香港科技大學
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

DATE 26

FlashGEMM: Mesh-Aware Efficient GEMM for 3D-Stacked LLM Accelerators

Xin Fan[1], Chen Bai[1,2,†], Xin Yang[2], Zhenhua Zhu[1,3], Yanhong Wang, Zhaode Zhang, Yuan Xie[1]

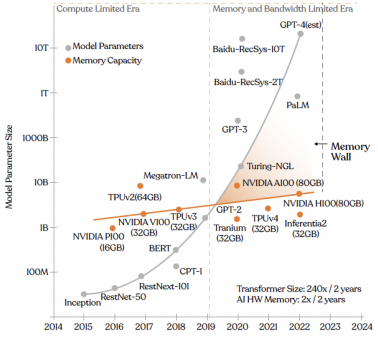
[1] The Hong Kong University of Science and Technology
[2] Fudan University
[3] Tsinghua University

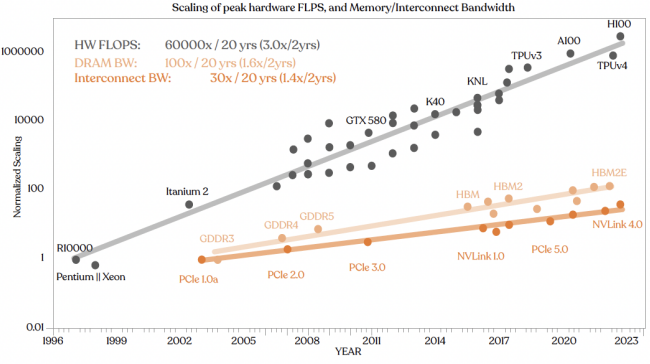
Apr. 20, 2026

Outline

Introduction

Memory Wall in Large Language Models (LLM) Era





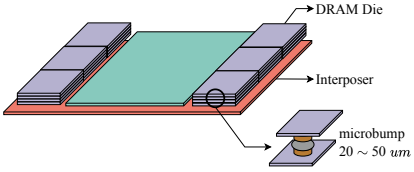
Memory Wall [1]

DRAM bandwidth scaling: only 1.6× per two years

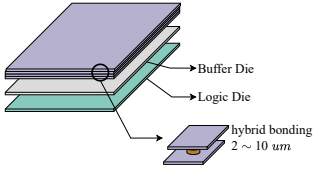
This model-hardware scaling disparity creates bottlenecks in both memory capacity and bandwidth.

[1] MVP Ventures (n.d.). *Venture Bytes #111: AI Has a Memory Problem.* https://www.mvp.vc/venture-bytes/venture-bytes-111-ai-has-a-memory-problem. Accessed: 2025-11-13

High Bandwidth Memory (HBM) & 3D-Stacked Memory



NVIDIA H100 with HBM



Accelerator with 3D-Stacked Memory

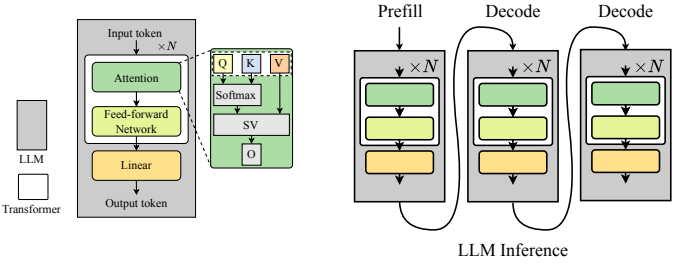
HBM: Memory dies (HBM stacks) are placed **side by side** around the logic die.

- Bandwidth density $\propto$ logic die side length N.
- Example: NVIDIA B200 achieves up to **8 TB/s** aggregate bandwidth.

3D-Stacked Memory: Memory dies are stacked **vertically** on top of the logic die.

- Bandwidth density $\propto$ logic die area N^2
- Provides **4**$\times$ higher memory bandwidth than HBM.

Large Language Model



Large Language Models Architecture

- LLMs primarily use transformer blocks.
- **Three key operations:** general matrix-matrix multiplication (GEMM), general matrix-vector multiplication (GEMV) and element-wise operator.
- **GEMM** is the dominant operation.

LLM Inference Workflow

Prefill Stage

- Processes the entire input prompt in one pass.
- Typically compute-bound due to GEMM.

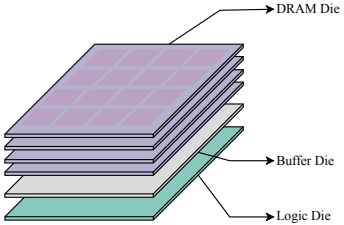
Decode Stage

- Use key-value (KV) cache to avoid recomputation.
- Process one new token in one pass.
- Memory-bound due to GEMV and its batched version, flat GEMM.

