MoNDE: Mixture of Near-Data Experts for Large-Scale Sparse Models

Taehyun Kim[1,2], Kwanseok Choi[1], Youngmock Cho[1,2], Jaehoon Cho[1], Hyuk-Jae Lee[1,2], Jaewoong Sim[1]

[1]Seoul National University [2]Inter-University Semiconductor Research Center

① Introduction

- Large-scale Mixture-of-Experts (MoE) models offer fixed-complexity computation but has **massive memory capacity requirements** that are out of reach for commodity GPU settings
- Existing solutions are highly **bottlenecked by communication over PCIe**
- We present a near-data processing solution designed on emerging CXL memory devices to resolve communication overhead in MoE inference

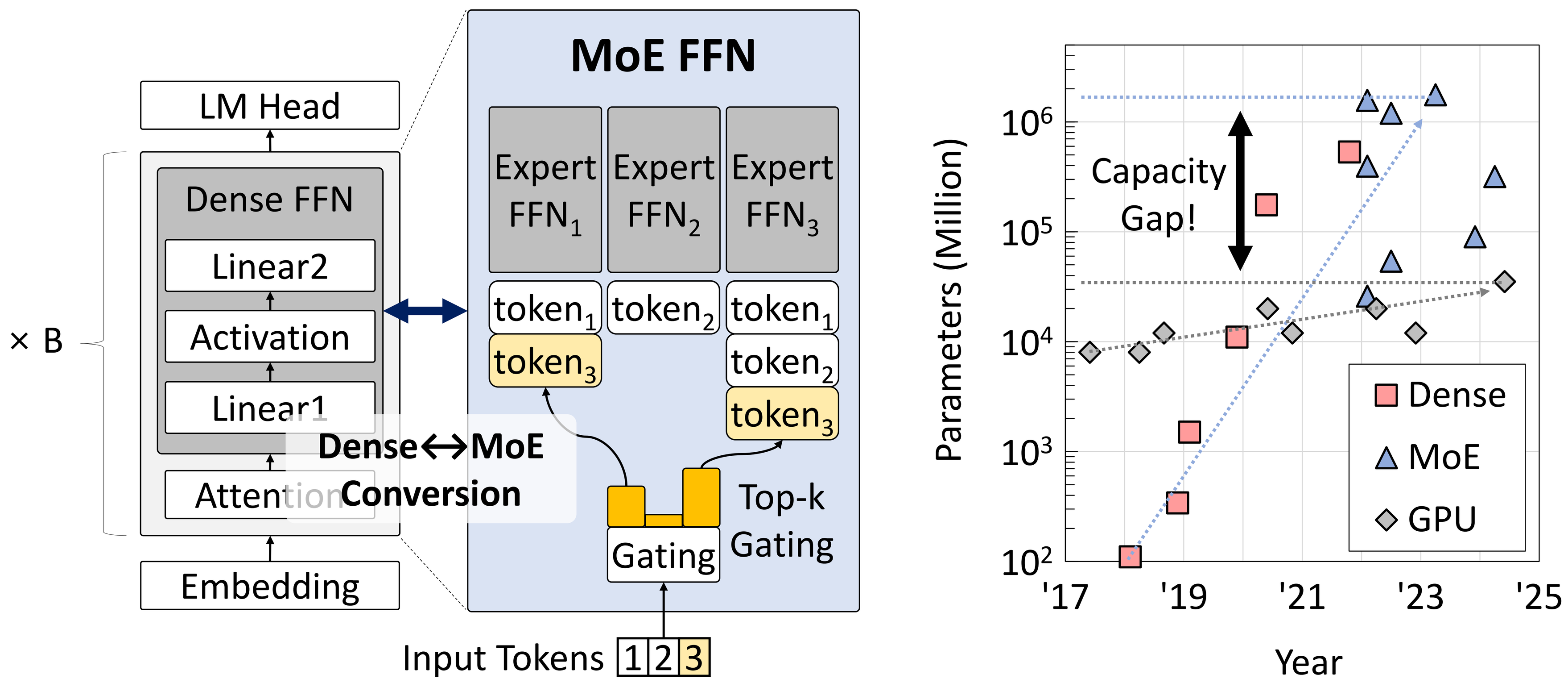


Figure 1. MoE Overview (left) and large language model scaling trend (right)

② Background & Motivation

- **Memory Cost Analysis (Single FFN layer)**
  - Linear scaling to E
  - Quadratic scaling to d_m (∵ e.g., $d_{ff} = 4 \cdot d_m$)

