ClusterFusion: Expanding Operator Fusion Scope for LLM Inference via Cluster-Level Collective Primitive

Xinhao Luo, Zihan Liu, Yangjie Zhou, Shihan Fang, Ziyu Huang,
Yu Feng, Chen Zhang, Shixuan Sun, Zhenzhe Zheng, Jingwen Leng, Minyi Guo

SJTU, NUS, SQZ

2025.11.15

Outline

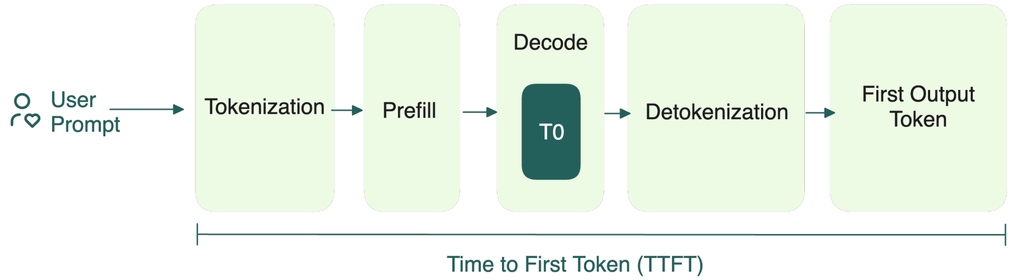
- Background
- Design
- Evaluation
- Discussion