2

[2] Flat GEMM denotes $\boldsymbol{C}_{m\times n} = \boldsymbol{A}_{m\times k}\boldsymbol{B}_{k\times n}$ where $m \ll k, n$. For example, $m = 8$, and $k = n = 2048$.

3D-Stacked LLM Accelerators



3D-Stacked LLM Accelerators

Problem Formulation

- Previous distributed GEMM algorithms are inefficient due to unawareness of 3D memory and mesh network-on-chip (NoC) topology.

How to customize an efficient GEMM algorithm for 3D-stacked LLM accelerators?

LLM inference is GEMM-intensive.

34567

[3]Robert A Van De Geijn and Jerrell Watts (1997). "SUMMA: Scalable Universal Matrix Multiplication Algorithm". In: 9.4, pp. 255–274.

[4]Congjie He et al. (2025). "WaferLLM: Large Language Model Inference at Wafer Scale". In: *USENIX Symposium on Operating Systems Design and Implementation (OSDI)*.

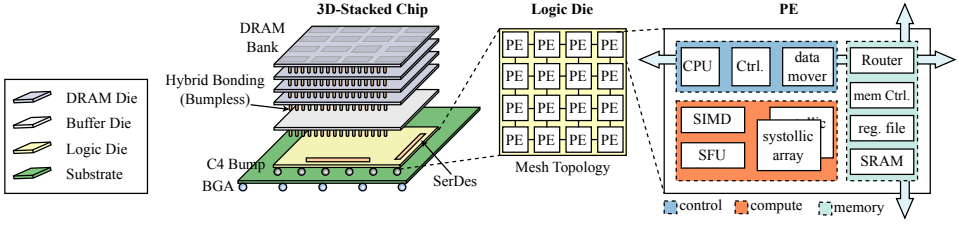
[5]Lynn Elliot Cannon (1969). *A Cellular Computer to Implement the Kalman Filter Algorithm*. Montana State University.

[6]Martin D. Schatz, Robert A. van de Geijn, and Jack Poulson (Jan. 2016). "Parallel Matrix Multiplication: A Systematic Journey". In: *SIAM J. Sci. Comput.*, pp. C748–C781.

[7]Hyoungwook Nam, Gerasimos Gerogiannis, and Josep Torrellas (2025). "MeshSlice: Efficient 2D Tensor Parallelism for Distributed DNN Training". In: *IEEE/ACM International Symposium on*

FlashGEMM

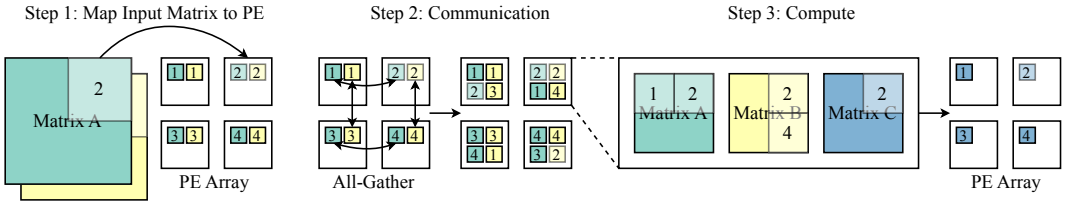
Architecture Overview



Architecture Overview of the 3D-Stacked LLM Accelerator.

- Memory dies are stacked vertically on top of the logic die via hybrid bonding.
- Processing Elements (PEs) on logic die are connected through a 2D mesh NoC.

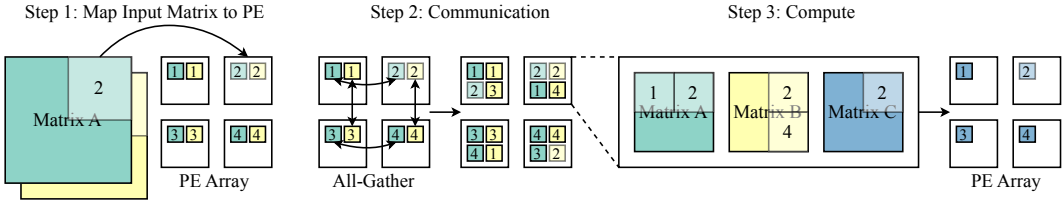
Scalable Universal Matrix Multiplication Algorithm (SUMMA)



Original Scalable Universal Matrix Multiplication Algorithm

- Poor communication-computation overlap.
- Long communication path in mesh topology.
- Static output-stationary dataflow.

