清華大學
Tsinghua University

组会论文分享

Paper Info: 【SIGCOMM 24】

Paper Name: Alibaba HPN: A Data Center Network for Large Language Model Training

Reporter Name: Senj Lee

Report Date: 2024/09/19

Content

- Backgrounds & Goals
- Architecture
- Evaluation
- Experience & Lessons

Backgrounds & Goals

- Distinct features of LLM training.
- Goals with practical considerations.

Background

Due to the differences between LLMs and general cloud computing, traditional data center networks are not well-suited for LLM training!

Problem 1: Traffic Patterns

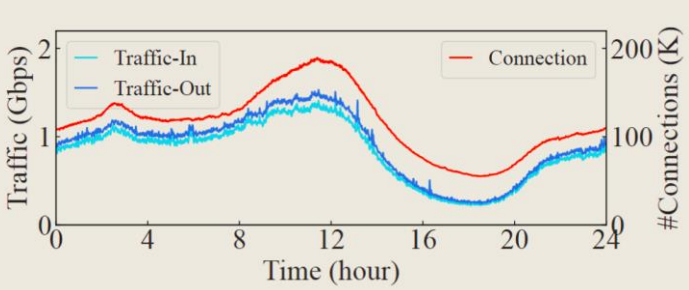


Figure 1: Traditional cloud computing traffic pattern.

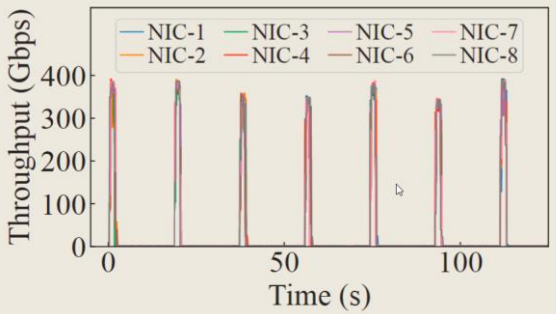


Figure 2: NIC egress traffic pattern during production model training.

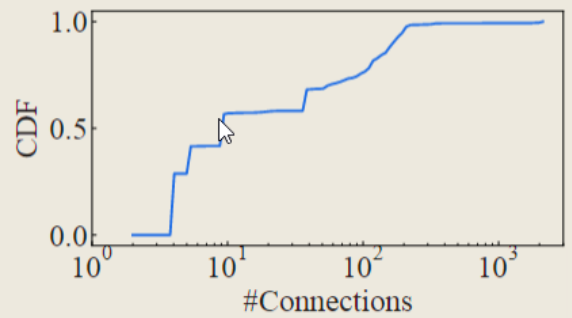


Figure 3: Number of connections per host.

- Generate millions of flows, which gives the network high entropy. (→ ECMP scheme)
- Each flow is continuous and low-utilization for NIC capacity (under 20%).

- Generate very few[^1] but periodically bursty flows,
- resulting in low entropy and high utilization (directly reach the NIC capacity).

ECMP scheme works terribly bad in the case of LLM training!

Background

Problem 2: Higher sensitivity to faults, especially single-point failures.

1. First, LLM training is more sensitive to failures.
2. Second, failures in LLM training can result in significant costs.

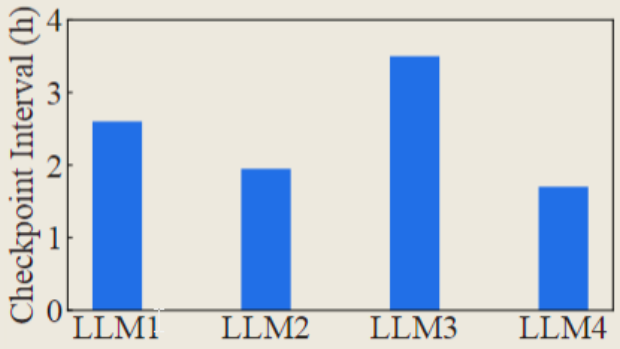


Figure 4: Checkpoint intervals of representative LLM jobs.

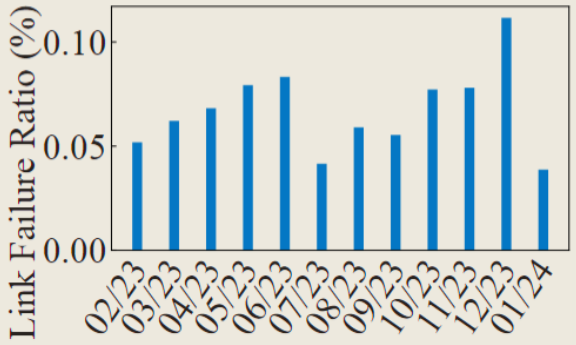


Figure 5: Monthly link failure ratio.

Single-point failure matters!

Under this high failure rate, a single LLM training job would encounter 1-2 crashes each month.

Goals

G1: Scalability: #GPUs 1K → 3K → 15K → 100K.