Background

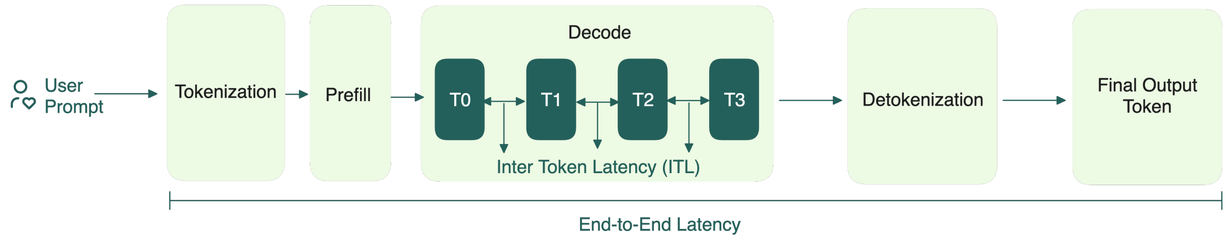
Decoding Bottleneck in LLM Inference

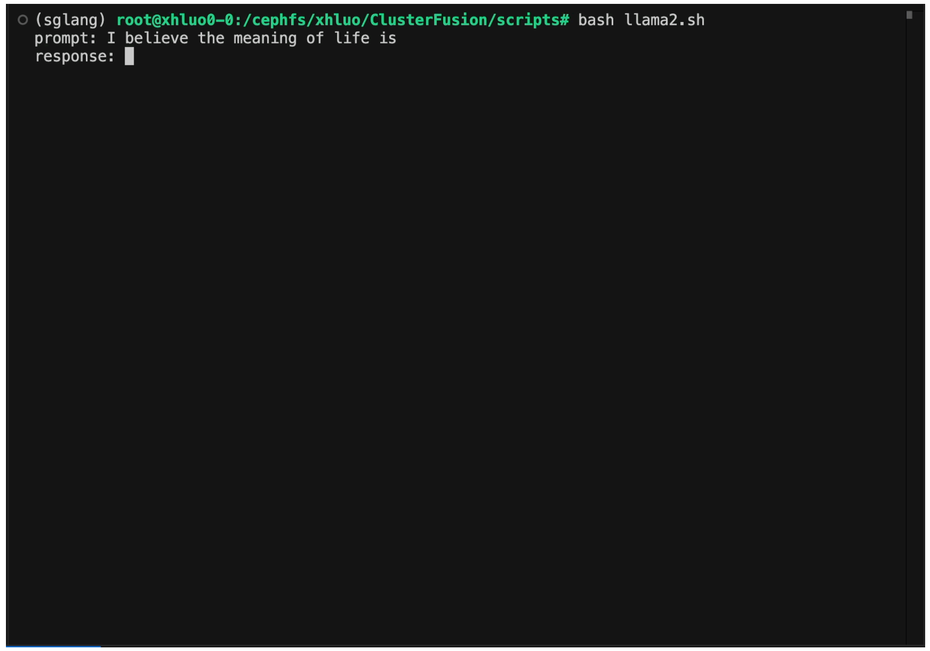
>95% Latency

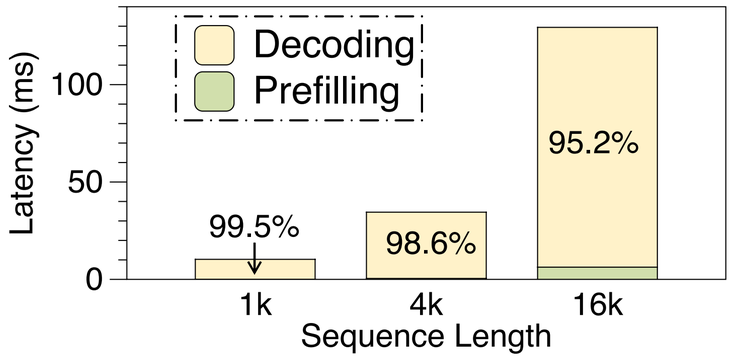
- Autoregressive
  - Prefilling



  - Decoding

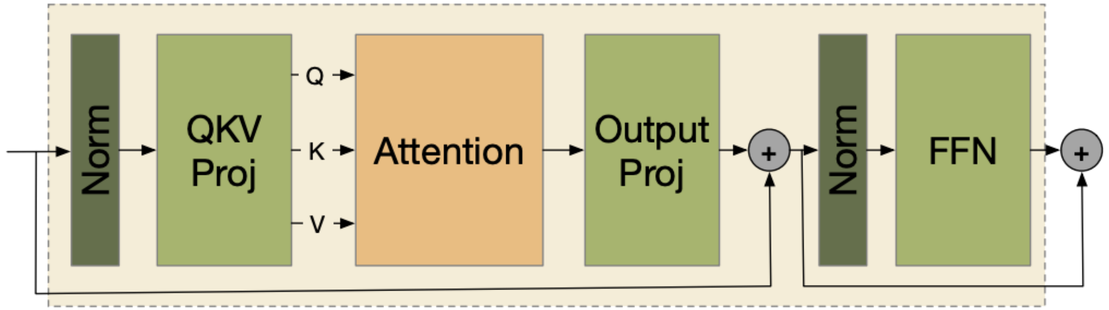




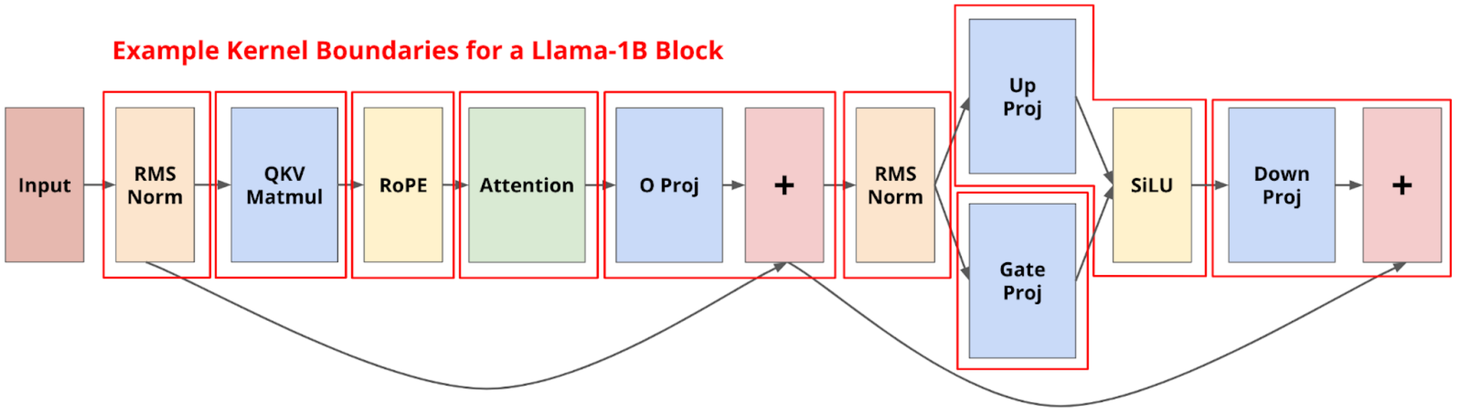


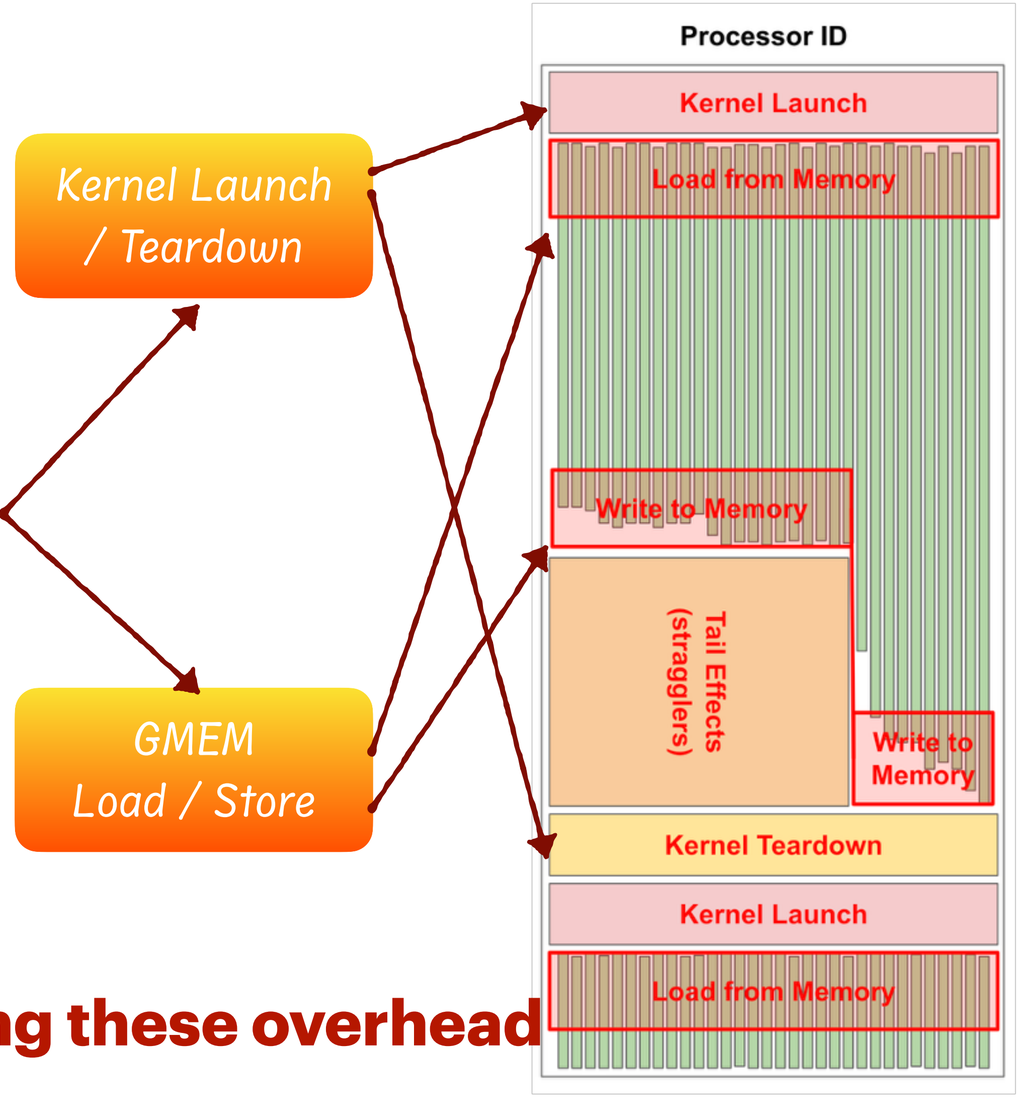
Fragmented Kernel Execution

Intermediate results via global memory & frequent kernel launches



Traditional kernel-by-kernel inference backend





Operator fusion is key to eliminating these overhead

https://hazyresearch.stanford.edu/blog/2025-05-27-no-bubbles

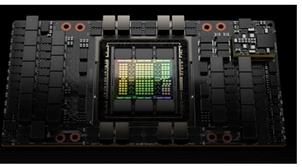
Challenges of Operator Fusion

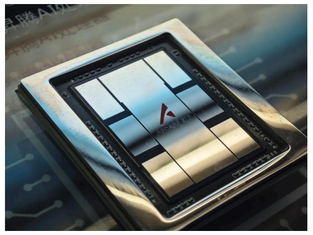
Diverse Backend Kernel Source

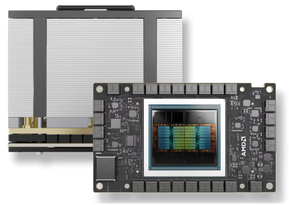
LLM Inference Frameworks

SGL vLLM TensorRT-LLM mlc-llm

.....







