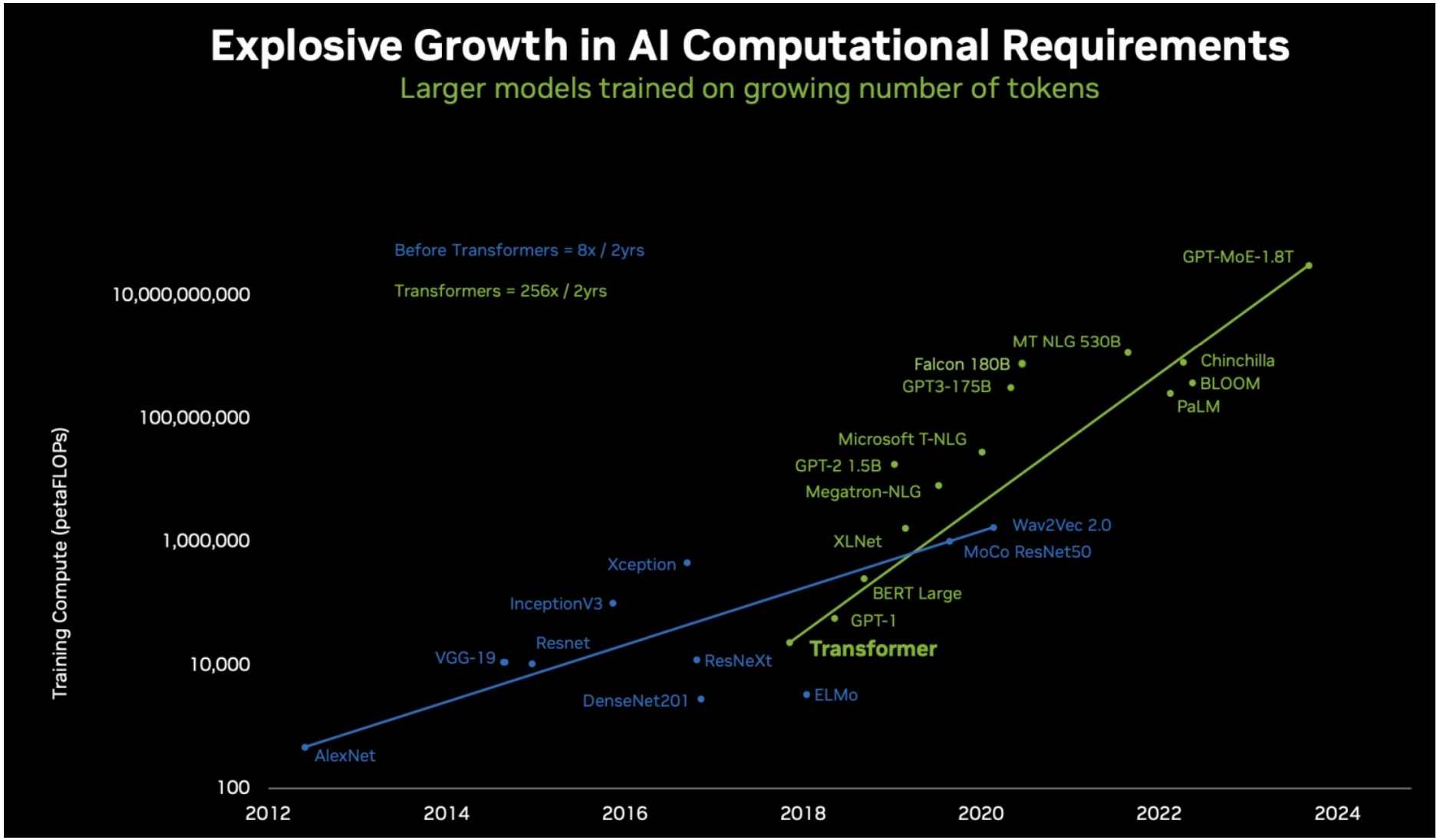
香港科技大学(广州)
THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY (GUANGZHOU)

