Aristos: Pipelining One-sided Communication in Distributed Mixture of Experts (MoE)

Osayamen Jonathan Aimuyo[†]

oja7@cornell.edu [†] *Cornell Ann S. Bowers College of Computing and Information Science, Cornell University*

Cornell Bowers CIS Computer Science

Background

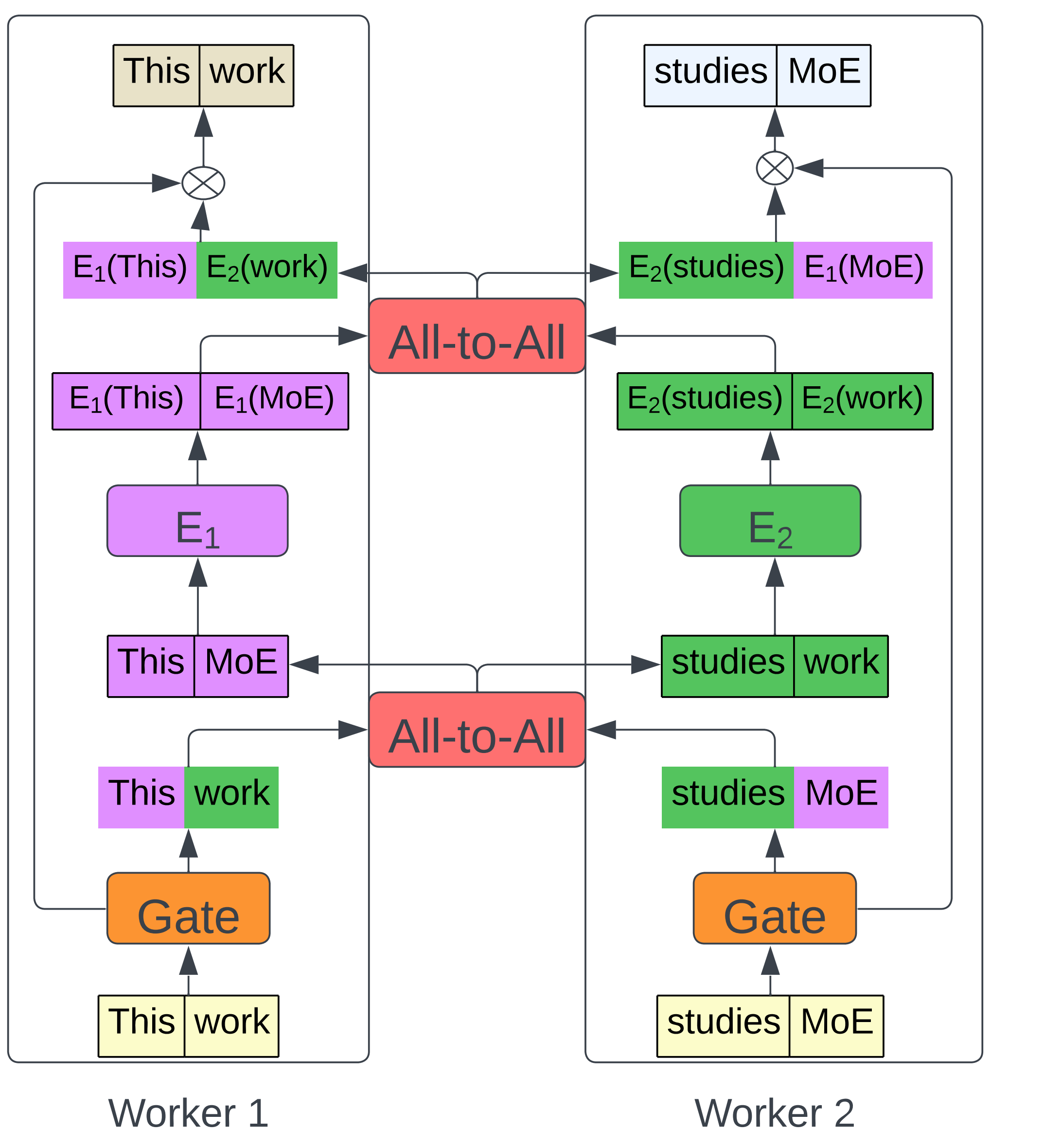
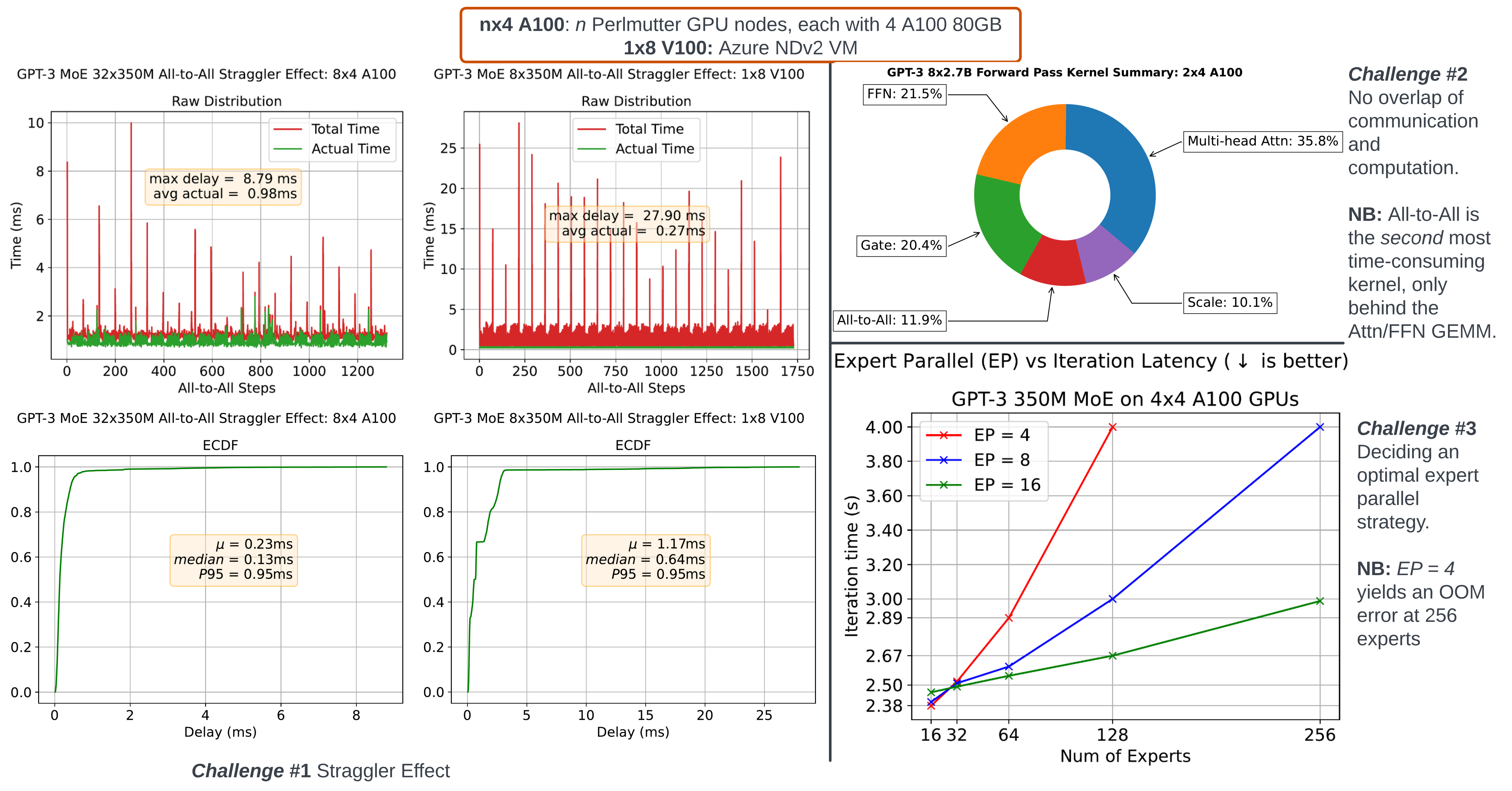