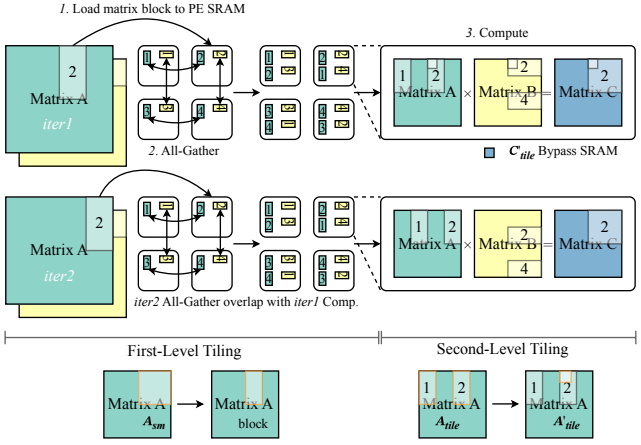
Scalable Universal Matrix Multiplication Algorithm (SUMMA)



Original Scalable Universal Matrix Multiplication Algorithm

- Poor communication-computation overlap.
- Long communication path in mesh topology.
- Static output-stationary dataflow.
- In the following, we discuss **FlashGEMM**, our efficient distributed GEMM algorithm.
  - Communication-computation overlap. → Reduce exposed communication time.
  - All-gather algorithm. → Shorten communication hops.
  - Dynamic dataflow. → Minimizes communication volumes.

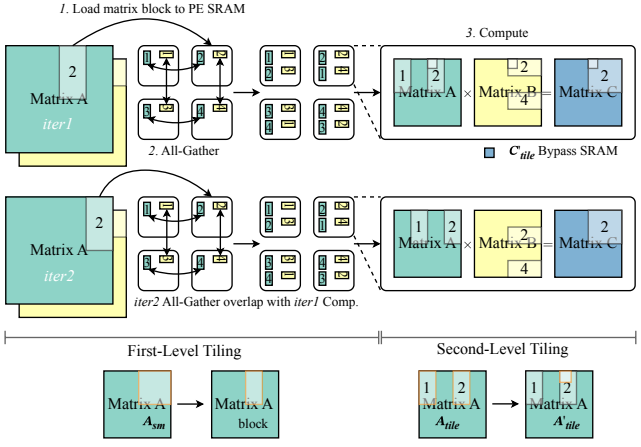
Communication-Computation Overlap



Two-level Tiling Strategy.

- **First-level tiling:** Slices matrix along K-dimensions. Row/column all-gather of slices run in parallel with computation.
- **Second-level tiling:** Splits every SRAM-level tile into smaller, register-level tiles.

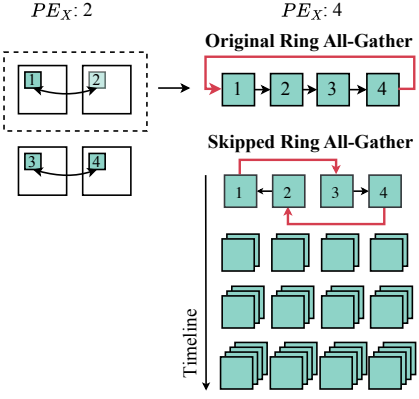
Communication-Computation Overlap



Two-level Tiling Strategy.

- The stationary variables, i.e. *C*, bypass SRAM and load directly from stacked DRAM into registers.

All-Gather Communication Algorithm

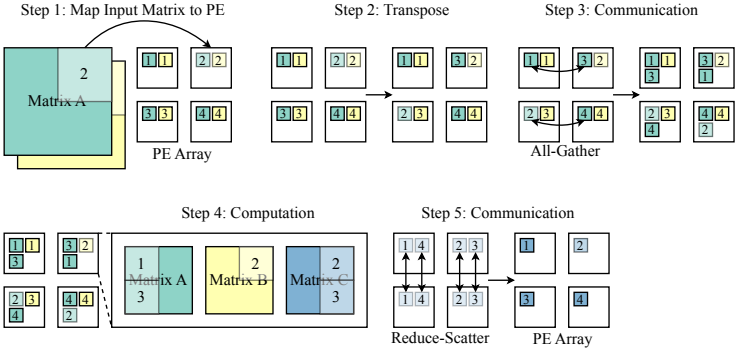


Skipped-Ring All Gather

Skipped Ring All-Gather

- **All-Gather.** Every GPU collects data from all others.
- Odd-numbered (Even-numbered) PEs send data to the next odd-numbered(Even-numbered) PE.
- Eliminate the warparound path resides in orginal ring all-gather.
- The longest path is 2 hops.