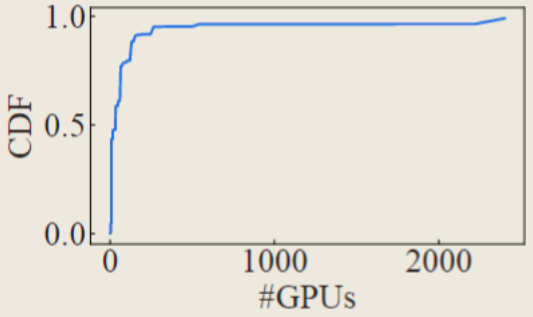


Figure 6: #GPUs used in production training jobs.

G2: High Performance: Minimize network hops as much as possible.

G3: Single-ToR fault tolerance.

Architecture

- Overview.
- Frontend Network / Backend Network (Dual-ToR, Tier1, Tier2, Tier3)

Architecture

Overview:

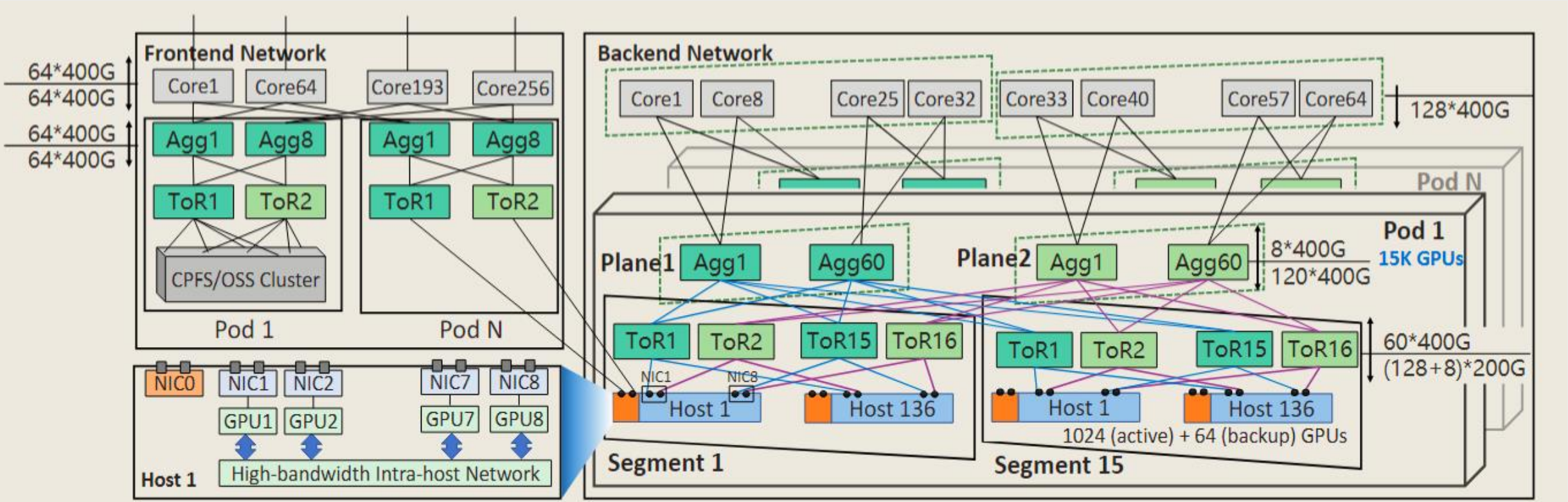
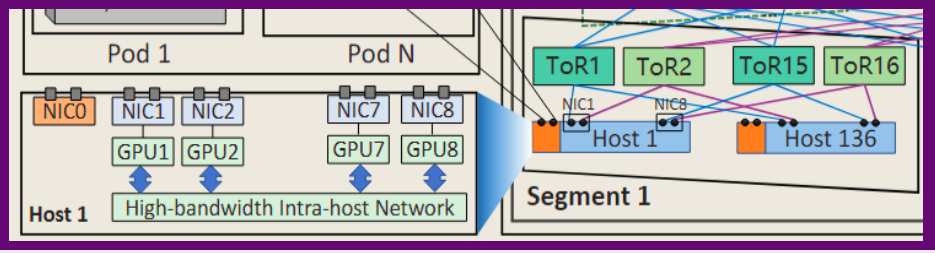
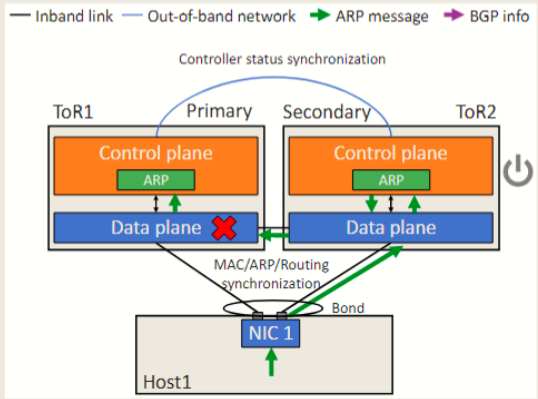


Figure 7: HPN overview. A solid parallelogram represents a segment (containing 1024 active GPUs and 64 backup GPUs). Two dotted parallelograms represent dual-plane. A cube contains an entire Pod (containing 15K GPUs).