Figure 1: DMoE [4] with $W = EP = 2$. The **Gate** routes tokens to experts; **All-to-All** disseminates tokens; expert/**FFN** computation occurs; **All-to-All** reconsitutes tokens followed by the **Scale** computation.

Challenges

The widely-adopted [3] MoE architecture, promising **5x** faster training and **9x** reduced inference costs [6], is currently plagued by three *open* challenges [2, 5, 1] in the distributed setting.



Challenge #1 Straggler Effect

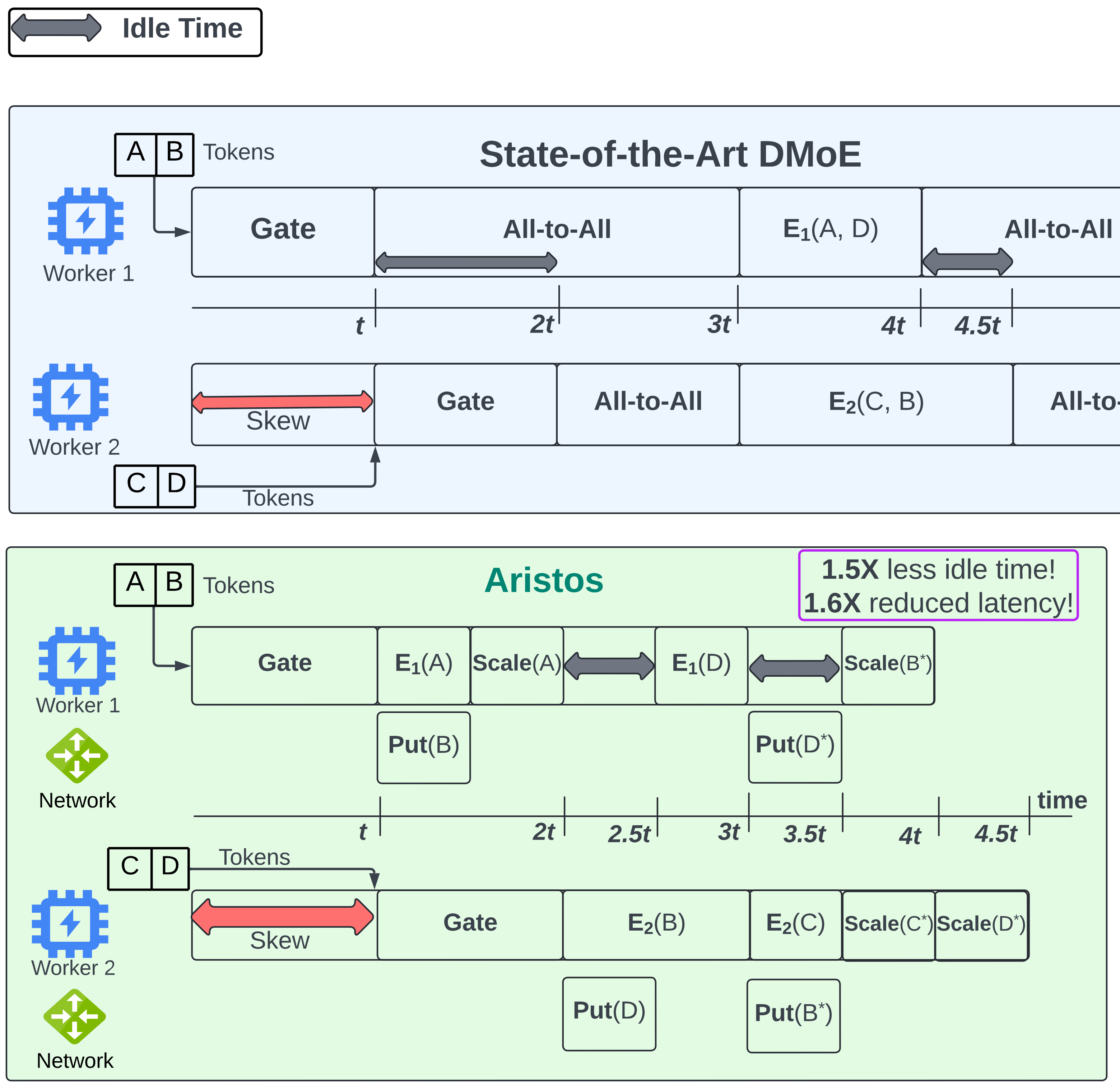
Challenge #2 No overlap of communication and computation.