Dynamic Dataflow



Weight-Stationary Dataflow

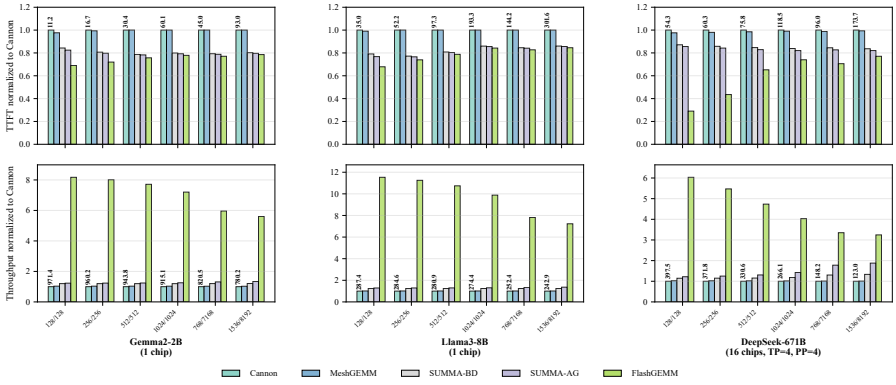
- In the decode stage, the input and output data volume is negligible compared to the weight matrix, owing to the KV cache.
- Adopt output-stationary and weight-stationary dynamically to minimize data movement.

Evaluation

Architecture configurations

Metric	Value
Silicon area of the monolithic design	32×25 mm^2
# of DRAM stacks	4
Total DRAM capacity	64 GB
Aggregate DRAM bandwidth	24 TB/s
# of processing elements (PEs)	16×16
Computing power	1024 TFLOPS@FP8
NoC bisection bandwidth	1.0 TB/s
Scale-up bandwdith	800 GB/s

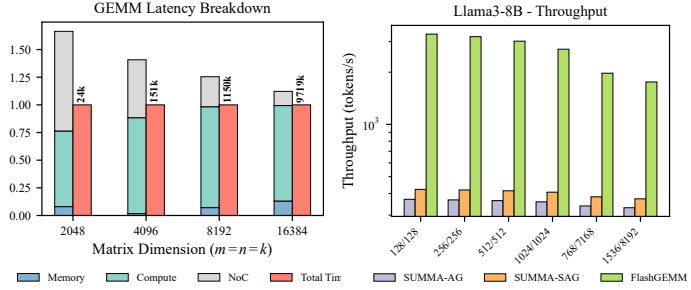
Performance Comparison



Performance comparison of distributed GEMM algorithms on 3D-stacked LLM accelerators

- FlashGEMM delivers 1.23× TTFT reduction in prefill and 5.44× throughput improvement in decode, respectively.

Ablation Study



(a) Latency breakdown. (b) LLM throughput under different settings.

- FlashGEMM achieves computation-communication overlap and delivers 1.12∼1.66× speedup for GEMM with varying matrix dimensions.
- FlashGEMM and SUMMA with skipped-ring all-gather deliver 7.44× and 1.14× average throughput improvements over SUMMA, respectively.

Conclusion

Conclusion

- We propose FlashGEMM, an efficient matrix-multiplication algorithm for 3D-stacked LLM accelerators with mesh topology.
- Evaluations show that FlashGEMM delivers up to 1.50× improvement in time-to-first-token (TTFT) and 7.11× improvement in throughput over prior methods.

THANK YOU!

References I

Lynn Elliot Cannon (1969). *A Cellular Computer to Implement the Kalman Filter Algorithm*. Montana State University.

Congjie He et al. (2025). "WaferLLM: Large Language Model Inference at Wafer Scale". In: *USENIX Symposium on Operating Systems Design and Implementation (OSDI)*.

MVP Ventures (n.d.). *Venture Bytes #111: AI Has a Memory Problem*. https://www.mvp.vc/venture-bytes/venture-bytes-111-ai-has-a-memory-problem. Accessed: 2025-11-13.

Hyoungwook Nam, Gerasimos Gerogiannis, and Josep Torrellas (2025). "MeshSlice: Efficient 2D Tensor Parallelism for Distributed DNN Training". In: *IEEE/ACM International Symposium on Computer Architecture (ISCA)*, pp. 821–834.

Martin D. Schatz, Robert A. van de Geijn, and Jack Poulson (Jan. 2016). "Parallel Matrix Multiplication: A Systematic Journey". In: *SIAM J. Sci. Comput.*, pp. C748–C781.

Robert A Van De Geijn and Jerrell Watts (1997). "SUMMA: Scalable Universal Matrix Multiplication Algorithm". In: 9.4, pp. 255–274.