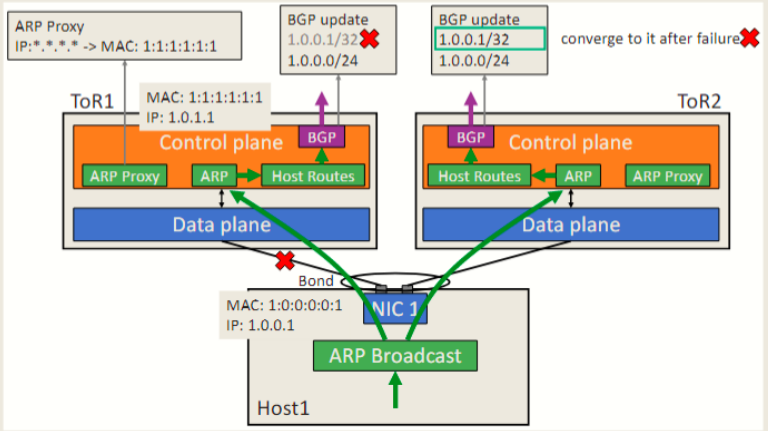
Architecture



Access1: Non-stacked Dual-ToR



(a) Stacked dual-ToR.



(b) Non-stacked dual-ToR.

Figure 8: Dual-ToR solutions. In stacked dual-ToR (a), the malfunction of ToR1's data plane would eventually trigger the offline of ToR2. In non-stacked dual-ToR (b), two ToRs run independently.

Stacked dual-ToR is good enough, but there are still some issues that could be improved:

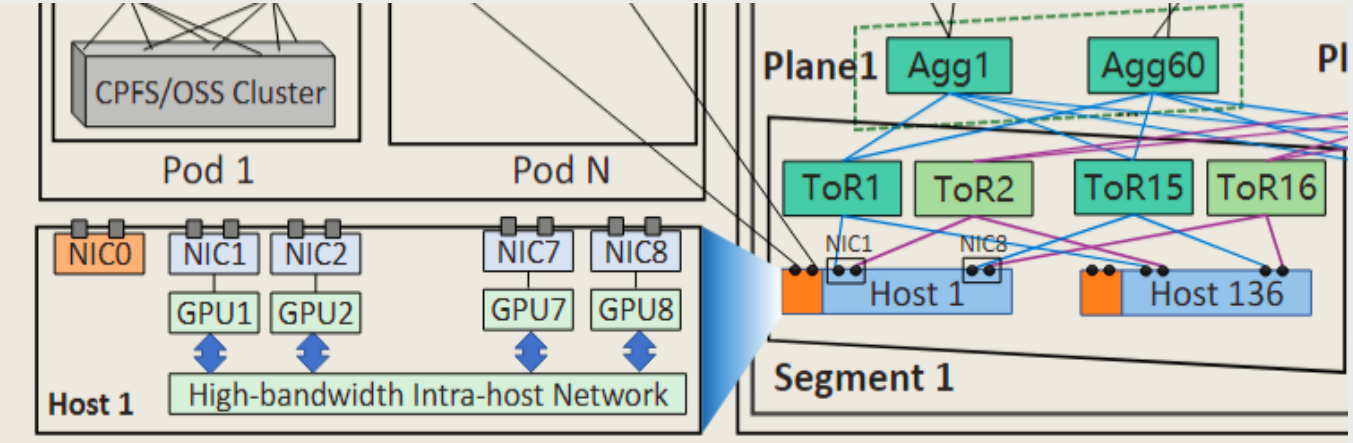
- Stack failures,
- Issues resulting from ToR upgrades.

The root cause of failures in stacked dual-ToR is the strong dependency on synchronization via the direct link between two ToRs. Therefore a new dual-ToR scheme is proposed in the paper:

1. Bundle two independent links.
   - Same Mac address
   - Different portIDs
   - Update ARP information concurrently on the two ToRs by duplication
2. Maximally leverage BGP under failures.

Architecture

Access2: Tier1: 1K GPUs in one Segment



HPN employs the latest 51.2Tbps (400Gbps×8×16) Ethernet single-chip switch. In tier1, each switch possesses 128 active + 8 backup 200Gbps downstream ports and 60 upstream 400Gbps ports.

Why single-chip switch?

