Accelerating MoE Model Inference with Expert Sharding*

* In the Proceedings of EuroMLSys '25

Oana Balmau†, Anne-Marie Kermarrec‡, Rafael Pires‡, André Loureiro Espírito Santo‡, Martijn de Vos‡, Milos Vujasinovic‡

† DISCS Lab, McGill University, Canada. ‡ SaCS Lab, EPFL, Switzerland. Correspondence: <first name>.<last name>@epfl.ch

Motivation

Mixture-Of-Experts (MoEs)

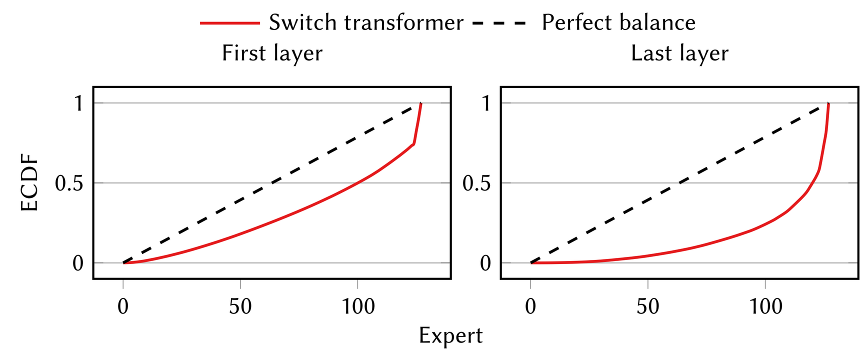
- Extremely large models with sparse per-token activation (DeepSeek V2.5 [1]: only 21B of 236B parameters active)
- Independent expert selection for each token
- Too large to fit on a single GPU

Require optimization of expert placement across GPUs

Expert parallelism

- Widely adopted in practice
- Each GPU holds a subset of experts
- GPU workload scales with the popularity of held experts
- Severe imbalance can lead to token dropping

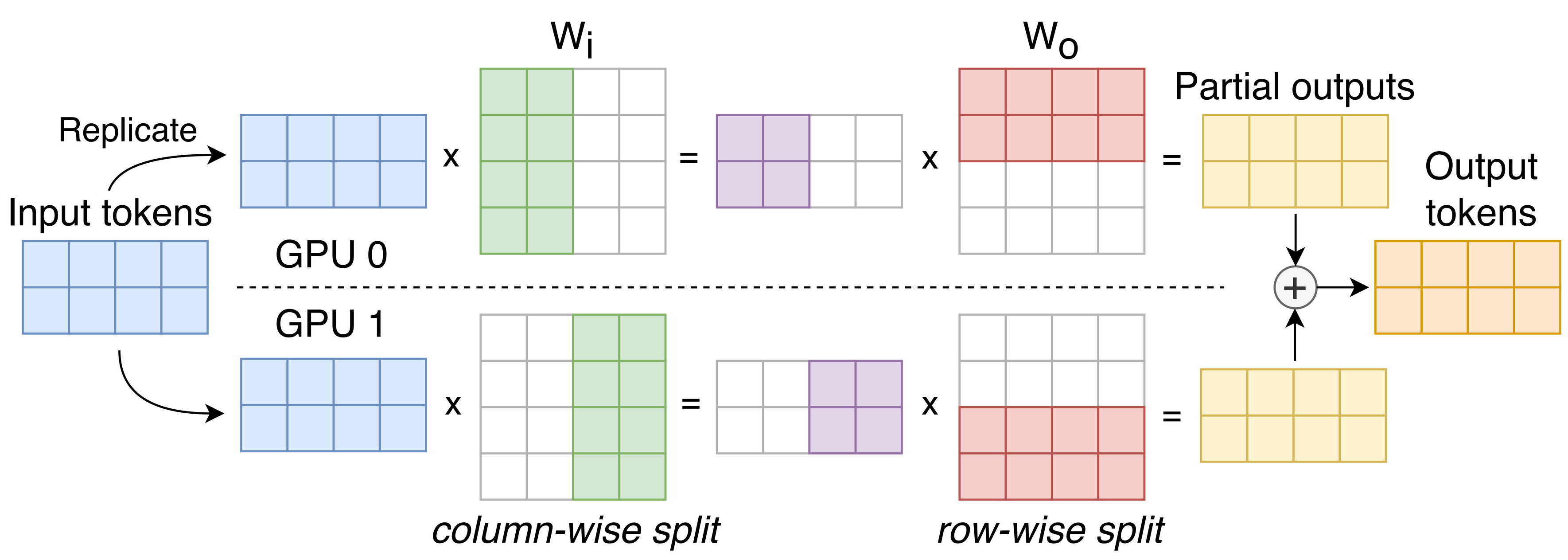
Expert selection imbalance



Expert selection is often imbalanced

Our solution

MoEShard (our solution)