Backend Kernel Libraries

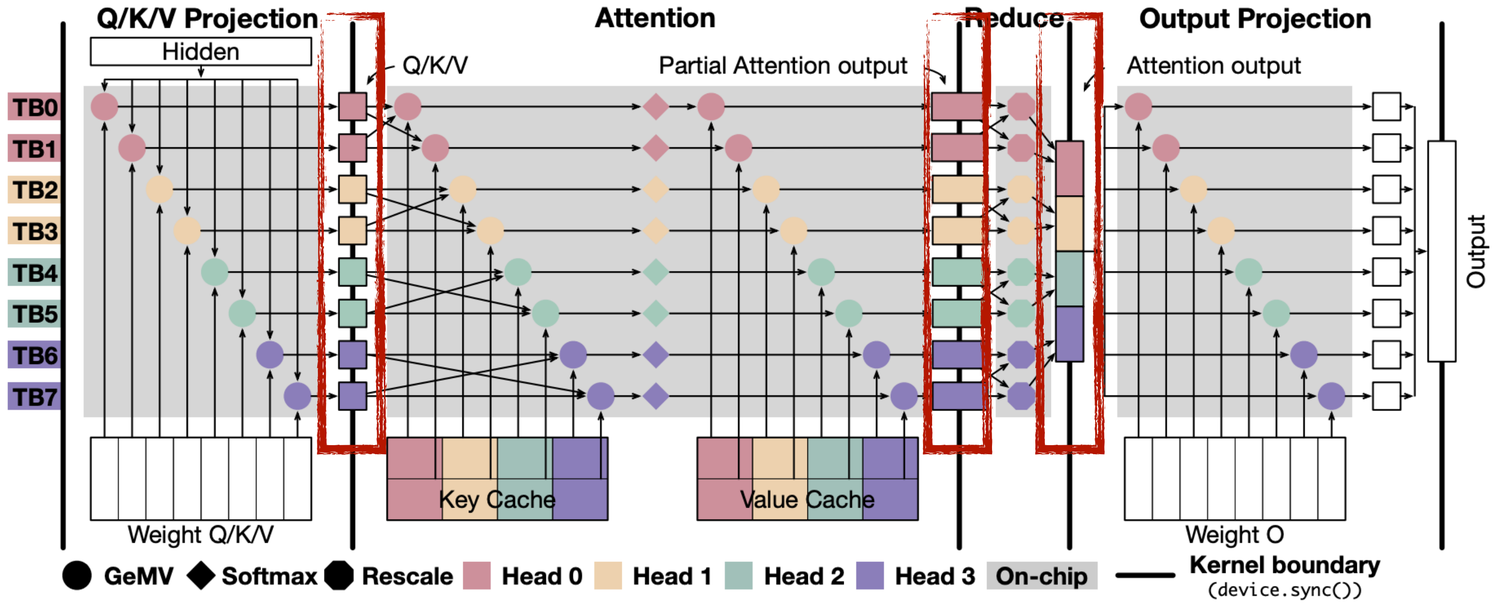
cuBLAS FlashInfer triton tvm

Diverse collection of specialized kernel implementation

Challenges of Operator Fusion

Cross-block data dependencies

- Linear Projection: GeMM (Column-wise partition and GMEM gather)
- Attention: Flash Decoding
- Output Projection: GeMM

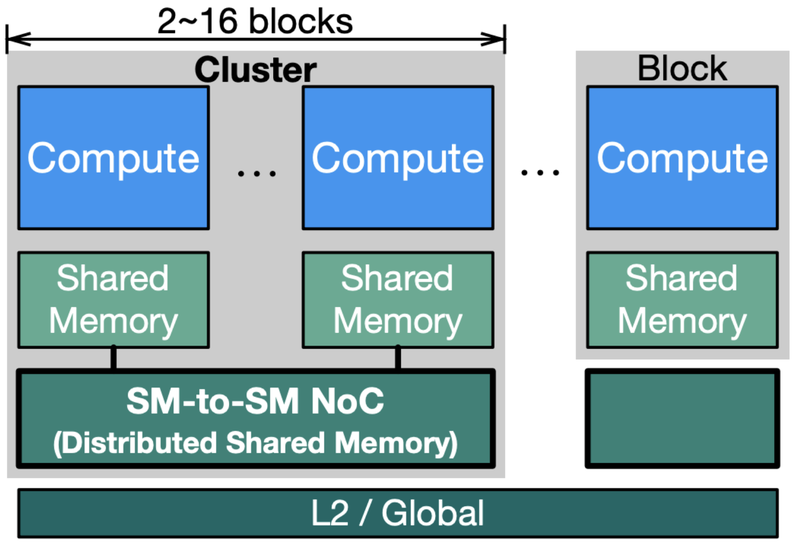


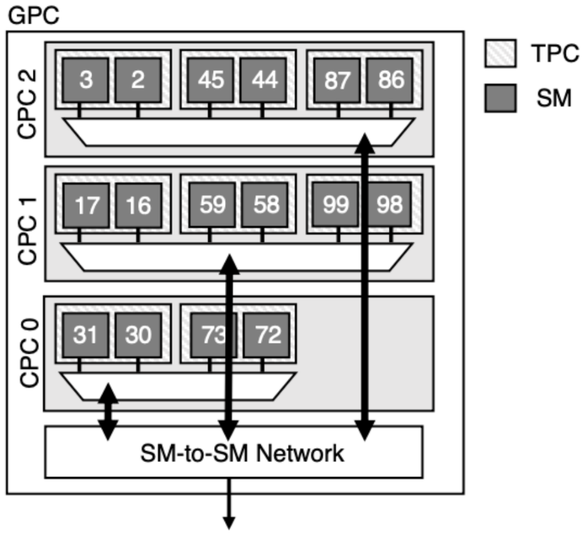
Cross-block data dependencies prevent further operator fusion

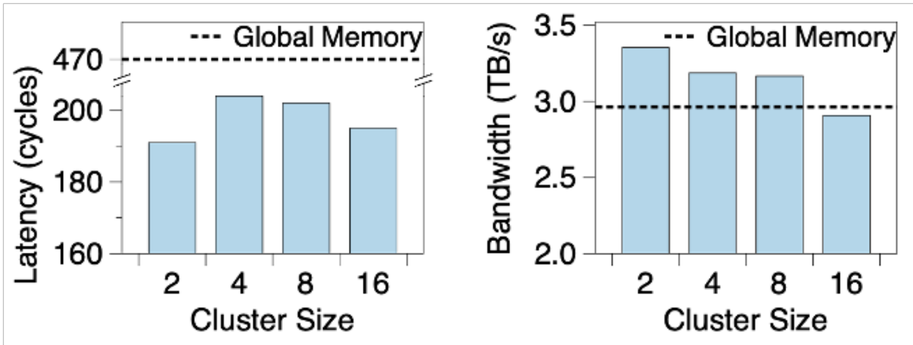
Design

New Interconnect: Distributed Shared Memory

DSM support direct, low-latency communication in a cluster.



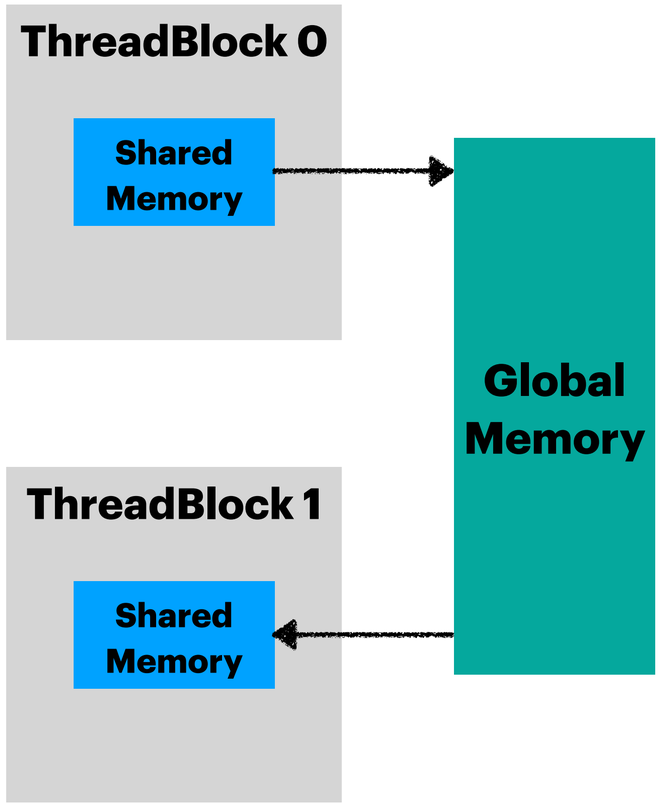





```
__global__ __cluster_size__(X,Y,Z) void compute_kernel(...) { ... }
__shared__ float buffer[1024];
remote_buffer = cluster.map_shared_rank(buffer, rank_in_cluster)
```