	Non-MoE	MoE
Computation	$4 \cdot d_m \cdot d_{ff} \cdot T$	$4 \cdot k \cdot d_m \cdot d_{ff} \cdot T$ (k×)
Mem. Access	$2 \cdot d_m \cdot d_{ff}$	$2 \cdot k \cdot d_m \cdot d_{ff}$ (k×)
Mem. Capacity	$2 \cdot d_m \cdot d_{ff}$	$2 \cdot E \cdot d_m \cdot d_{ff}$ **(E×)**

※ Input tokens **T**, experts **E**, embd. dims **d_m&d_{ff}**, top-k routing

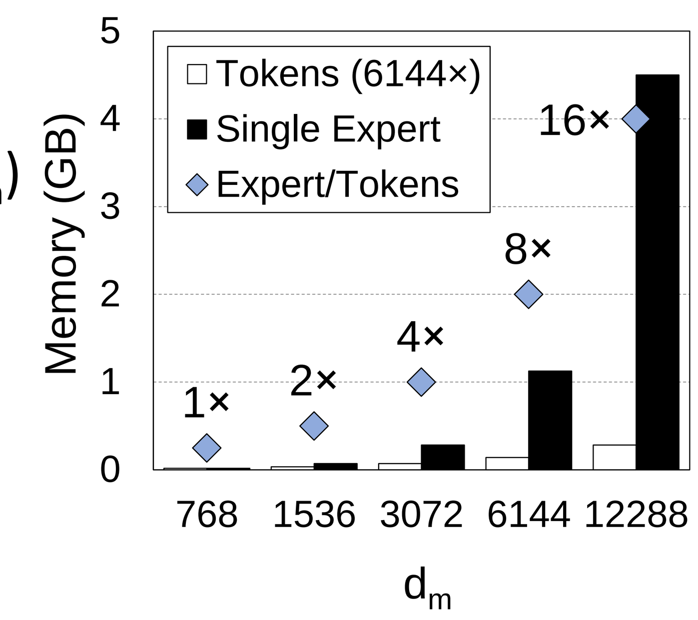


Figure 2. Cost formulation (left) and scaling trend comparison of a single expert and input token (right)

- **Existing Solutions**
  - Expert parallelism (resource-inefficient)
  - Expert offloading (PCIe bottleneck)

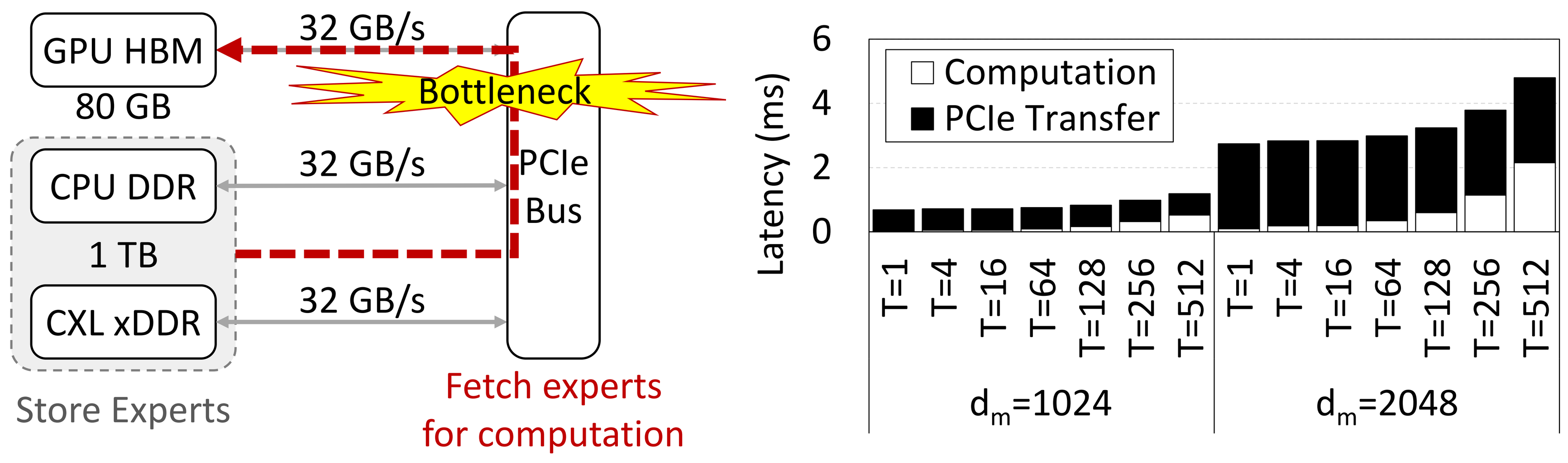


Figure 3. PCIe bottleneck in expert offloading (left) and latency comparison of computation and PCIe transfer of a single expert (right)