Sharding expert matrices instead of assigning whole experts to individual GPUs

Setup

- Each expert consists of two fully connected layers (matrices W_i and W_o)
- Single computational node interconnected via high-speed, high-throughput links
- All GPUs have equal capacity, and the collective memory fits the entire model

Detailed algorithm

- Each GPU stores an equal share of W_i columns and W_o rows for each expert
- All non-expert layers (including routers) are fully replicated on each GPU
- **Expert computation with sharding:**
  - Inputs are replicated across all GPUs
  - On each GPU, the local slice of expert matrices multiplies the replicated inputs
  - Partial outputs are aggregated via an all-reduce operation across GPUs
- Optimized multiplications within GPU via Block Sparse Matrix Multiplication (MegaBlocks [2])
- Achieves perfect load balancing without token dropping

Existing solutions

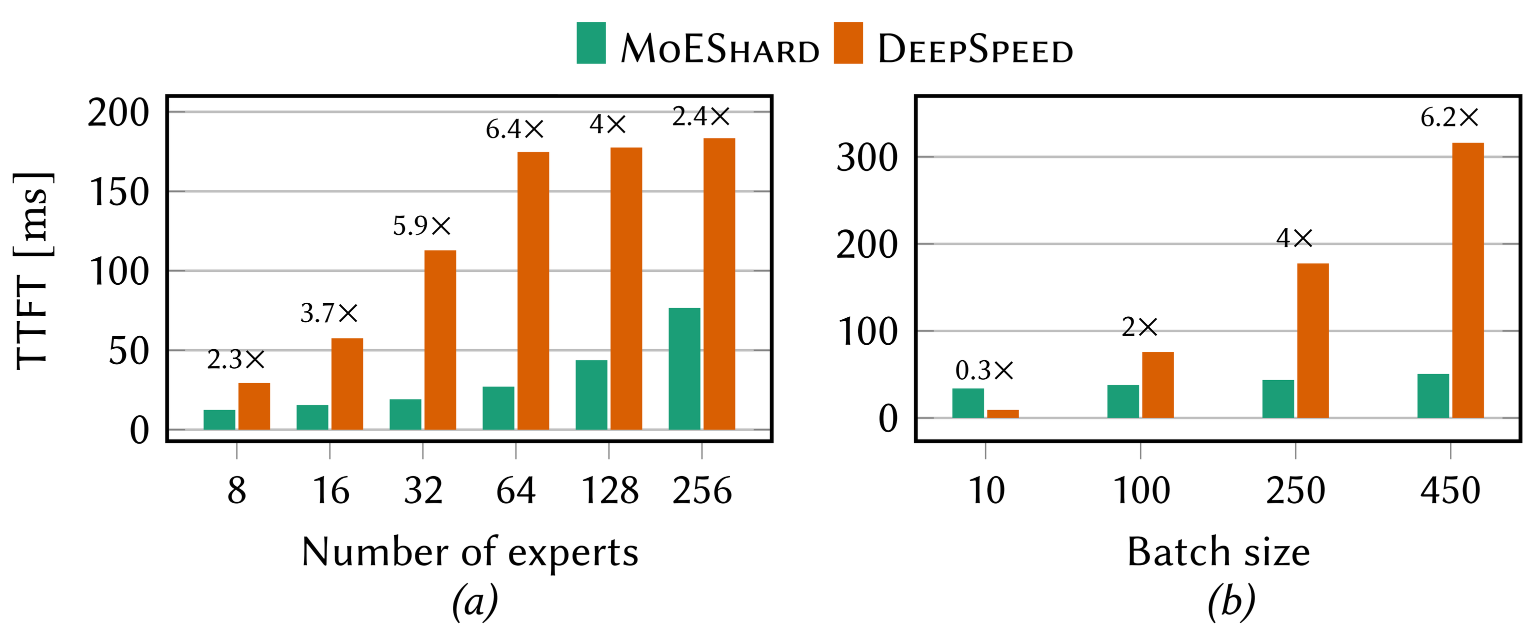
- DEEPSPEED - expert parallelism; optimized kernels
- TUTEL - expert parallelism; dynamic parallelism
- LAZARUS - expert parallelism; replication of frequently used experts
- PROPHET - expert parallelism; uses load balancing placement model
- LINA - expert parallelism; expert profiling and selection predicting
- EXFLOW - expert parallelism; expert assignment based on inter-layer affinity

Evaluation

Experimental Setting

- **Model:** SwitchTransformer-Base [3] (8 to 256 experts)
- **Dataset:** BookCorpus
- **Metric:** Time to First Token (TTFT) – duration of Prefill stage
- **Baseline:** DeepSpeed [4] using expert parallelism with capacity factor $\min(|E|, 50)$
- **Batch Size:** 250 when varying the number of experts
- **Experts:** 128 when varying batch size
- A custom router is employed to induce skew in expert selection