Chimera: Communication Fusion for Hybrid Parallelism in Large Language Models

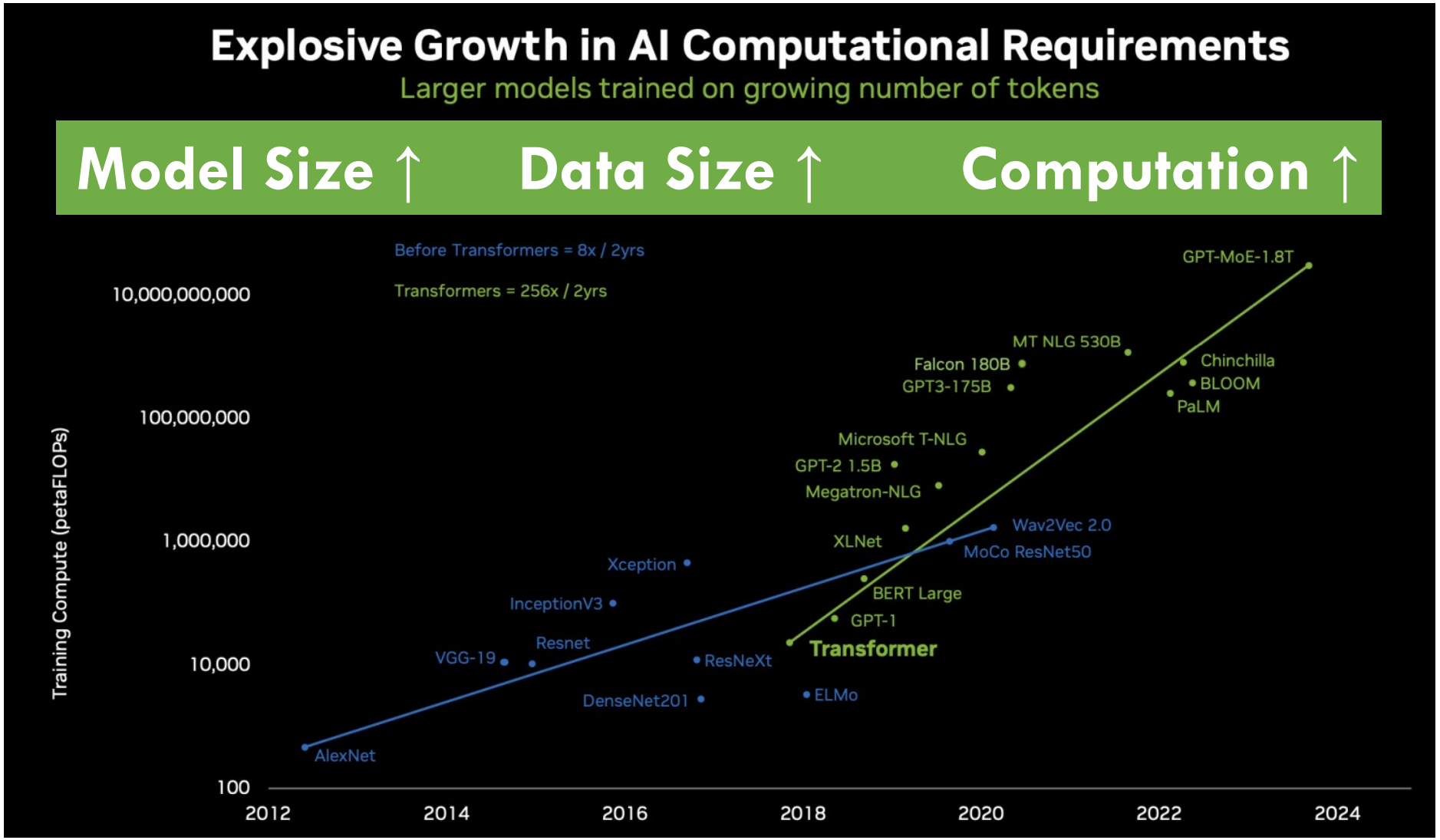
Le Qin, Junwei Cui, Weilin Cai, Jiayi Huang