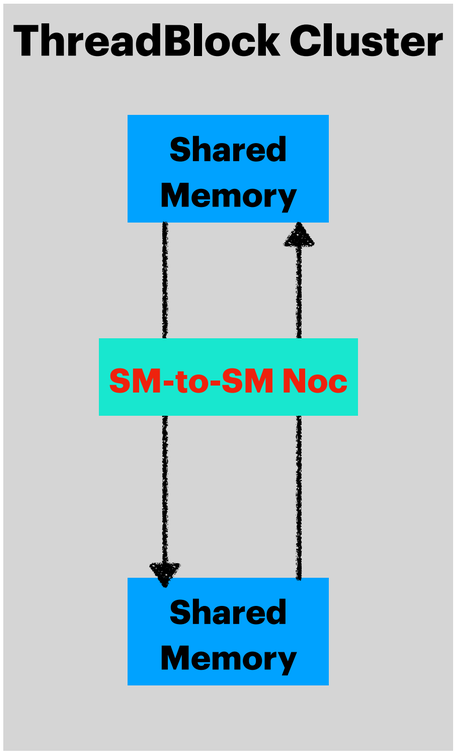
Core Insight

Cluster as a Unit



Shift from independent "Thread Blocks"

to collaborative "Thread Block Clusters"

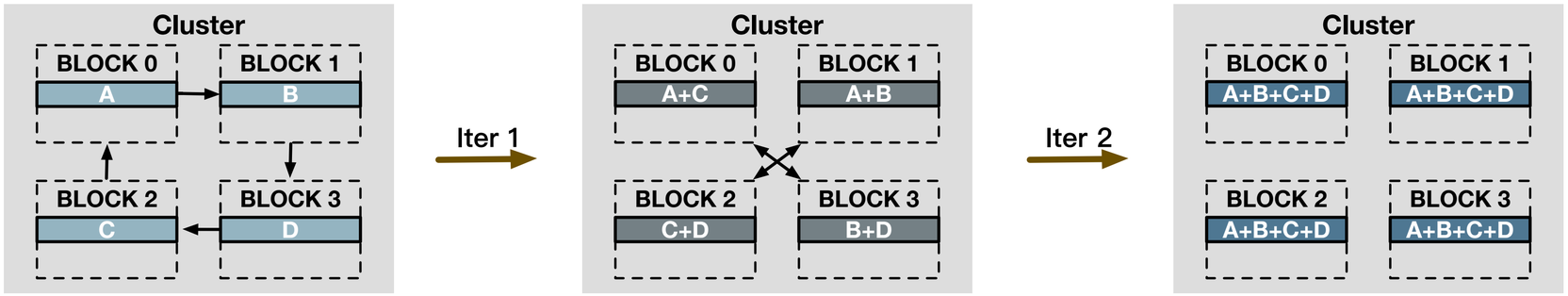


Our Goal: Expand fusion scope across multiple operators.

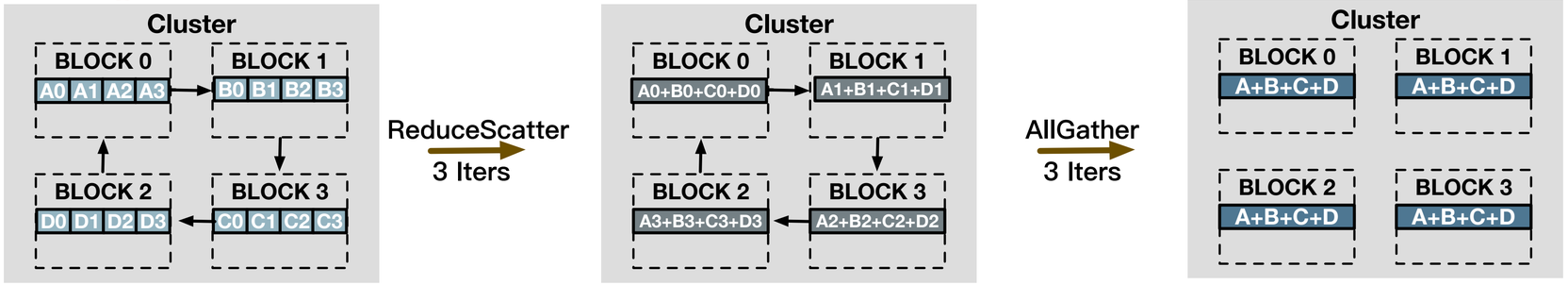
Cluster-Level Collective Primitives

Efficient on-chip reduction

Tree AllReduce:



Ring AllReduce:



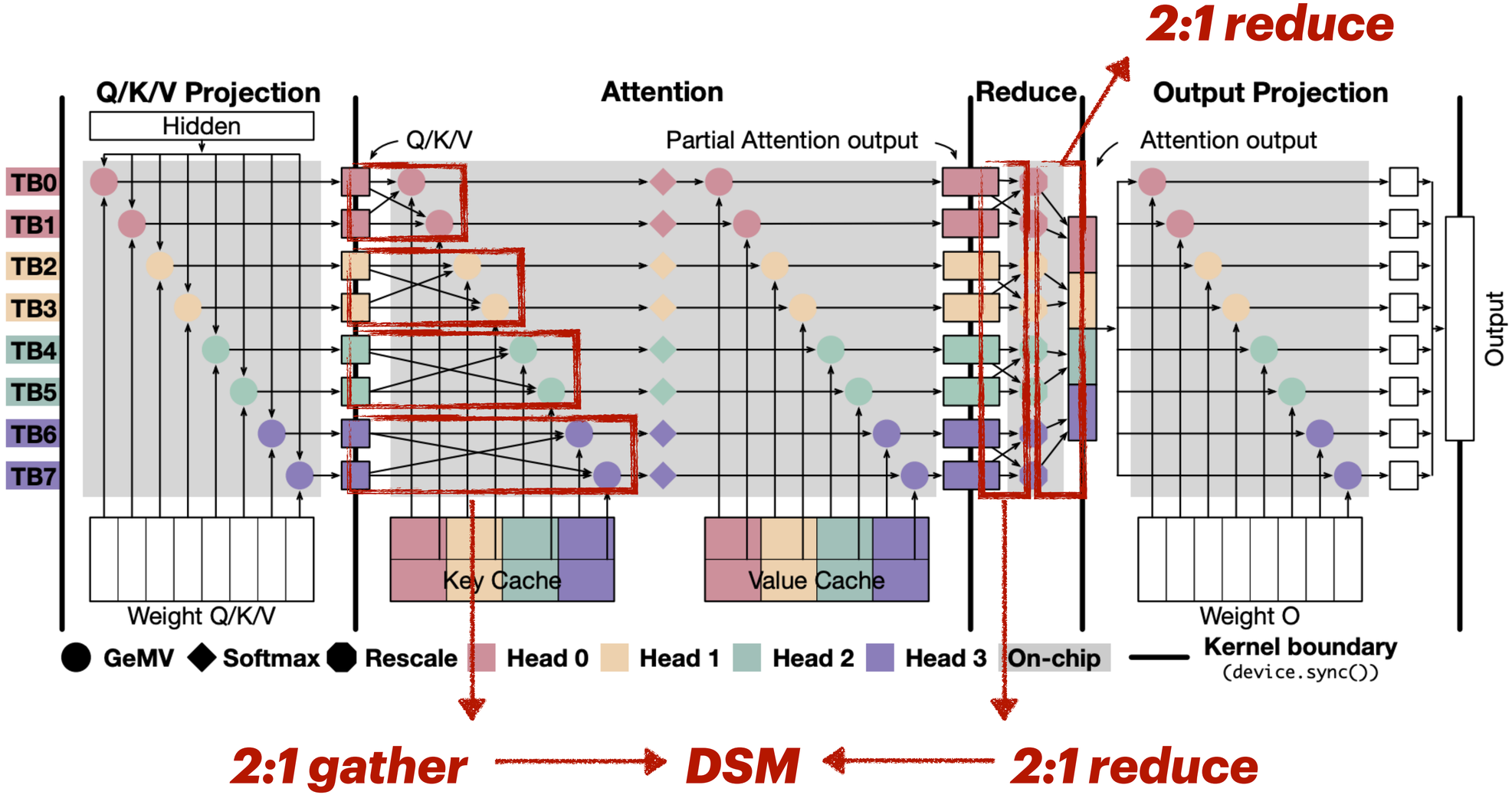
Pattern	Rounds	Comm volume per Round	Total comm volume
Ring	2(N-1)	**K / N**	**2(N - 1) * (K / N)**
Tree	**$\log_2(N)$**	K	K * $\log_2(N)$