- **Expert Skew**
  - Unbalanced token distribution to experts of an MoE layer
  - A compute-to-memory ratio gap exist between the popular (hot) experts and the remaining (cold) experts

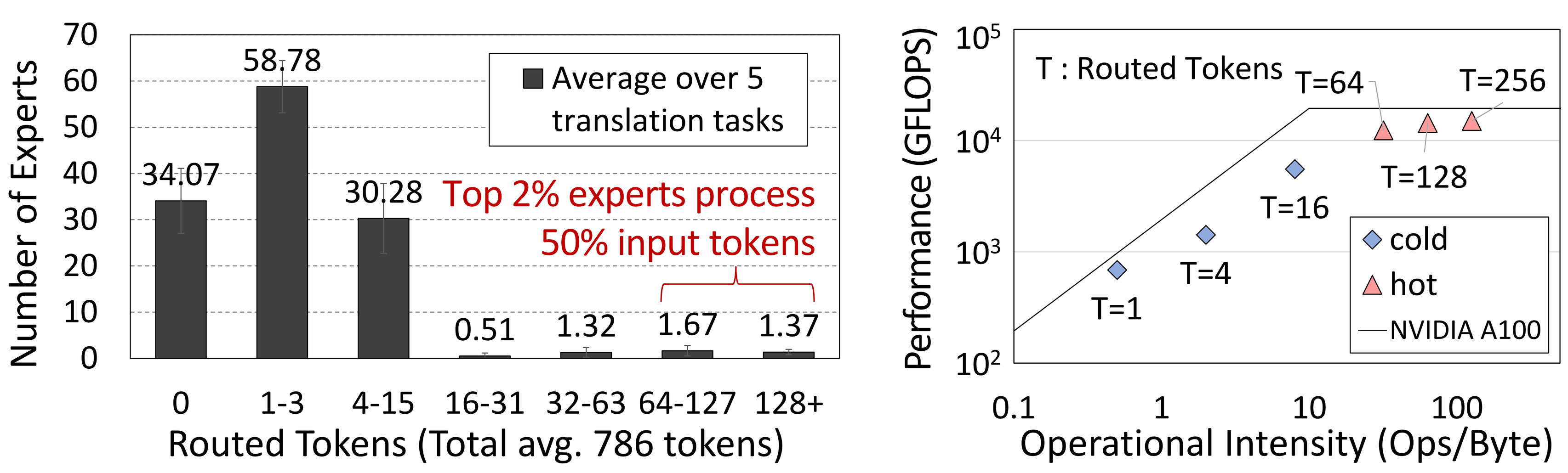


Figure 4. Histogram of NLLB-MoE expert routing for ranges of token counts (left) and roofline model wrt routed tokens (right)

Key insights
- Emerging CXL memory device technology offer **large add-in memory capacity (and bandwidth)**
- **Token data are much smaller** and easier to move across devices
- Cold experts can run with **comparable performance on weaker compute** with sufficiently large mem bandwidth (mem-bound)

③ Mixture of Near-Data Experts

- **Near-data processor (NDP) on CXL memory device : PMove → AMove**
- **GPU-MoNDE Load-balancing : Run GPU & MoNDE in parallel**
  - Run the hottest *H* experts on the GPU
  - Find *H* such that expert movement latencies for *MoNDE→GPU ⓐ* and *MoNDE memory→NDP Core ⓑ* are equalized