Scaling of LLM Computational Requirements



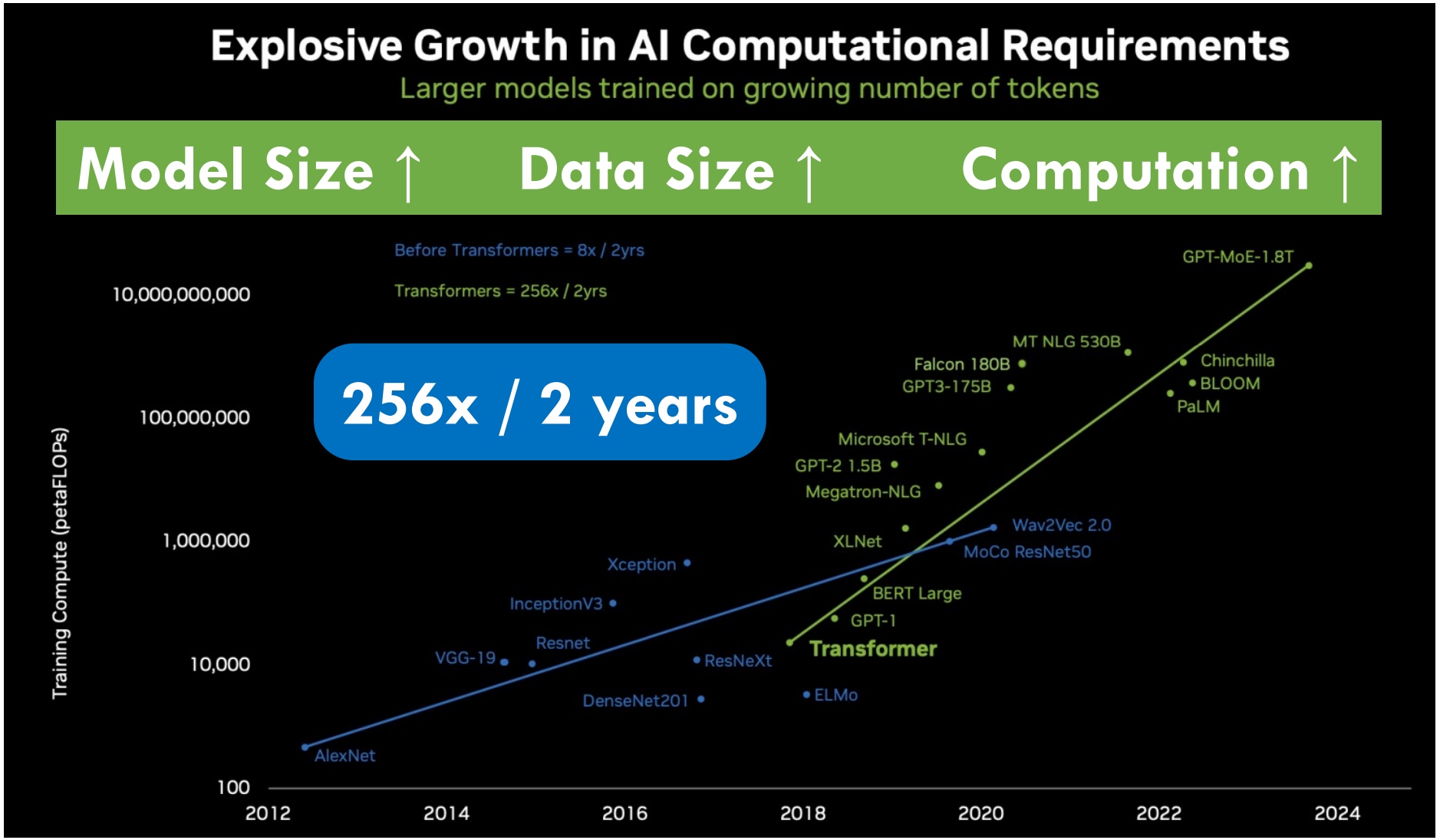
Source: NVIDIA, GTC March 2024 Keynote.