Operation	Data Size (KB)	Off-chip (μs)	On-chip (μs)	Speedup
ClusterReduce	32	8.03	6.77	1.18×
	64	9.01	6.61	1.36×
	128	14.95	7.42	2.01×
	256	22.44	9.17	2.44×
ClusterGather	32	6.26	3.90	1.60×
	64	6.27	4.12	1.52×
	128	6.31	4.39	1.44×
	256	6.61	4.15	1.59×

- ***Pros: Lower communication rounds and latency for small data***
- ***Cons: Larger total communication volume***

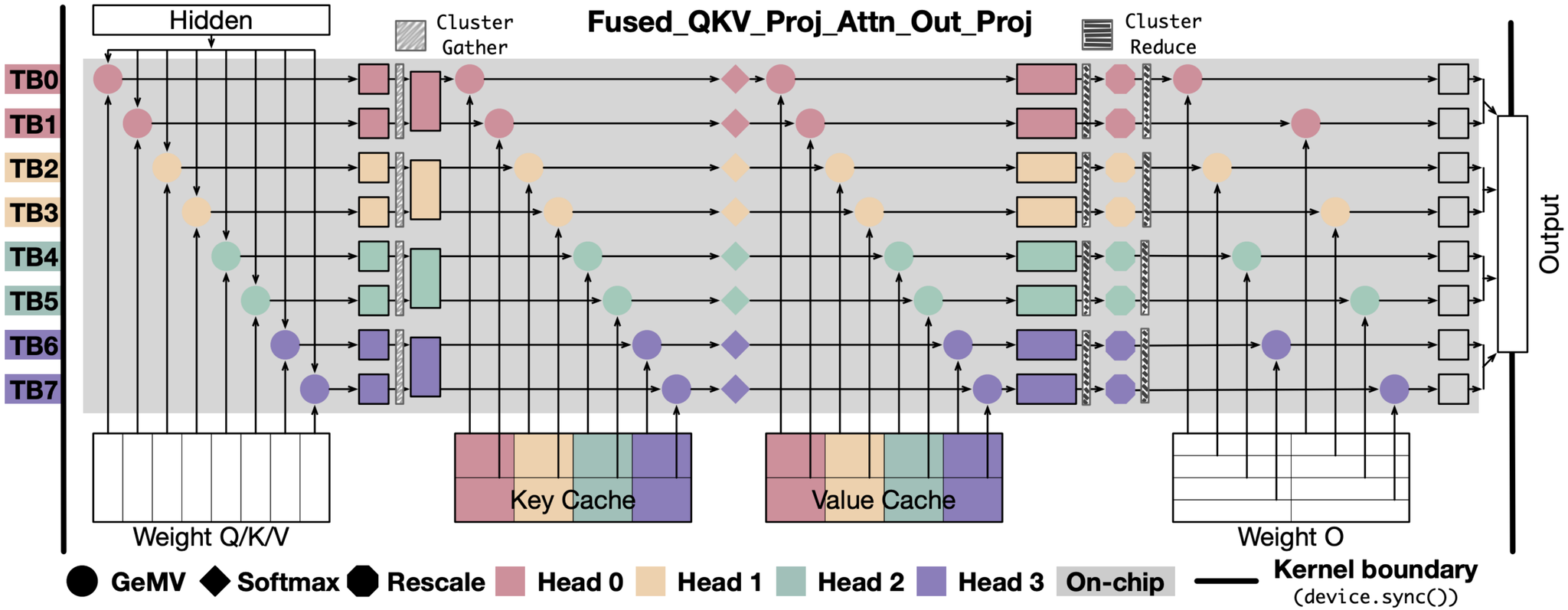
Cluster-Centric Dataflow

Analysis



Cluster-Centric Dataflow

Intermediate results stay on-chip

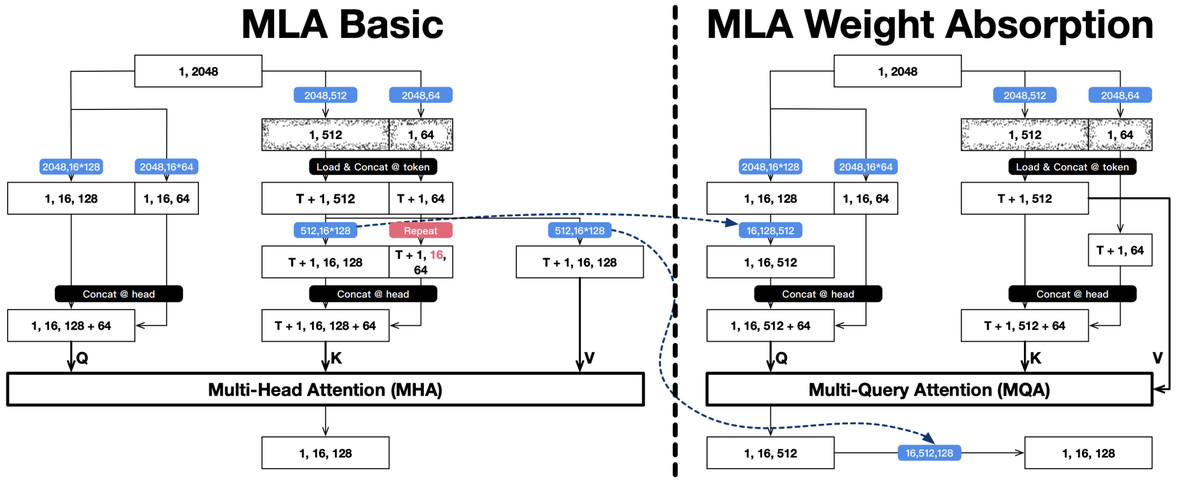


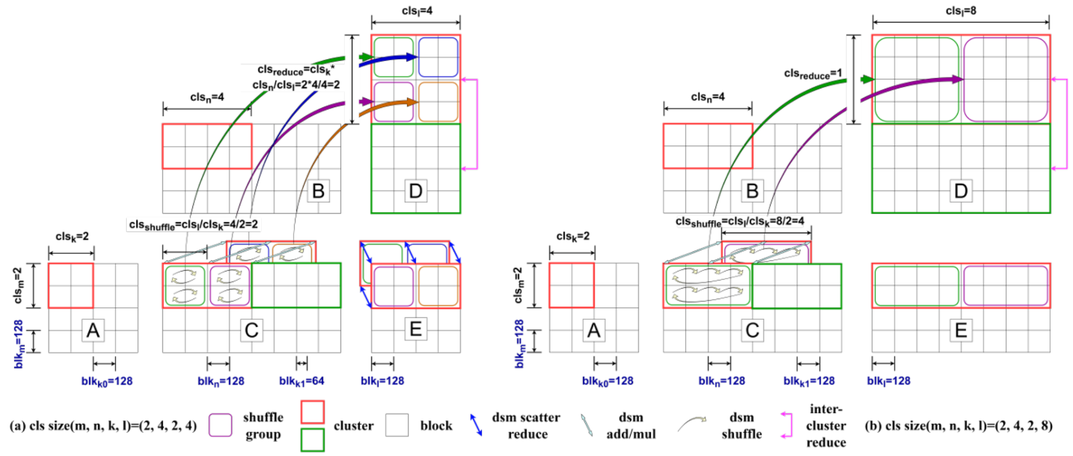
Blocks of one head get the whole head result with all-reduce via DSM