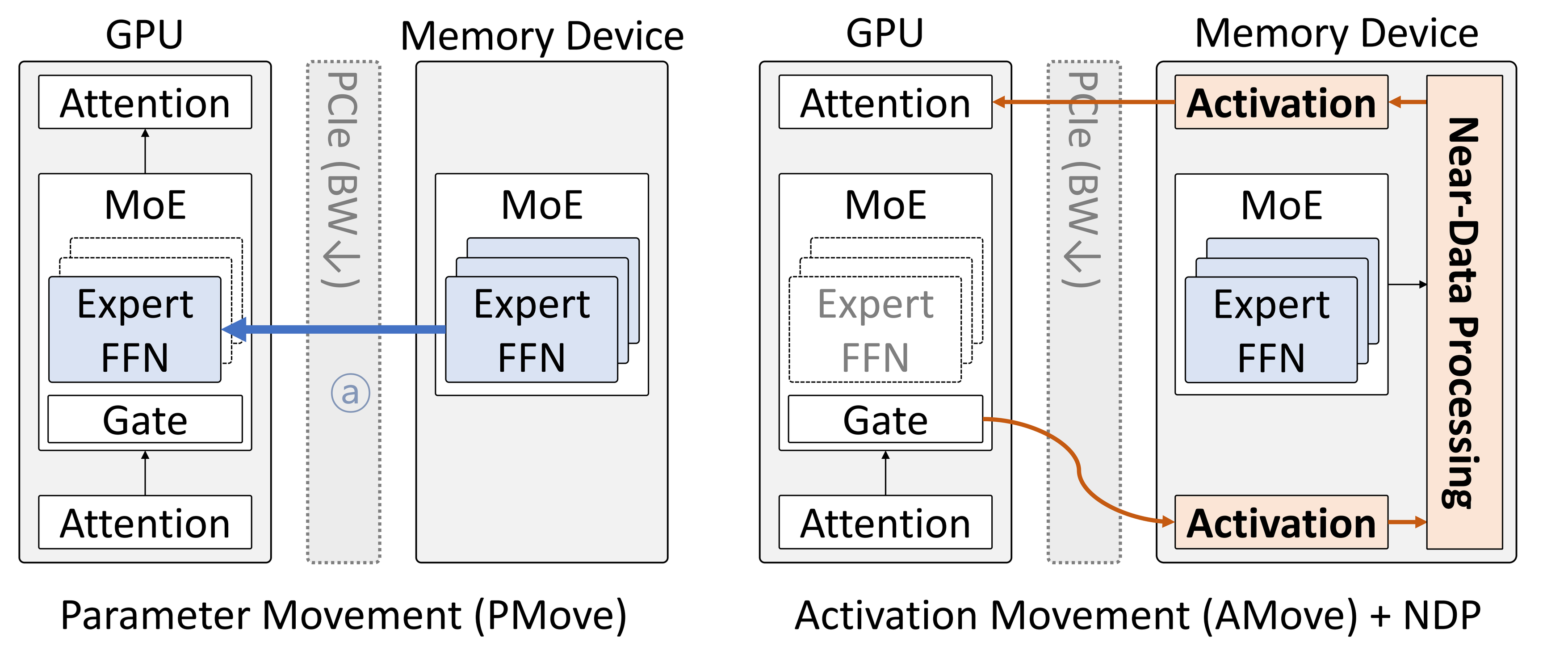
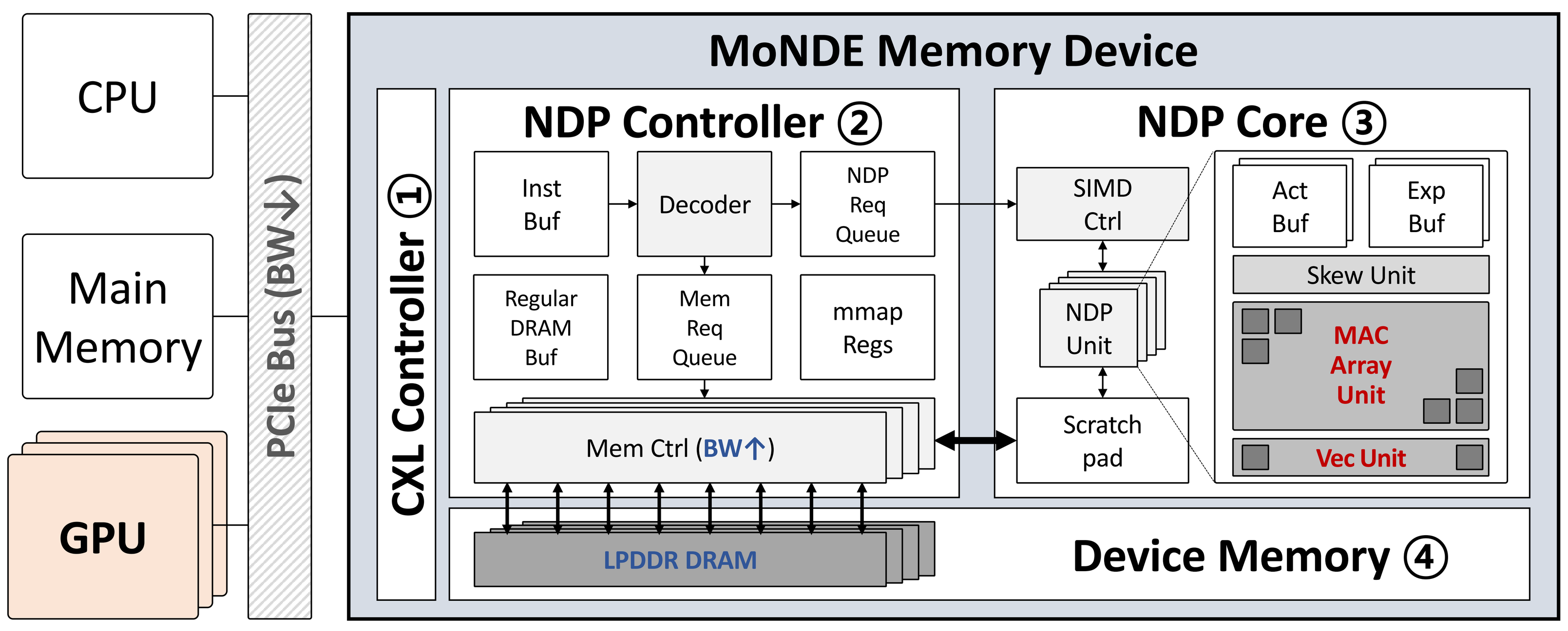