Scaling of LLM Computational Requirements



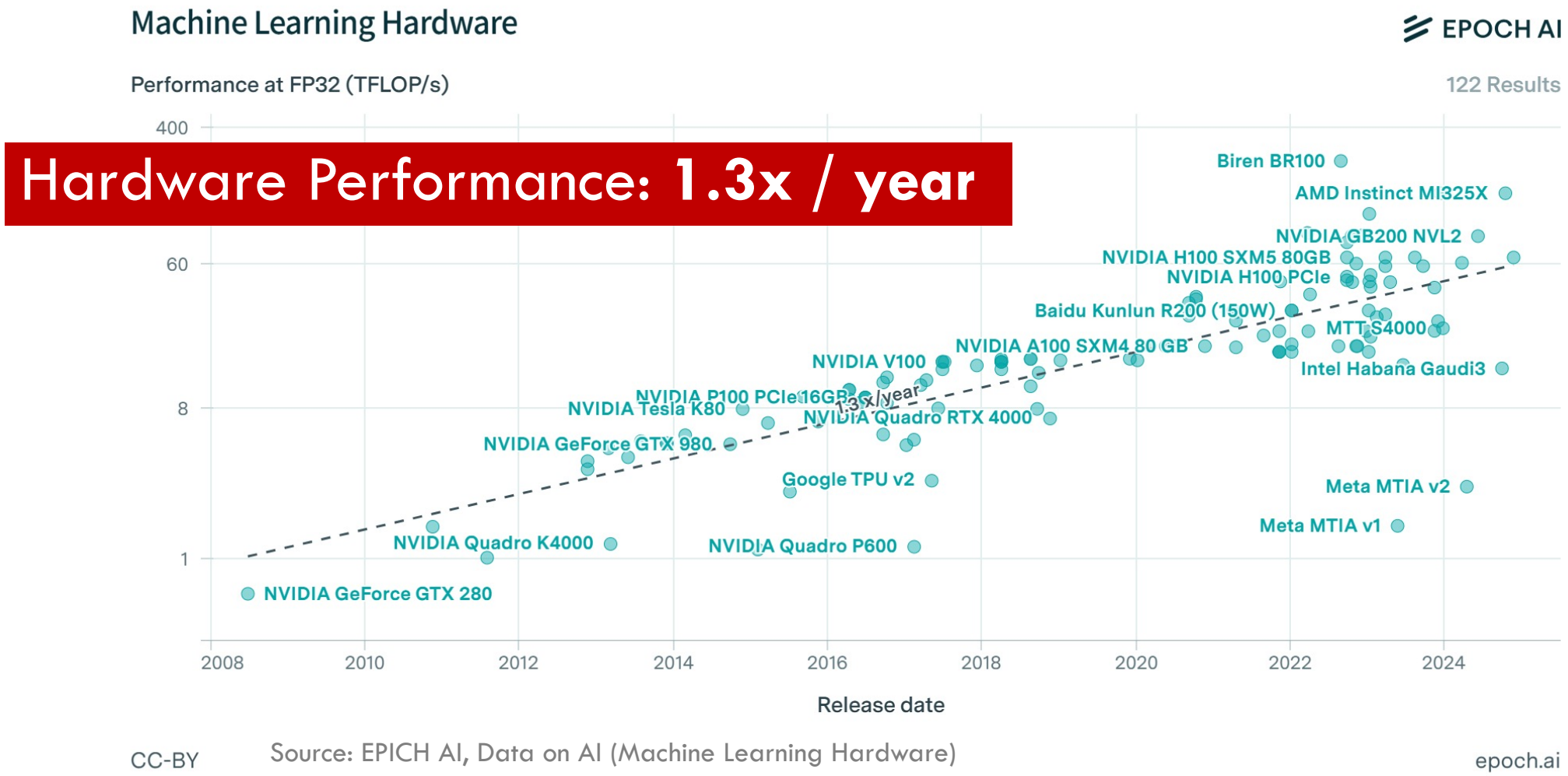
Source: NVIDIA, GTC March 2024 Keynote.