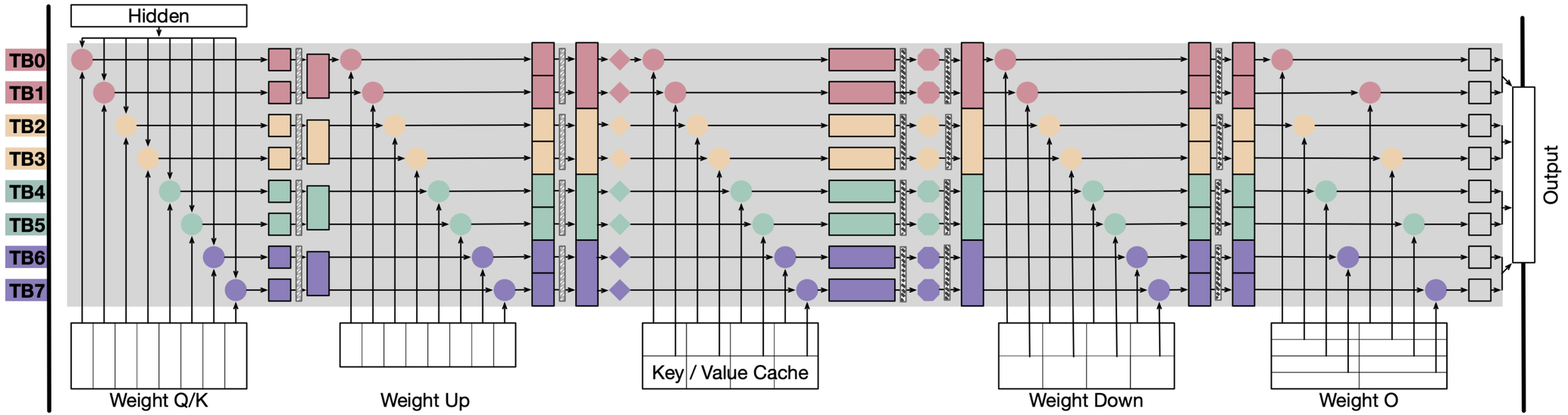
Compute partial GeMM and do the final all:1 reduce at Global

Generality: Beyond MHA

Also support +FFN, MLA, ..., via DSM Primitives







Evaluation

Evaluation

Integration & Implementation

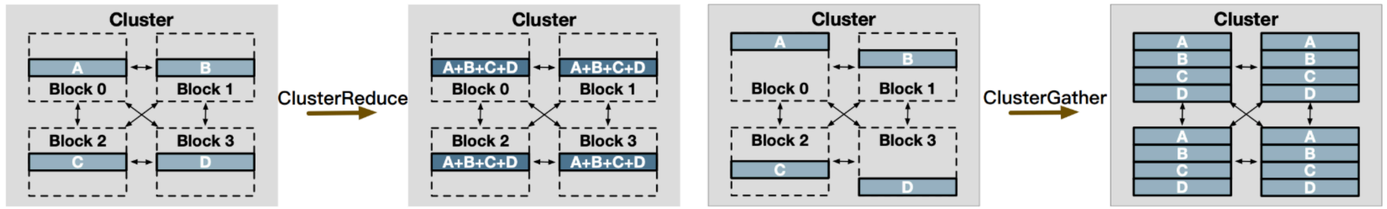
- ***We construct the cluster-level primitives with low-level PTX ISA.***

E2E

```
from clusterfusion import lama_decoder_layer

llama_decoder_layer(
    output,
    residual_output,
    hidden_states,
    residual,
    qkv_weight,
    o_weight,
    paged_kv_indptr_buf,
    paged_kv_indices_buf,
    k_data_ptrs,
    v_data_ptrs,
    layer_id,
    rms_weight,
    eps,
    positions,
    cos_sin
)
```

Primitives



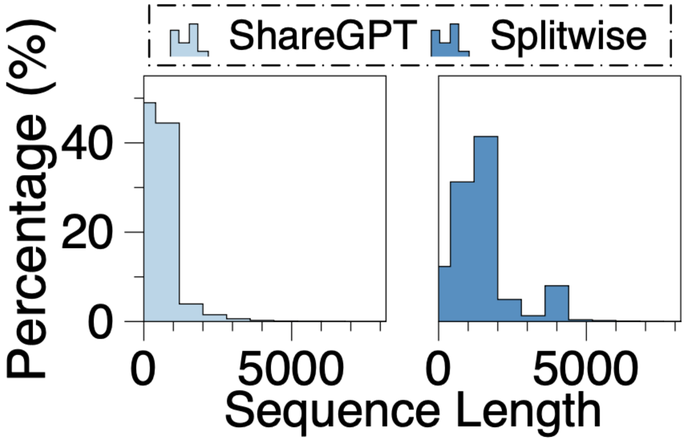

```
cluster_reduce<CLUSTER_SIZE, Stage::LINEAR>(
        size, tid, HEAD_DIM, cluster_block_id,
        src_addr, dst_addr, bar_ptr,
        neighbor_dst_bar, local_qkv, weight);
```

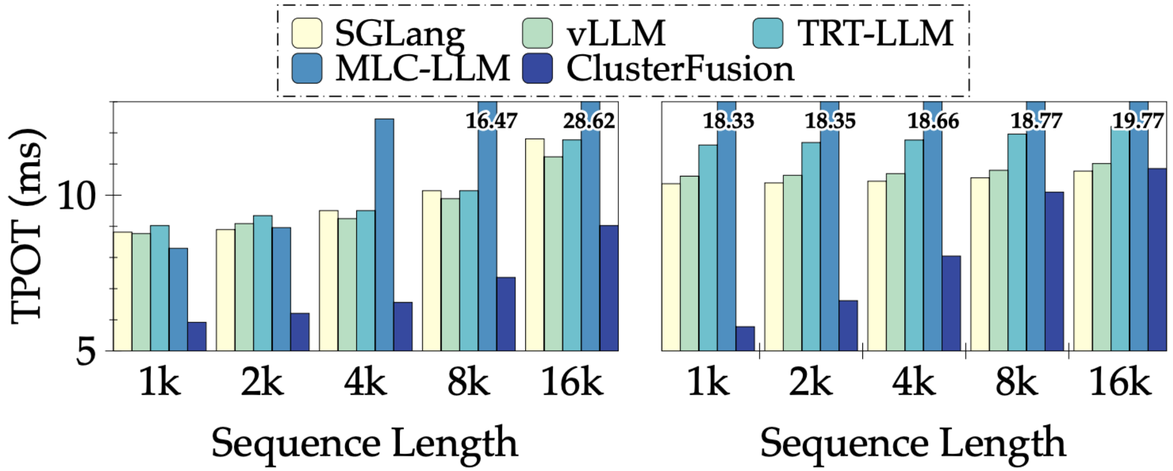
- ***Cluster-Centric Dataflow purely CUDA @ H100, now we support 5090.***