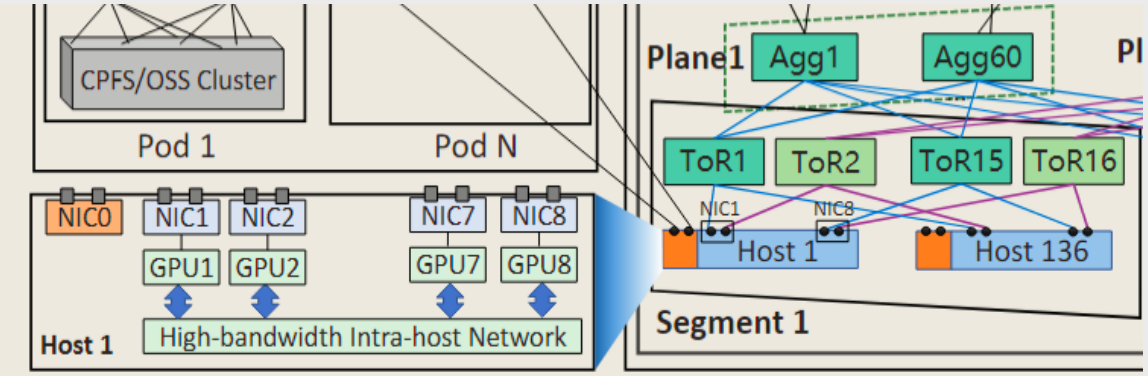
- The bandwidth capacity of the ToR switch directly determines the number of GPUs in the same tier1 network.
- The root cause is that the multi-chip switch is a distributed switching system, with multiple chips interconnecting through a chip fabric. Failures in the internal fabric, inter-chip interactions, and chip-to-CPU communication all contribute to the overall critical outages.

Challenges introduced by high-throughput single-chip switch.

- Power consumption.
- Cooling system.

Architecture

Access2: Tier1: 1K GPUs in one Segment



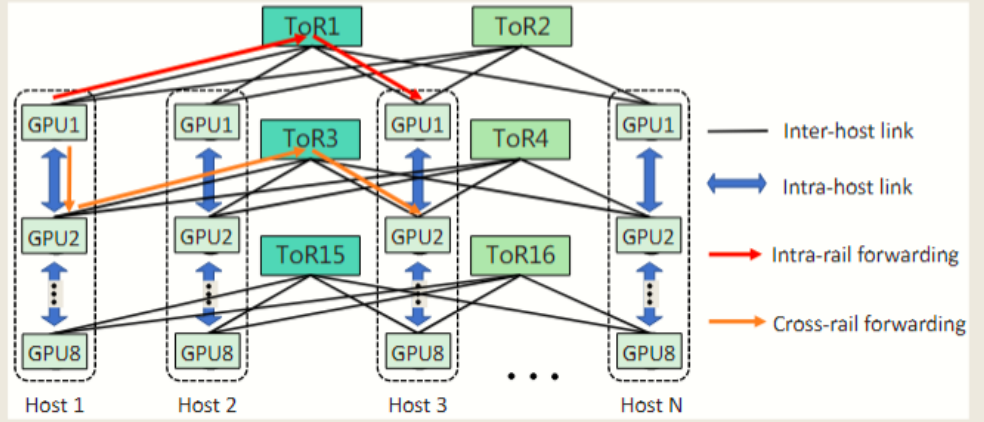


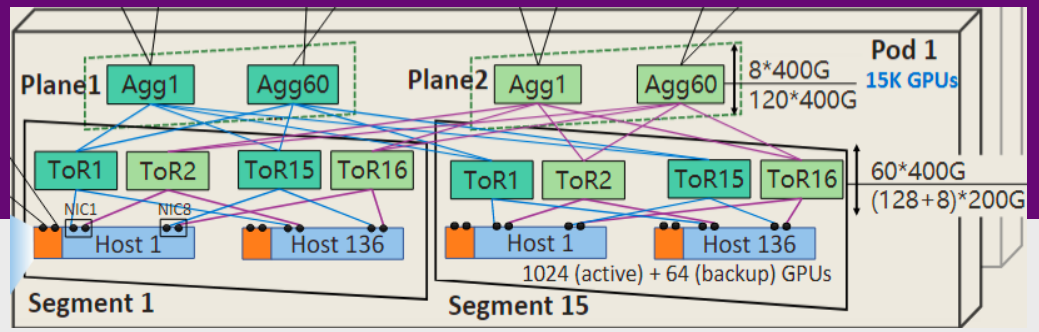
Figure 11: Rail-optimized network under dual-ToR.

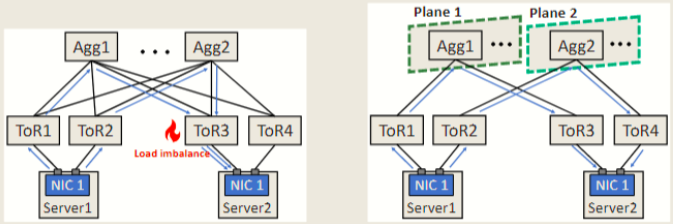
Rail-Optimized Network.

- What is rail? (Same order GPU in different hosts)
- Why proposed the concept of the rail-optimized network? (Nvidia)
- How rail-optimized network runs?
- What are the advantages of rail-optimized networks? Each set of dual-ToR switches can serve 128 GPUs, and the 16 ToRs collectively connect 1024 GPUs in a segment, substantially reducing the forwarding latency and providing the utmost performance. More importantly, it significantly reduces traffic crossing the Aggregation layer, lowering the possibility of load imbalance in the network.