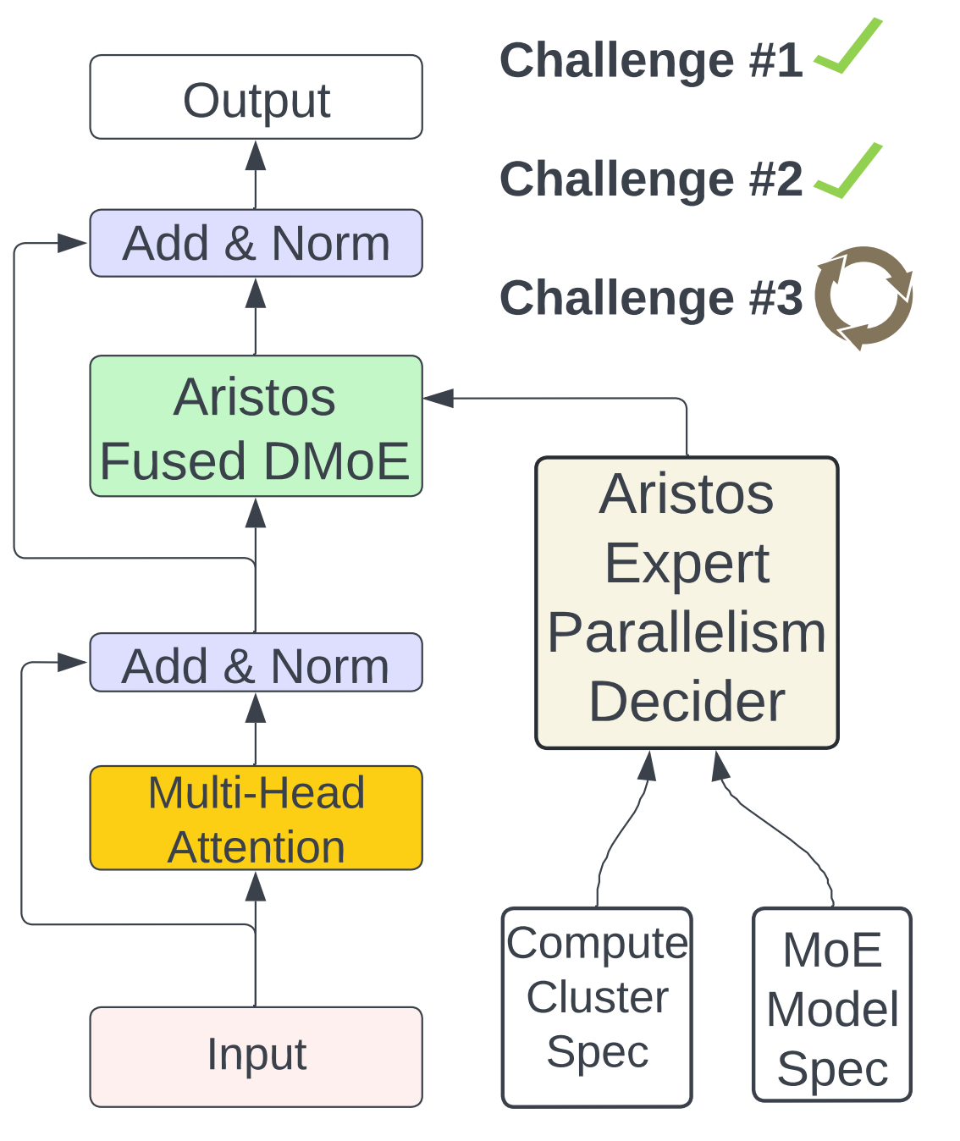
NB: All-to-All is the *second* most time-consuming kernel, only behind the Attn/FFN GEMM.

Challenge #3 Deciding an optimal expert parallel strategy.

NB: *EP = 4* yields an OOM error at 256 experts

Method





System architecture of Aristos

Ongoing Work

Define $G = (V, E)$ as the cluster topology, where V denotes devices and E communication links. Equation 1 formulates the heart of challenge #3: finding $G^* = \{g = (V_g,\ E_g) \mid g \subseteq G\}$ the set of optimal expert parallel groups or the *topology-aware sharding specification.*