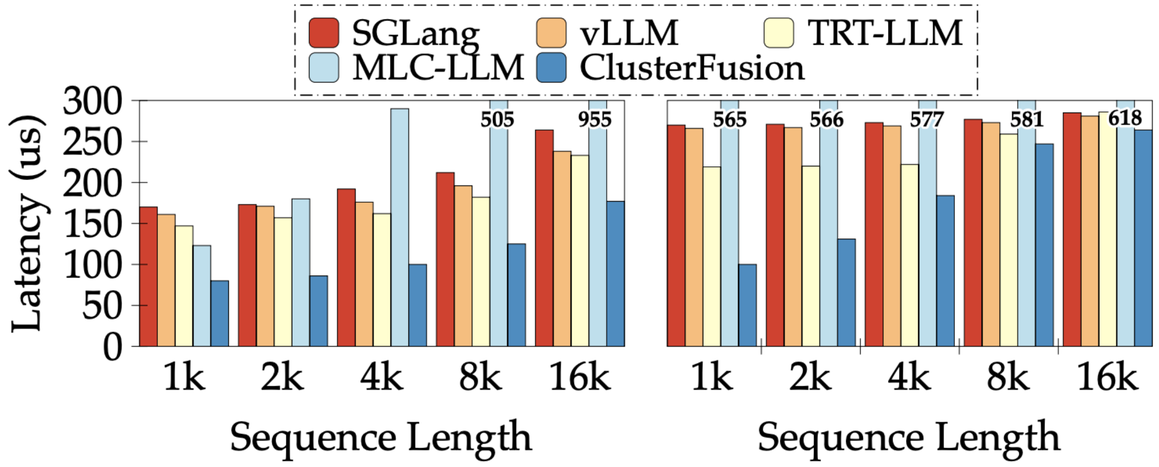
Evaluation

Setup & End-to-End Results

- Framework: SGLang, vLLM, TensorRT-LLM, MLC-LLM
- Backend: FlashInfer, CUTLASS, Triton
- Optimization: CUDA Graph, Torch.compile()
- Models: Llama2-7B, DeepSeek-V2-Lite



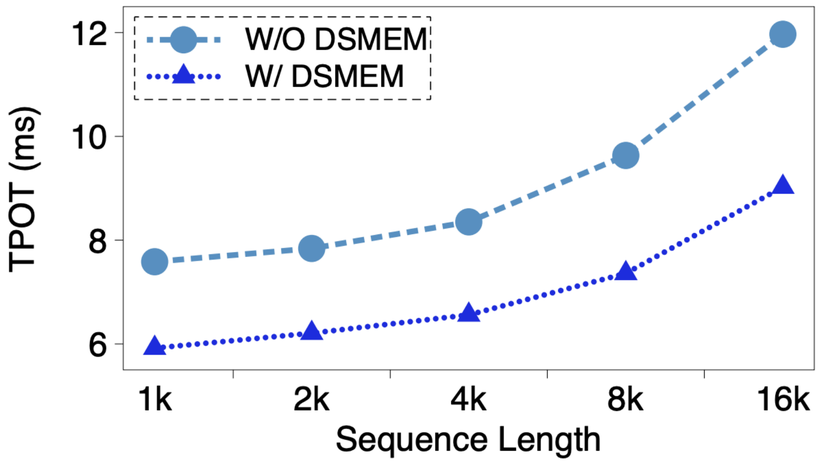


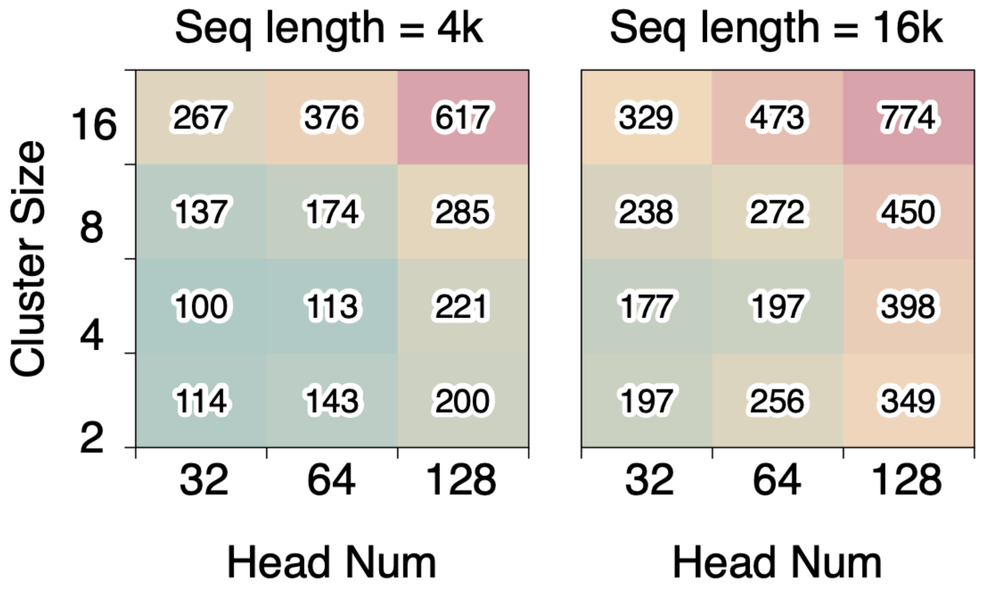


Evaluation

More analysis

- 1. If we use the same dataflow but without DSM?
  - Ascend 910B, AMD RDNA3 do not support inter-connect temporarily.
- 2. Does different cluster setting matter?
  - Yes, also further study.