Architecture

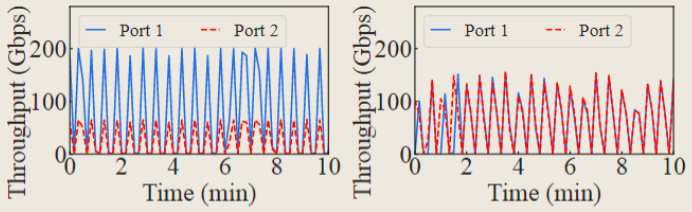
Access3: Tier2: 15K GPUs in one Pod





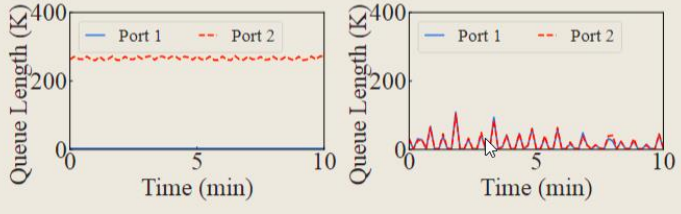
(a) Typical Clos. (b) Dual-plane.

Figure 12: Tier2 network architecture.



(a) Typical Clos. (b) Dual-plane.

Figure 13: Traffic on ToRs' ports towards the same NIC.



(a) Typical Clos. (b) Dual-plane.

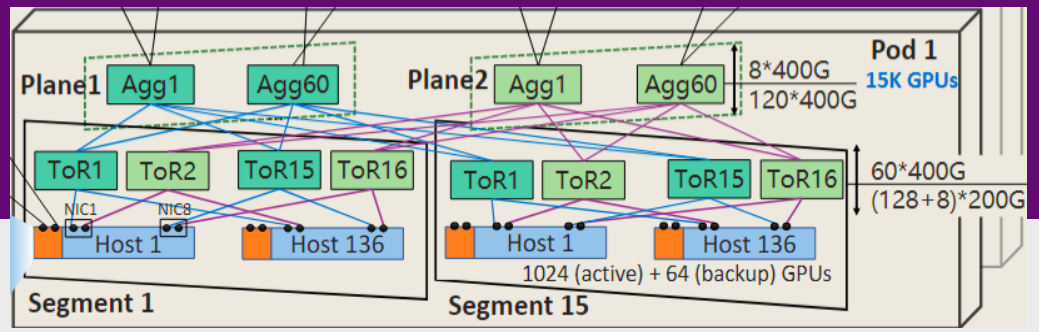
Figure 14: Queue length at downstream ports of ToR.

Minimize load imbalance.

1. If data center simply deploy a typical Clos topology between ToR and Aggregation, hash polarization would still exist. To eliminating hash polarization in a Pod, the paper proposed Dual-plane scheme. As shown in Figure 12b, in dual-plane, ToR switches in each dual-ToR set are categorized into two separate groups. Further ablation study reveals that the dual-plane design contributes up to 71.6% performance improvement for cross-segment traffic.
2. To solve the load imbalance caused by ECMP[^2], HPN gets precise disjoint equal paths efficiently and balances the load on them in the collective communication library.
    - First, for each new connection request, HPN generates a set of connections passing through disjoint paths.
    - Second, HPN implements a simple yet effective application layer load-balancing scheme to fully utilize all RDMA connections.

Architecture

Access3: Tier2: 15K GPUs in one Pod



Stuff 15K GPUs in One Pod.

The dual-plane design brings another important benefit: it halves the number of link connections between ToR and Aggregation, allowing Aggregation switches to support more segments in the same Pod. As a result, the scale of the tier2 network is doubled.

Table 2: Key mechanisms affecting maximal scale

	Tier1 scale	Tier2 scale
51.2Tbps Clos	64	2K
Dual-ToR	128 (×2)	4K (×2)
Rail-optimzed	1K (×8)	-
Dual-plane	–	8K (×2)
Oversubscription of 15:1	–	15K (×1.875)

Architecture

Access4: Supporting Larger Scale

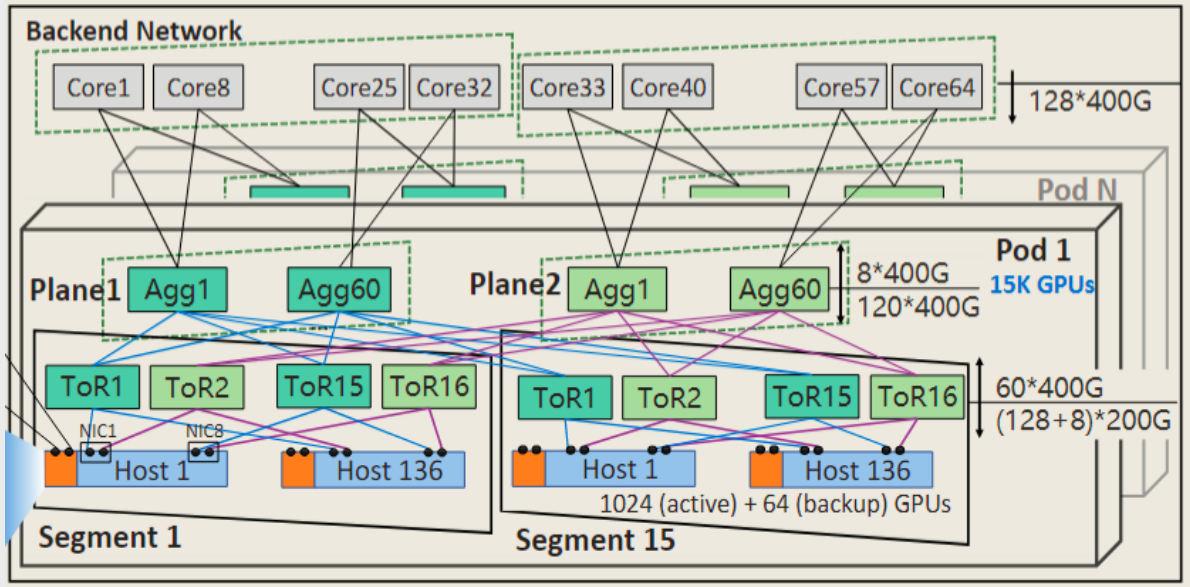
Tier3: Connect multiple Pods via Core Layer.