Scaling of LLM Computational Requirements

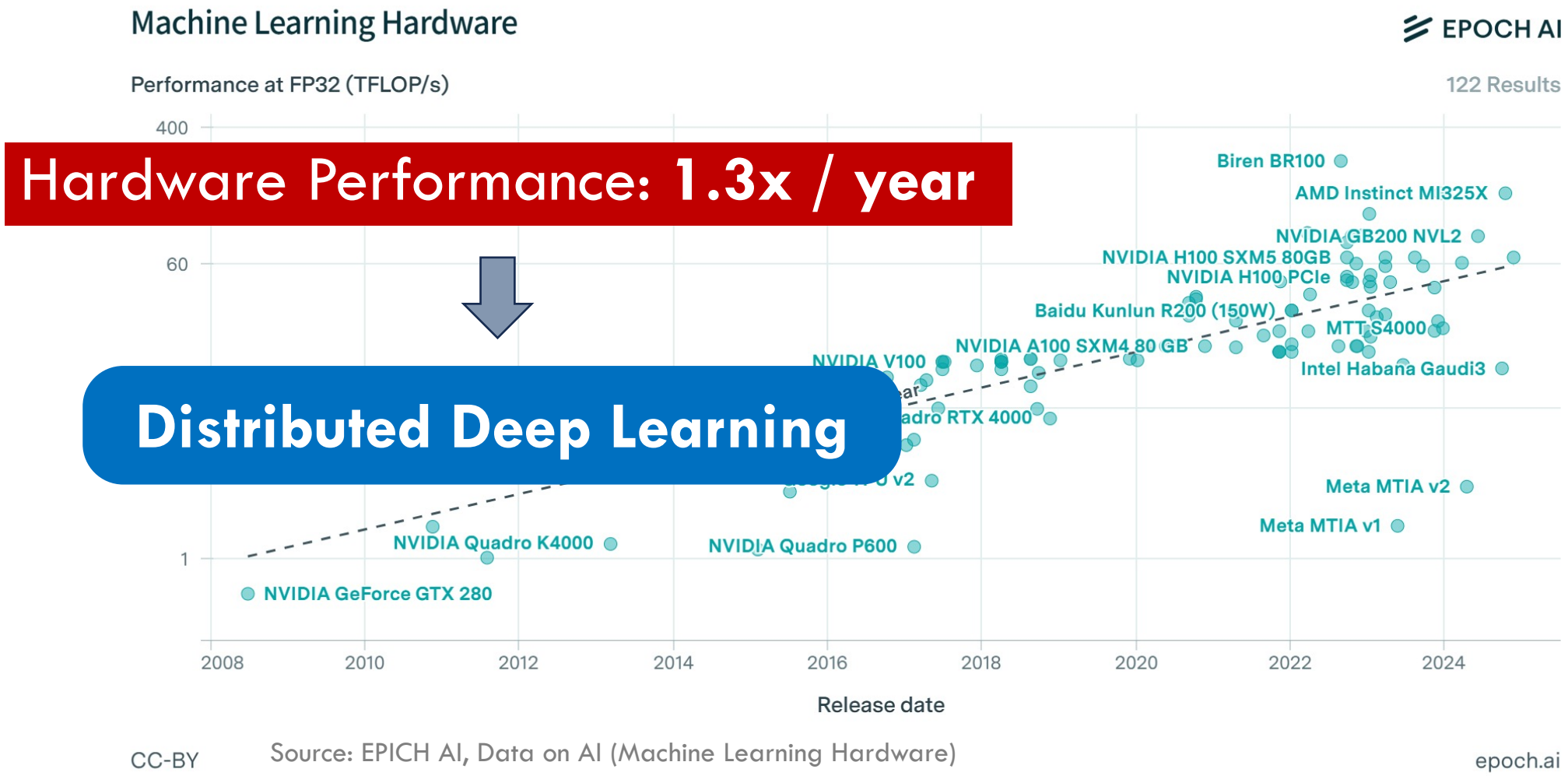


Source: NVIDIA, GTC March 2024 Keynote.