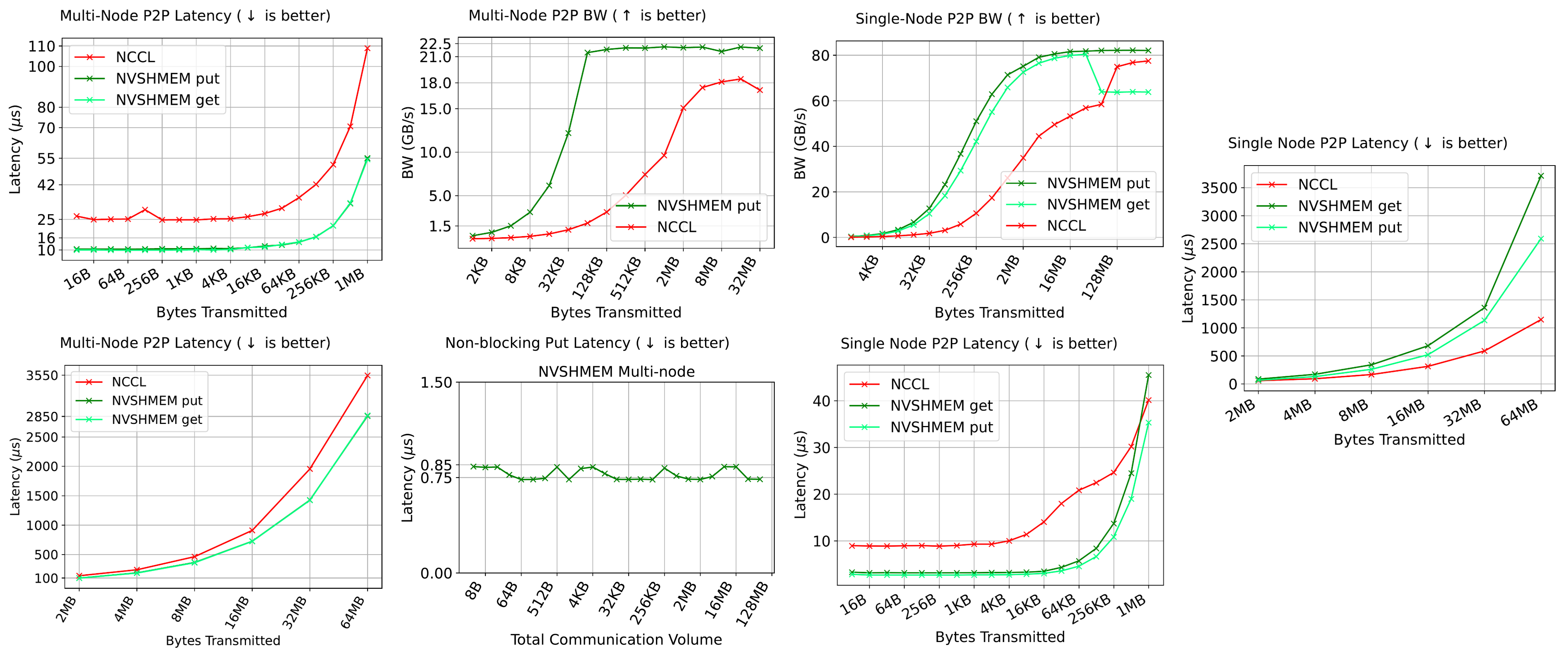
$$\min \max_{g \in G^*} \kappa \left(\pi(g) + \max_{i \in V_g} \mathcal{C}_i \right) + T_\rho(|G^*|) \quad (1)$$

subject to,

$$\sum_{j \in V_g} m_j \geq |\mathcal{X}| \qquad \forall g \in G^* \quad (2)$$

where, $\mathcal{X}$ is the set of all experts and m_j is expert memory capacity for device j. Also, κ identifies the frequency of MoE computation, $\pi(g)$ is the compute cost of g, C_i denotes the communication cost of device i and $T_\rho(|G^*|)$ is the cost of inter-group all-reduce on MoE parameters due to data parallelism. We also note that CUDA development for Aristos Fused is underway.

Microbenchmarks



Acknowledgements

We thank Dr. Rachee Singh for her guidance; Dr. Guila Guidi for Perlmutter access under award DDR-ERCAP0027296 of the National Energy Research Scientific Computing Center (NERSC); and Julian Bellavita for invigorating discussions.

References

[1] Fedus et al. “Switch Transformers”. In: *JMLR* 23 (2022).

[2] DeepSpeed. *Communication Logging*. 2024.

[3] GeminiTeam et al. *Gemini 1.5*. 2024.

[4] Lepikhin et al. “GShard”. In: *ICLR ’21*.

[5] Liu et al. “Janus”. In: *SIGCOMM ’23*.

[6] Rajbhandari et al. “DeepSpeed-MoE”. In: *ICML ’22*.