	PTCP utilizes its eBPF EDT logic to predictively blank host transmission during optical shifts, unlocking dynamic, zero-drop topology morphing for commodity Ethernet.

Redefining Hyperscale ROI

15%
Recovery

Guaranteed baseline expansion in pure GPU compute yield.

150
GPUs

Capital directly recovered per 1,000-node cluster deployment

0
CapEx

InfiniBand-level determinism achieved entirely on commodity hardware via host-kernel software.

The physics of hyperscale AI have outgrown traditional Ethernet. Tensor Networks' PTCP proves that mathematical determinism is the only scalable foundation for the AI factories of tomorrow.