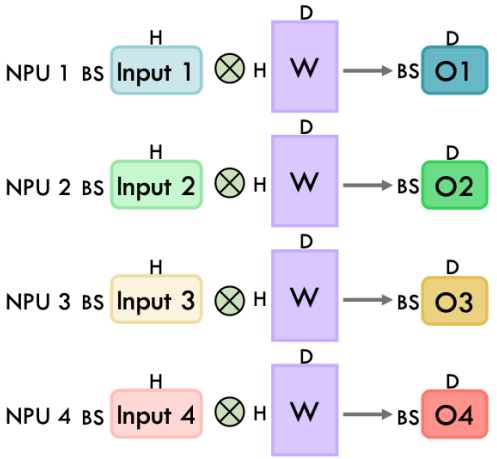
Scaling of Machine Learning Hardware



Scaling of Machine Learning Hardware



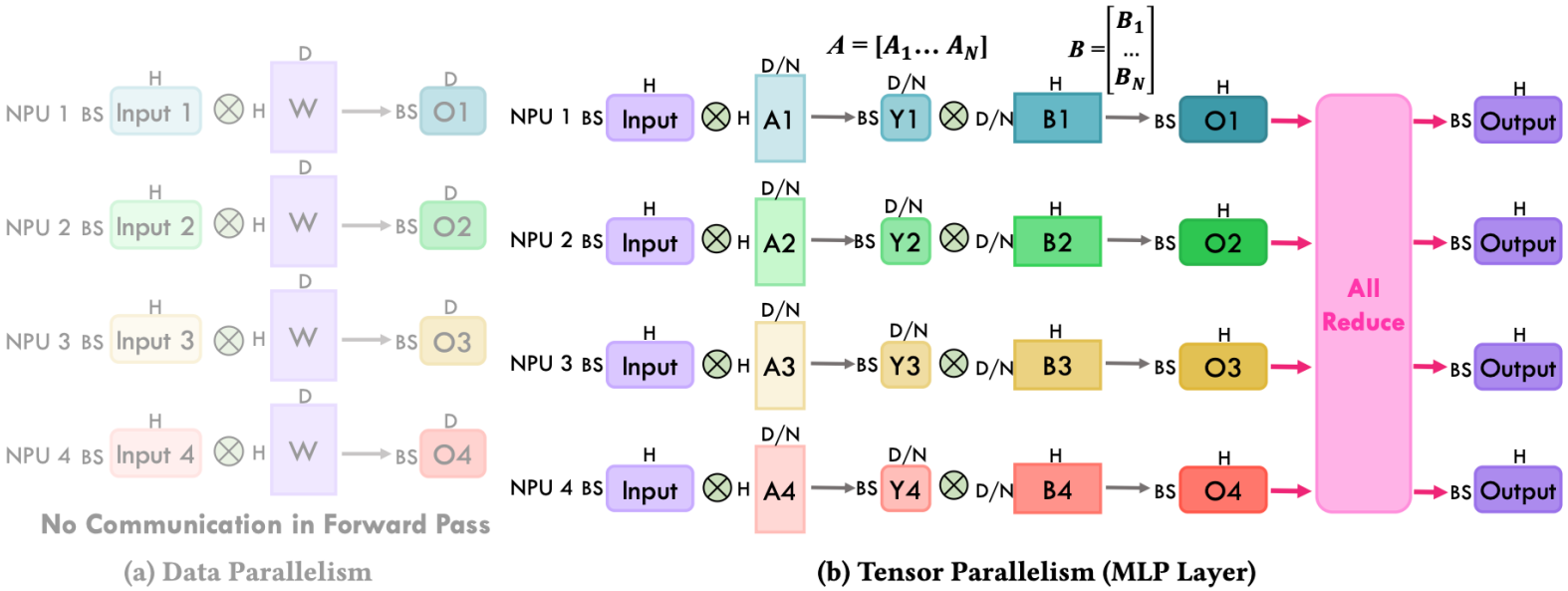
Parallel Patterns in Distributed LLM



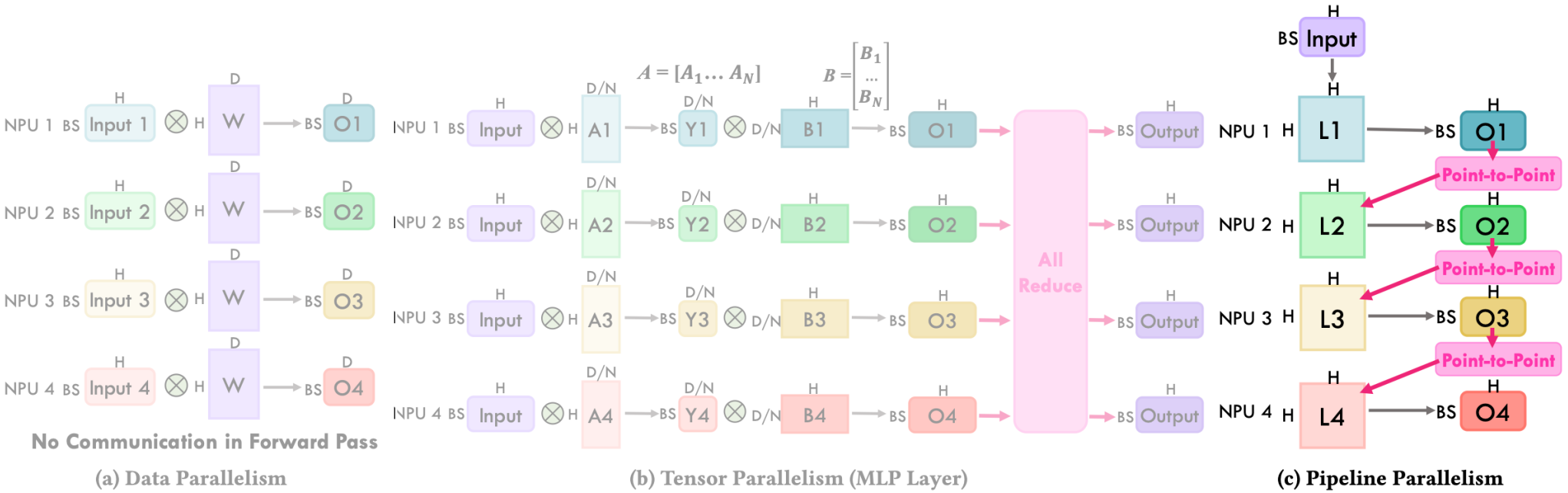