There is a trade-off between the Aggregation-Core oversubscription and the scale of the entire cluster.

- we set the Aggregation-Core oversubscription to be 15:1, which releases 87.5% more ports on the Aggregation switches for interconnecting extra segments.
- Deep diving into the communication pattern in the LLM training, we find that the single training job across tens of thousands of GPUs does not require excessive tier3 bandwidth capacity.

Table 3: Traffic patterns of different parallelisms

	DP	PP	TP
Traffic volume	5.5GB	6MB	560MB
Operations	AllReduce	Send/Recv	AllReduce/AllGather



Different parallel strategies introduce different volumes of data transmission. As shown in Table 3, PP generates the lowest traffic and utilizes the basic Send/Recv for communication, which is insensitive to network bandwidth.

Deploying tier3 may introduce additional load imbalance risks. To minimize this side effect, we make two enhancements. (1) We carry on the dual-plane design in the Core layer. (2) In each Core switch, we employ a prior perport hash [69] to ensure traffic towards Pod i from physical port $j$would uniquely forward to port k (5-tuple irrelevant), eliminating hash polarization.

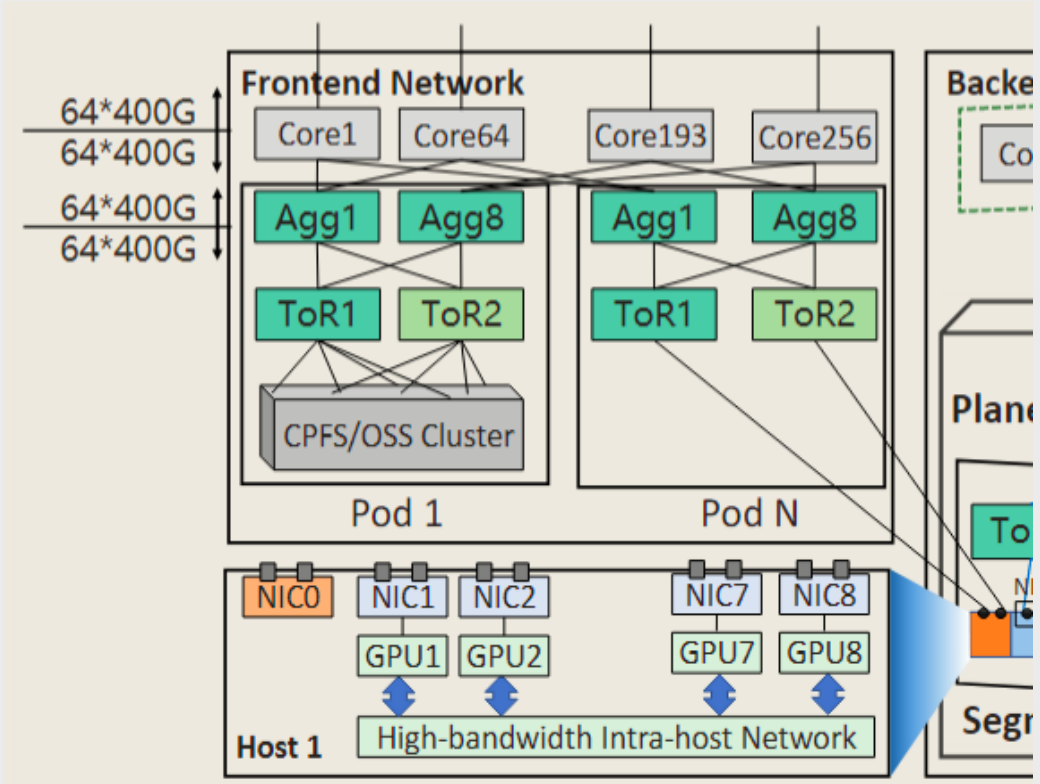
Architecture

Access5: Independent Frontend Network

The frontend network primarily handles management and storage traffic (e.g., cluster management, dataset loading, image loading and checkpoint saving/loading). It can also carry inference requests/responses while serving model inferences.