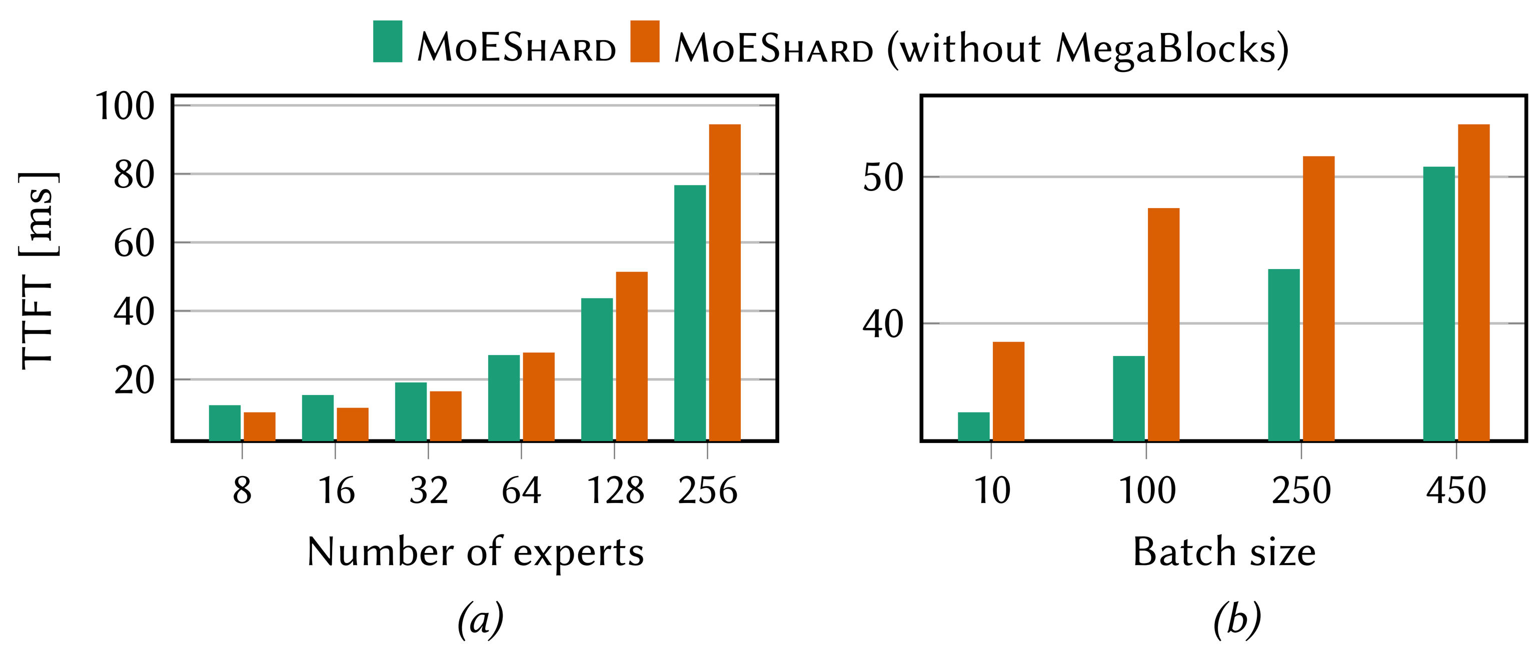
Comparison Against Baseline



Bar labels indicate the speedup of MoESHARD w.r.t. DeepSpeed.

- MoESHARD consistently outperforms the baseline until DeepSpeed begins dropping tokens (for more than 50 experts)
- Performance benefits of MoESHARD even more pronounced with larger batch sizes

Ablation Study



- For small matrices, the overhead of launching MegaBlocks outweighs its benefits
- MegaBlocks becomes advantageous when the number of experts exceeds 64

[1] DeepSeek-AI. *DeepSeek-V2: A Strong, Economical, and Efficient Mixture-of-Experts Language Model*. 2024. arXiv: 2405.04434 [cs.CL].

[2] Trevor Gale et al. "Megablocks: Efficient sparse training with mixture-of-experts". In: *Proceedings of Machine Learning and Systems* 5 (2023), pp. 288–304.

[3] William Fedus, Barret Zoph, and Noam Shazeer. "Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity". In: *Journal of Machine Learning Research* 23.120 (2022), pp. 1–39.

[4] Samyam Rajbhandari et al. "Deepspeed-moe: Advancing mixture-of-experts inference and training to power next-generation ai scale". In: *International conference on machine learning*. PMLR. 2022, pp. 18332–18346.