No Communication in Forward Pass

(a) Data Parallelism

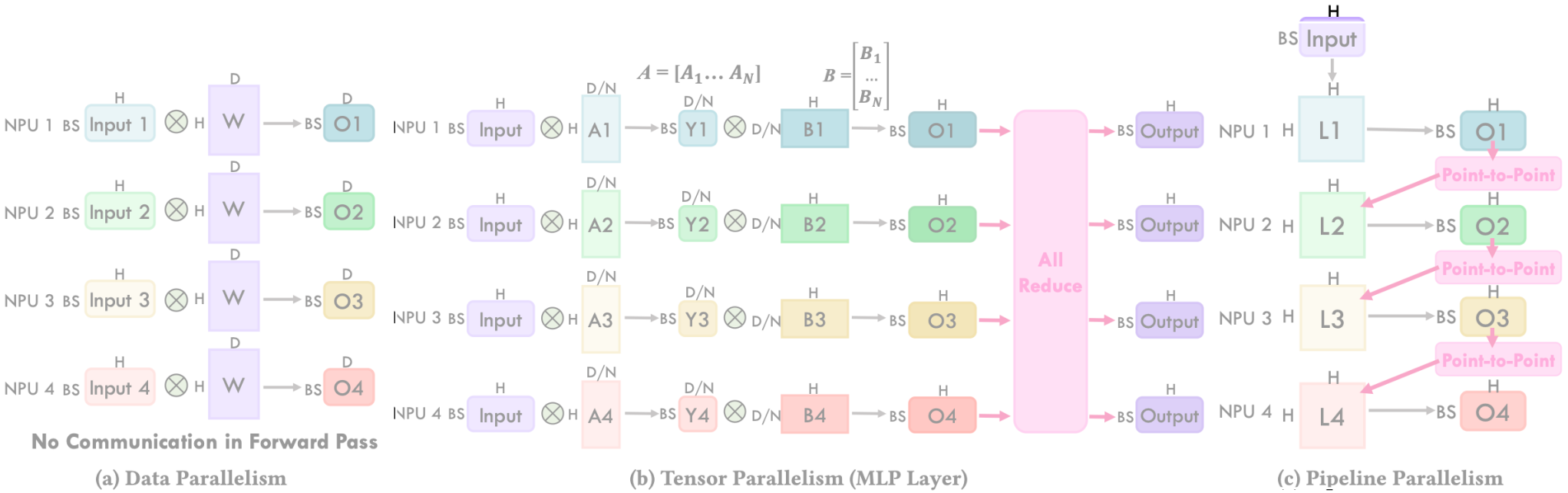
Parallel Patterns in Distributed LLM



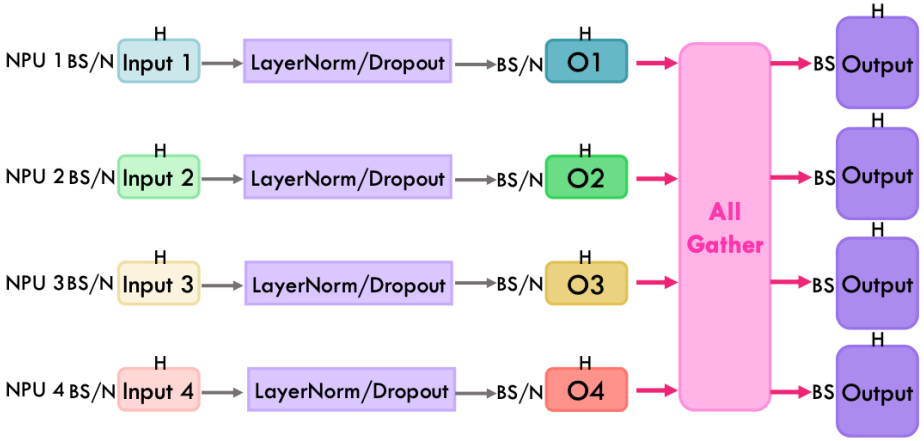
Parallel Patterns in Distributed LLM



Parallel Patterns in Distributed LLM