To ensure reliability, each frontend NIC connects to two ToRs in the non-stacked dual-ToR way. In the frontend network, we design the convergence ratio to be 1:1 at both Aggregation and Core layers, guaranteeing maximal bisection bandwidth.

- Isolate storage traffic from training.
- Support inference compatibly. There is a trend to use training GPUs in inference. The reasons are twofold: (1) As the size of models increases, requiring GPUs with higher memory and performance for inference services, the specifications for GPUs used for inference are becoming increasingly similar to those used for training. (2) We observe that many customers prefer deploying both training and inference jobs on the same rented cluster for better GPU utilization.



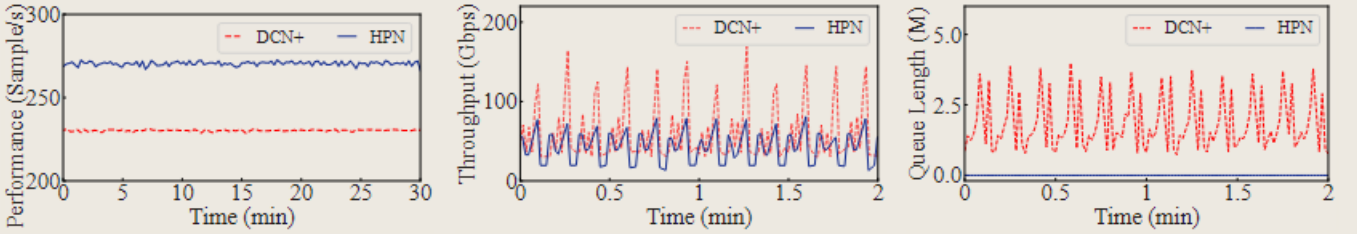
Evaluation

- Performance from the perspective of LLM training / network-level.
- Reliability.

Evaluation

Environment.

- HPN is deployed in multiple clusters connecting O(10K) GPUs in Alibaba Cloud, and serves thousands of model training jobs from dozens of customers.
- HPN versus DCN+ (previous generation of training network architecture, whose backend network is a traditional 3-tier Clos Data Center Network with full bisection bandwidth and dual-ToR enabled). In DCN+, each segment contains 128 GPUs, and each Pod contains 4 segments.
- Each host is equipped with 8 NVIDIA H800 GPUs and 9 NVIDIA BlueField3 2×200Gbps DPUs. GPUs in the same host are interconnected with 400GBps (bidirectional) NVLINK.

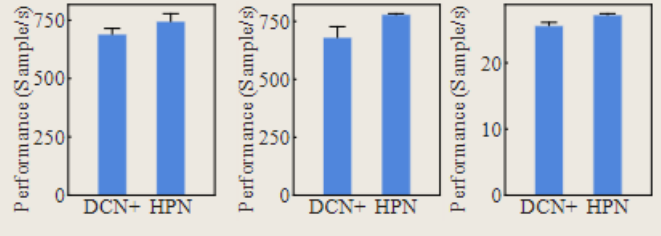


(a) End-to-end training performance.

(b) Aggregation switch ingress traffic (averaged every 10s).

(c) Queue length in Aggregation switches (max every 10s).

Figure 15: Model training performance on 2300+ GPUs under different network architectures.



(a) LLaMa-7B.

(b) LLaMa-13B.

(c) GPT-175B.

Figure 16: Performance of training representative LLMs on different network architectures.

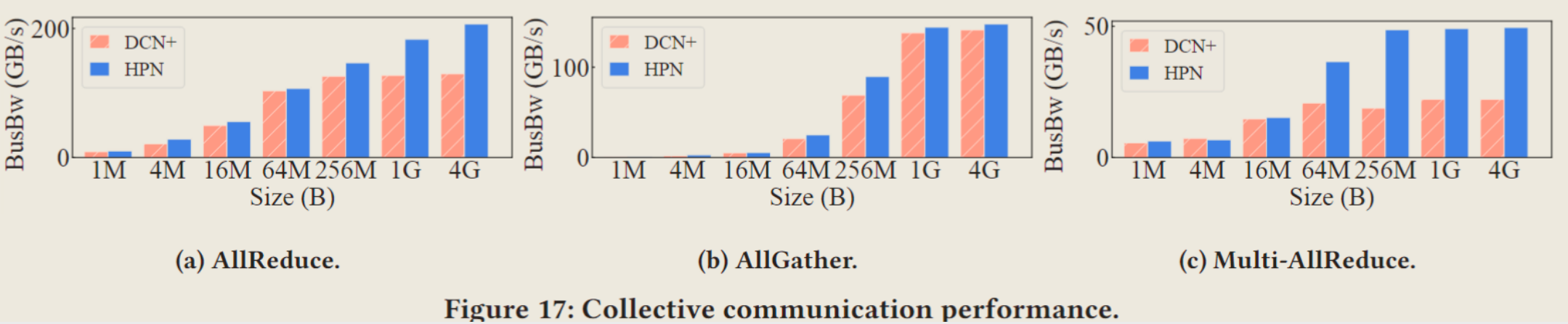
LLM Training Performance.