Figure 5. Parameter and Activation Movement (PMove & AMove)

CXL & NDP Controllers ①②
- Control decoupling + P2P comm.

NDP Core ③
- Expert matmuls & activation (e.g., ReLU)



Device Memory ④
- High capacity & bandwidth memory enabled by die bonding and stacking

Figure 6. MoNDE device architecture overview

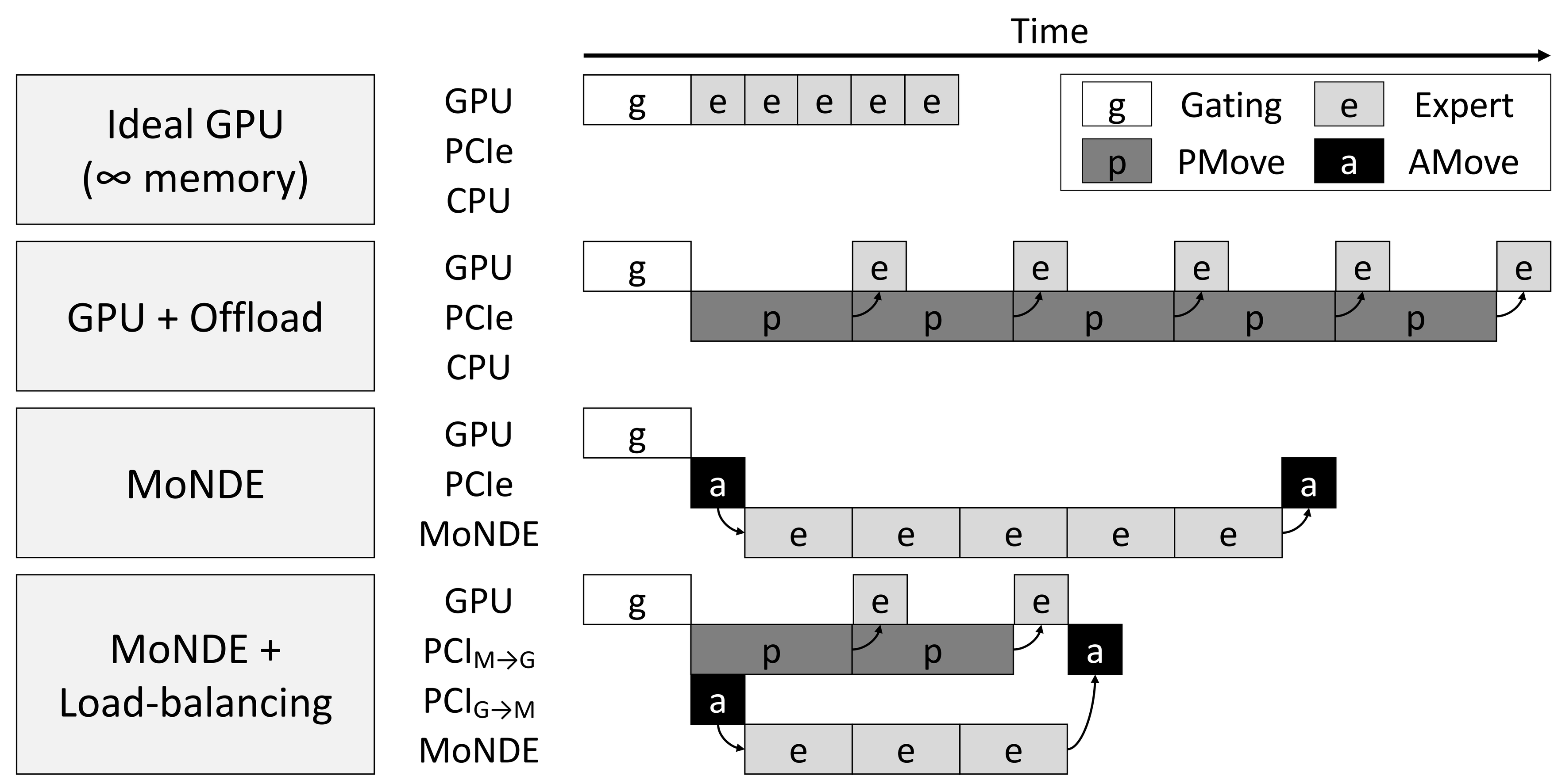


Figure 7. MoNDE execution flow

④ Evaluation and Conclusion

System	NVIDIA A100	MoNDE
BF16 Compute	312 TOPS	2 TOPS
Memory	80 GB 2 TB/s	512 GB 512 GB/s
