

Train Across Your Entire GPU Fleet:

Heterogeneous Mixed-Generation Training in Pure PyTorch

Roy Allela
Principal AIML Specialist SA, AWS

Aravind Neelakantan
AIML Specialist SA, AWS



Why mix GPU generations?

- Customers with mixed GPU fleets can use all their available capacity in a single training job instead of leaving non-matching instances idle.
- GPU throughput in heterogeneous mode achieves **76–94%** of an equivalently-sized homogeneous cluster, depending on model size and communication intensity.
- We train models across H100, H200, B200 and B300 nodes with **stock PyTorch** FSDP/HSDP. For cross-generation jobs add per-GPU-type micro-batch sizing.

GPU Instances Evaluated on AWS

Instance	GPU	HBM	EFA NICs	Rails/GPU
ml.p5.48xlarge	H100	80 GB	32 × 100G	4
ml.p5en.48xlarge	H200	141 GB	16 × 200G	2
ml.p6-b200.48xlarge	B200	179 GB	8 × 400G	1
ml.p6-b300.48xlarge*	B300	268 GB	16 × 400G	2

- PyTorch 2.10, NCCL 2.29.3, aws-ofi-nccl 1.18; one sm_90 + sm_100 image
- tok/s over all ranks; MFU per GPU type; repeats within 0.12%

The obstacle: asymmetric EFA rails

GPU generations expose different numbers of EFA rails, and aws-ofi-nccl rejects mismatched counts at NCCL init:

```
NCCL WARN NET/OFI Unexpected number of remote rails for dev 2. Expected 4 but got 2
```

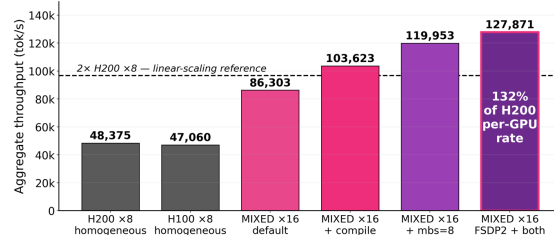
- Fix: force a common count. **OFI_NCCL_FORCE_NUM_RAILS** = 4 for Hopper only, 1 with any B200 (the minimum). The bare name is silently ignored.
- The rule is MATCHING counts:** Expand only when total per-GPU EFA bandwidth matches (Hopper: force 4); otherwise truncate to the minimum (any B200: force 1).

Recipe

	H100 + H200	H200 + B200
EFA rails	force 4	force 1
NCCL_PROTO	Simple	unset
Sharding	FULL_SHARD or HSDP	HSDP
Work per GPU	uniform	scaled to measured speed
To be Evaluated:	LocalSGD, DiLoCo	

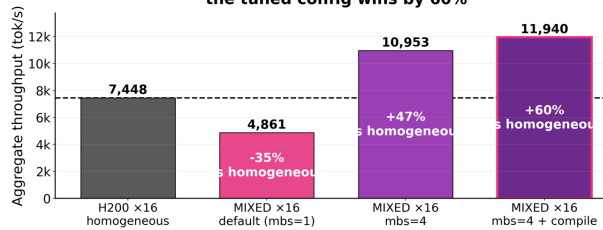
Same generation: H100 + H200

Llama 3 8B — mixed H100+H200 beats homogeneous, once configured

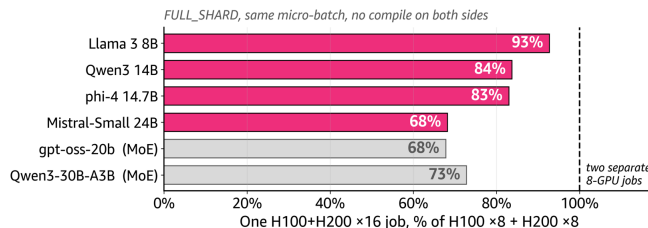


Llama 3 8B, 200 steps on C4. Tuned mixed 16-GPU beats homogeneous per GPU.

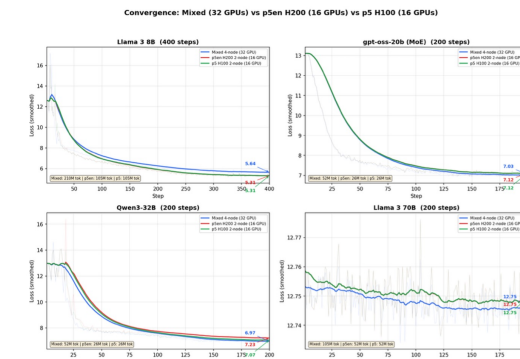
Llama 3 70B — the default config LOSES, the tuned config wins by 60%



Per-GPU Efficiency across 6 models, same recipe



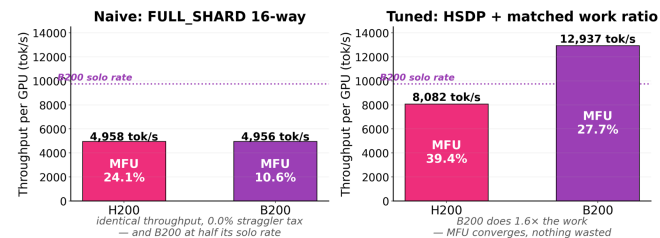
Model convergence not impacted



Smoothed loss at step 200 (52M tokens). Every mixed run is inside the homogeneous range.

Mixed generations: H200 + B200

H200 + B200: the metric that matters is per-architecture MFU



Large Models: Llama 3.1 405B

- For large models e.g. Llama 3 405B, heterogeneous training also gives customers a path to train models that don't fit within the memory constraints of a single instance type.
- Ran as one 16-way FSDP job across both p6-b200 and p6-b300

Nodes	GPUs	Result	tok/s (per GPU)
1 × p6-b200	8	OOM	—
1 × p6-b300	8	OOM	—
1 × p6-b200 + 1 × p6-b300	16	runs (151 GB/GPU)	864 (54)
both nodes, + compile, mbs 8	16	200 steps (144 GB/GPU)	4,320 (270)