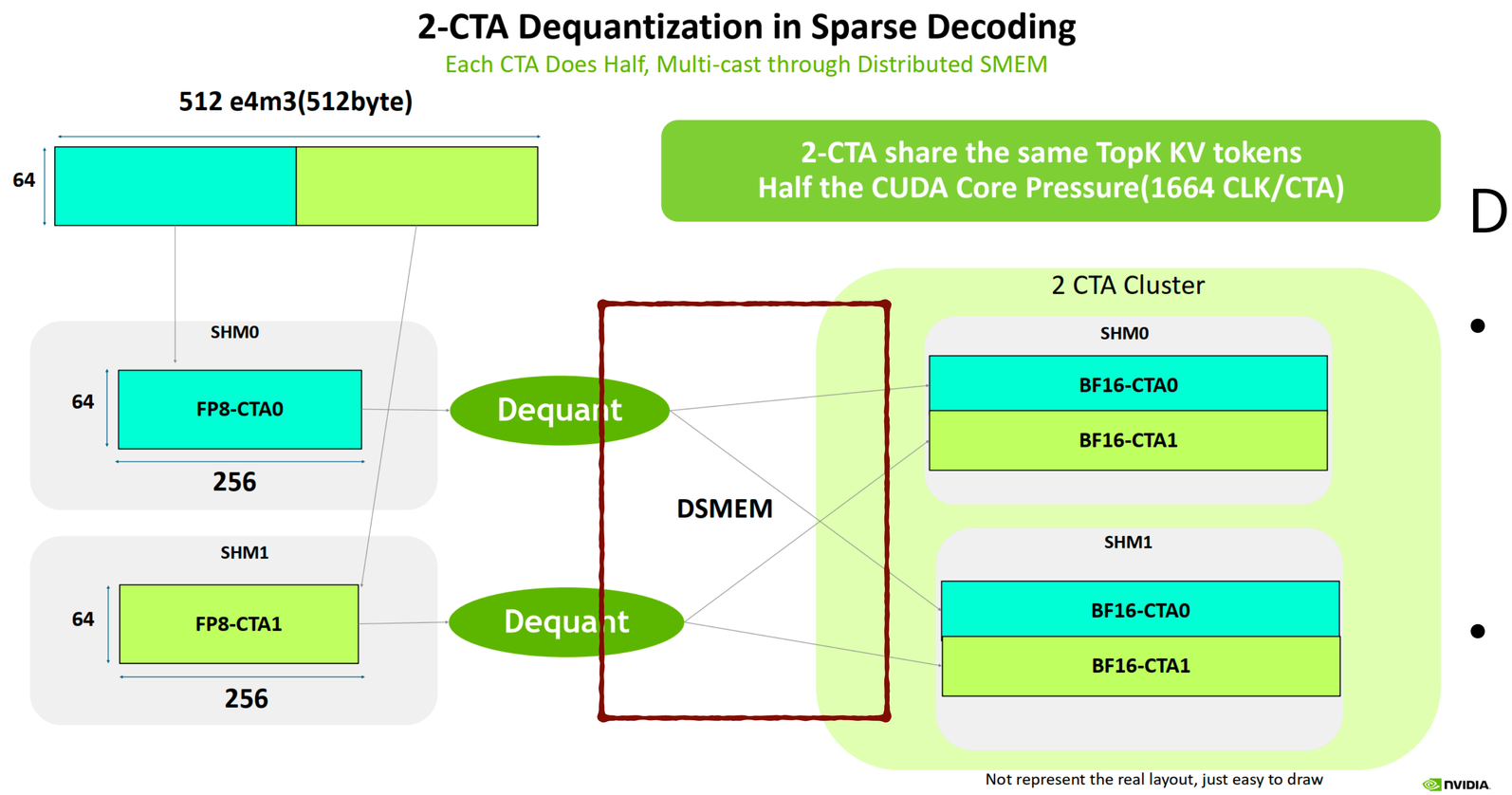




Discussion

New Interconnect: DSM

Existing Use Cases of DSM



DSA Kernel Design:

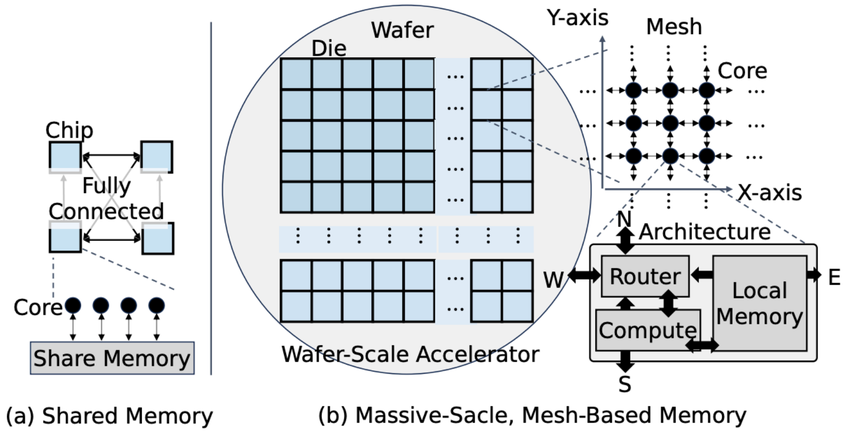
- Using DSM to **gather** the full dequantized kv cache.
- Nearly to our **ClusterGather**

Best Practice of MLA Kernel Optimizaition on Blackwell, NVIDIA, November 7, 2025.

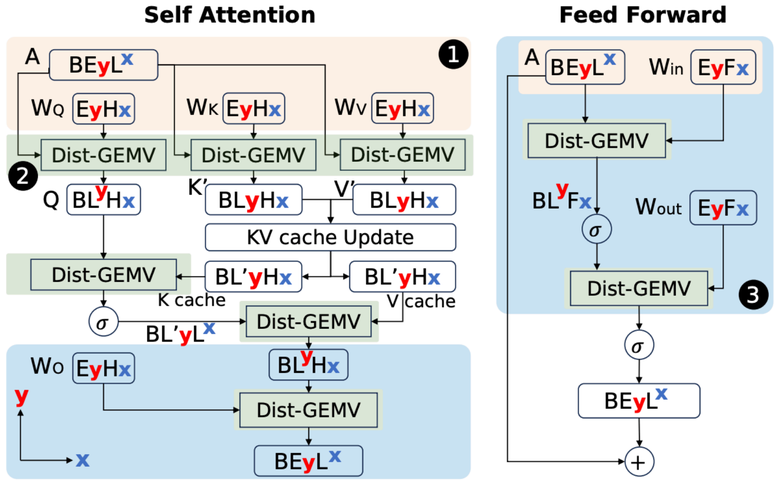
Discussion

Low-Latency Inference On Interconnect Chip

- Some works have explored LLM inference on interconnect chips.
  - WaferLLM (OSDI 25), T10 (SOSP 24), ...
  - Cerebras, Graphcore IPU, ...
  - Ultra-low-latency inference dataflow through on-chip interconnect.



(a) Shared Memory (b) Massive-Sacle, Mesh-Based Memory



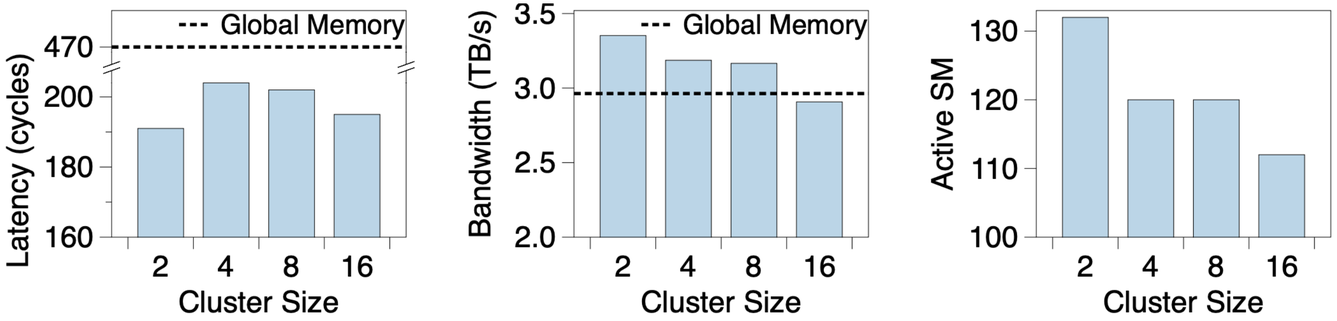
Our work implements low-latency dataflow leveraging interconnect of GPUs.

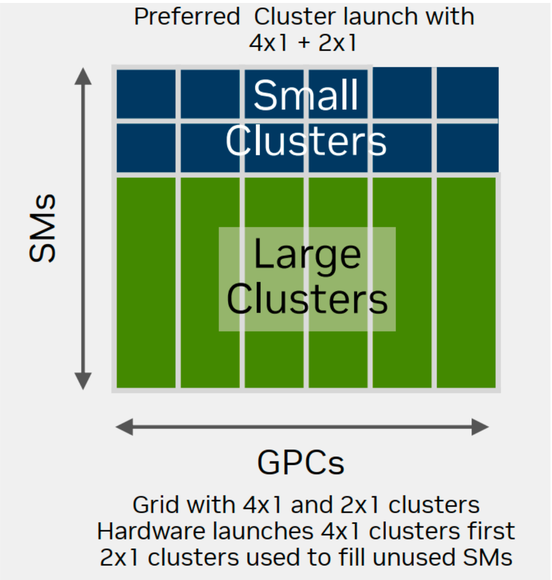
WaferLLM: Large Language Model Inference at Wafer Scale, OSDI 2025.

Discussion

Limitations & Future Work

- 1. **Compiler** techniques to automatically support for more workloads.





- 2. Integration with **sparse and quantization** algorithm.
- 3. Extend our kernel to support **preferred thread block clusters** on Blackwell GPUs.

Thanks

Q & A

- ClusterFusion: **Expanding** Operator **Fusion Scope** for LLM Inference via Cluster-Level Collective Primitive

- Open source

- Paper