PCIe	PCIe gen4 32 GB/s	

Model	Switch Transformers (Top-1 gating)	NLLB-MoE (Top-2 gating)
Experts	128 × 24 layers (each 8.4 MB)	128 × 12 layers (each 33.6 MB)
Model Size (Dense/MoE)	1.1 GB / 51.5 GB	5.7 GB / 103.1 GB

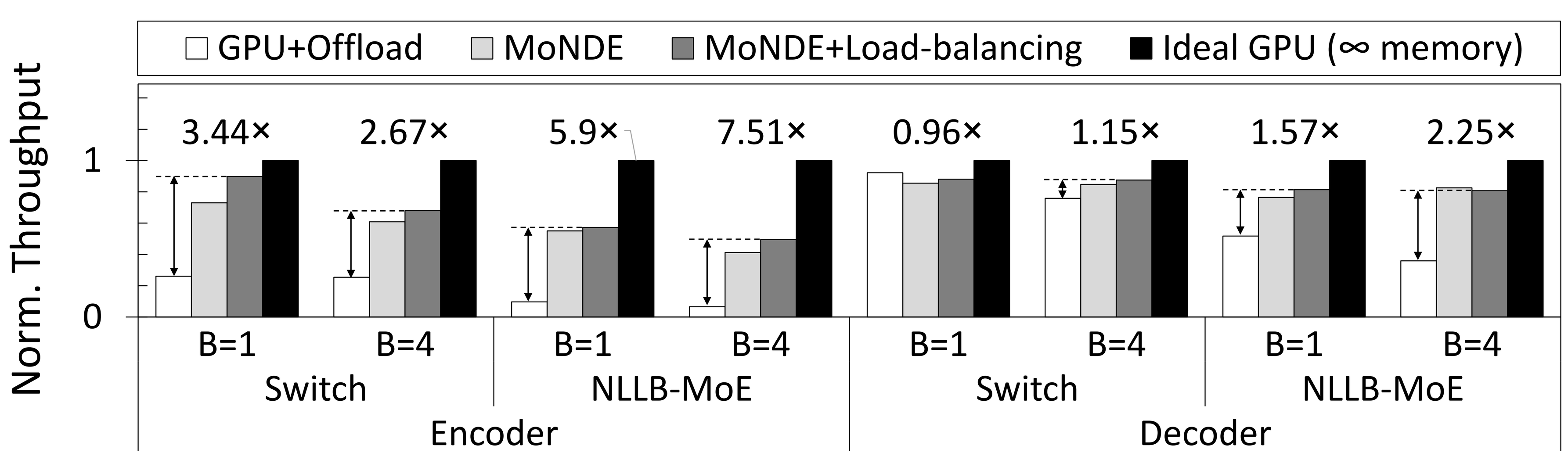


Figure 8. MoNDE throughput evaluation settings (top) and results (bottom)

- Achieves 4.9× and 1.5× speedup for the encoder and decoder ops
- Resolves both capacity shortage and communication overhead for MoE LLM inference

Slides *Author Bio.*