- Figure 15a, the end-to-end performance is improved by more than 14.9% (actually a big value in production).
- Figure 15b, the cross-segment traffic is decreased by 37% on average.
- Figure 15c, illustrate the queue length distribution of Aggregation switches' downlinks.
- Figure 16, end-to-end training performance is improved by 7.9%, 14.4% and 6.3%, respectively.

Evaluation

Network-Level Performance.

Evaluate the performance of typical collective communication operations (AllReduce and AllGather) with 448 GPUs (56 hosts), where AllReduce is the predominant operation in LLM training jobs.



(a) AllReduce. **(b) AllGather.** **(c) Multi-AllReduce.**

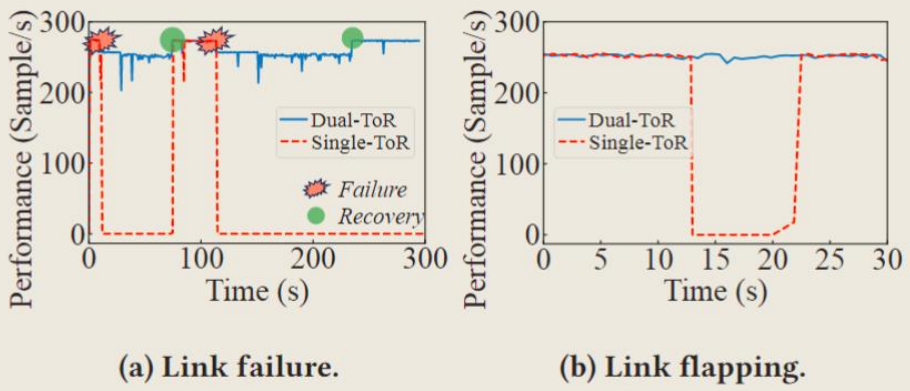
Figure 17: Collective communication performance.

- Figure 17a, HPN increases the performance of AllReduce by up to 59.3%.
- Figure 17b, the performance of AllGather is similar between HPN and DCN+.
- Figure 17c, HPN can increase Multi-AllReduce’ s performance by up to 158.2%.

Evaluation

Reliability.

In this subsection, we train LLaMa-7B with 256 GPUs (32 hosts), and inject link malfunctions (link failure and link flapping) on a NIC-ToR link. We compare dual-ToR with typical single-ToR design to validate the reliability improvement.



(a) Link failure. (b) Link flapping.

Figure 18: Performance under different types of NIC-ToR link malfunctions.

- Figure 18a, with dual-ToR, the failure of a single link only causes 6.25% performance degradation (repair within 1 min).
- Figure 18b, in single-ToR, the temporary link flapping halts the training for more than nine seconds, but the performance of dual-ToR is neligible.

Experience & Lessons

- Experience during the paper’s work.
- My gains after reading the paper.

Experience

One Pod in a single data center building.

- In conjunction with HPN, each single building perfectly houses an entire Pod, making predominant links inside the same building.

Asymmetric link state[^3] are possible.

- Thanks to dual-ToR design, this link leads to training performance degradation rather than the entire training job crashes.

HPN complicates wiring.

- To eradicate wiring mistakes before end-to-end testing, we employ INT-based probes to check that each hop (switchID and PortID) in paths precisely aligns with HPN's blueprint definition.

Why not employ the rail-optimized idea on tier2 to support larger scale?

- Rail-only tier2 heavily relies on models only generating intra-rail traffic. In current mainstream dense large models, all traffic patterns have been specifically optimized to satisfy this constraint. However, the evolution of new models would break this assumption. Relaying traffic across the intra-host network would be greatly limited, making the cross-rail network vital. Therefore, we decide to construct any-to-any tier2, and leverage tier3 to support a larger scale.

Experience

The location of the storage cluster. There are three main disadvantages to placing the storage cluster in the backend network:

1. Typically the container images and dataset used by customers are usually stored outside (in other data centers or customers network).
2. As aforementioned, injecting storage traffic in the backend network would result in fluctuations in training performance.
3. Deploying a storage cluster in the backend consumes ToR ports, reducing the number of GPUs the backend network can support. Therefore, we finally chose to place the storage cluster in the frontend network.

Why not leverage rail-optimized topology for handling ToR-related failures?

1. Proactively rerouting requires significant modifications on NCCL, which makes it hard to be employed by customers.
2. Manipulate the I/O direction of collective communication would introduce extra risks in production.

Lessons

Alibaba HPN 是一个最新的用于LLM训练的数据中心架构的技术报告，分层次地详细讲述了整体结构的设计思路和创新点，这启发我如果想要在此领域深入了解研究，可以关注以下方向的最新研究动态：

1. Topology for LLM training.
2. Dual-ToR solutions.
3. Load balance.

并且HPN在宏观上讲解了整体架构及部分技术，但更细节的实现以及进一步地优化，在这个新的数据中心架构中还有很大潜力，值得深入考量。

清華大學
Tsinghua University

Thanks for listening!

Senj Lee

2024 / 09 / 19