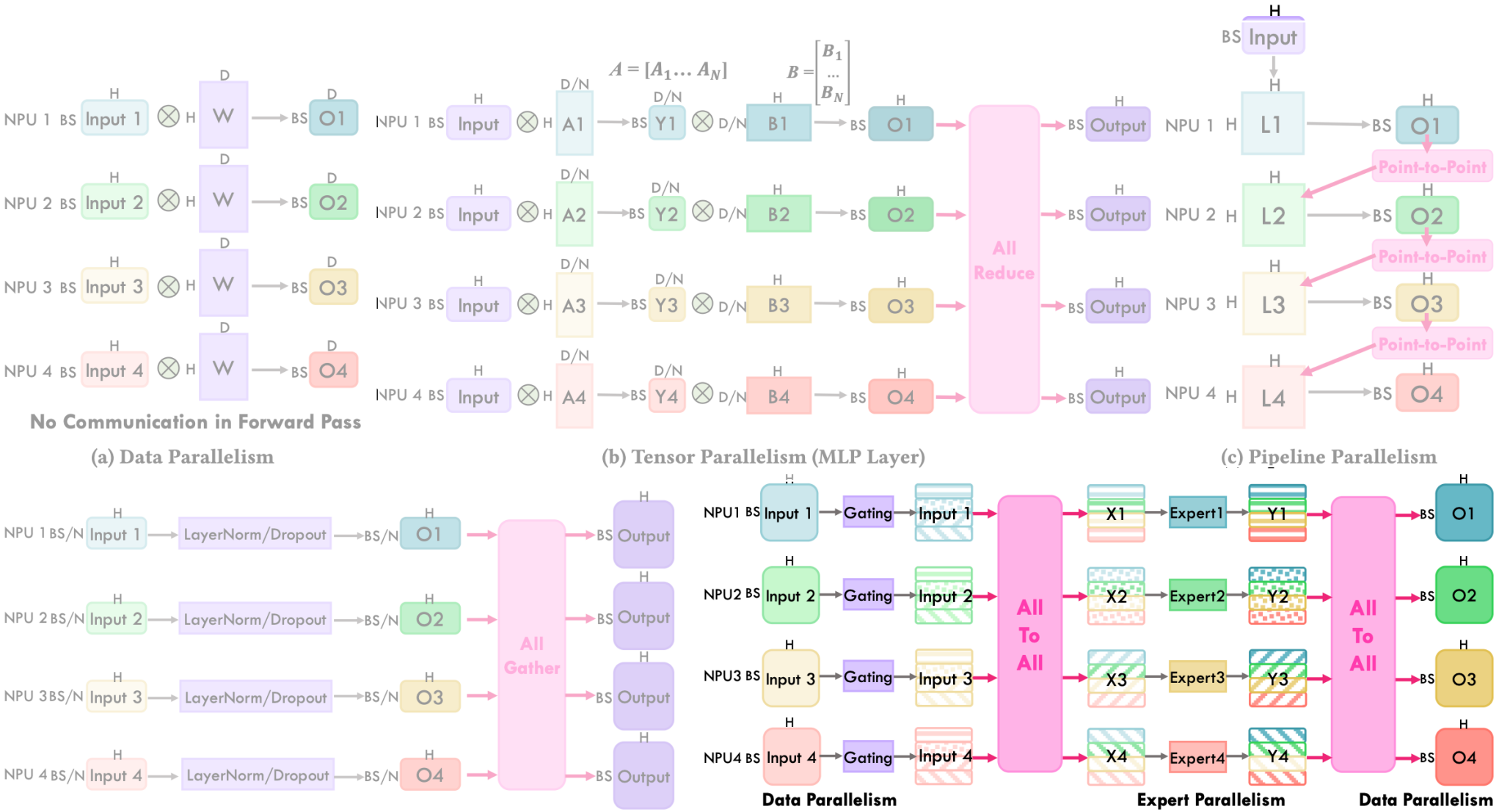


(a) Data Parallelism (b) Tensor Parallelism (MLP Layer) (c) Pipeline Parallelism



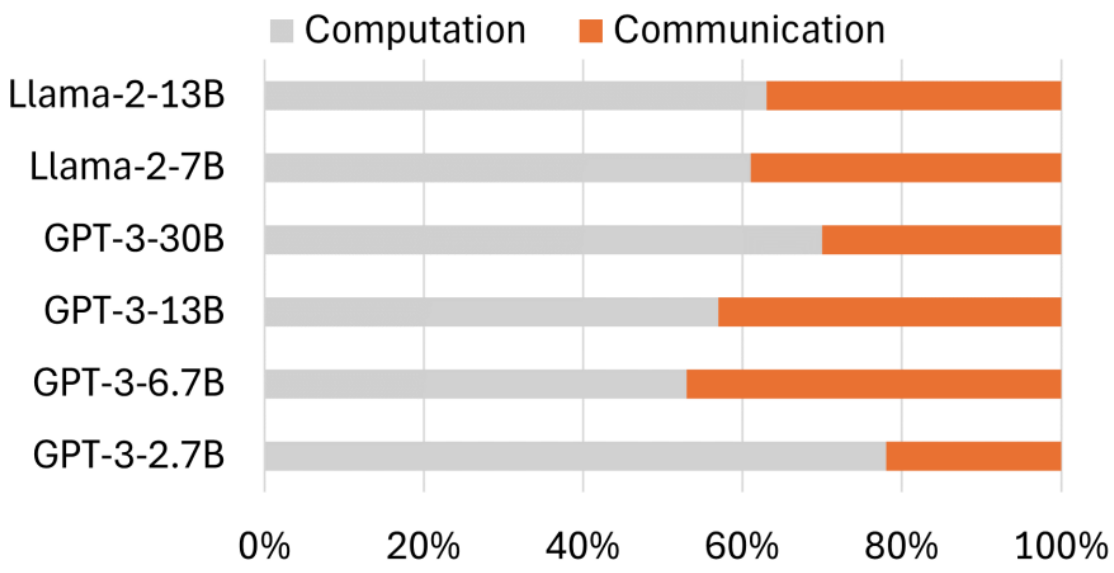
(d) Sequence Parallelism

Parallel Patterns in Distributed LLM



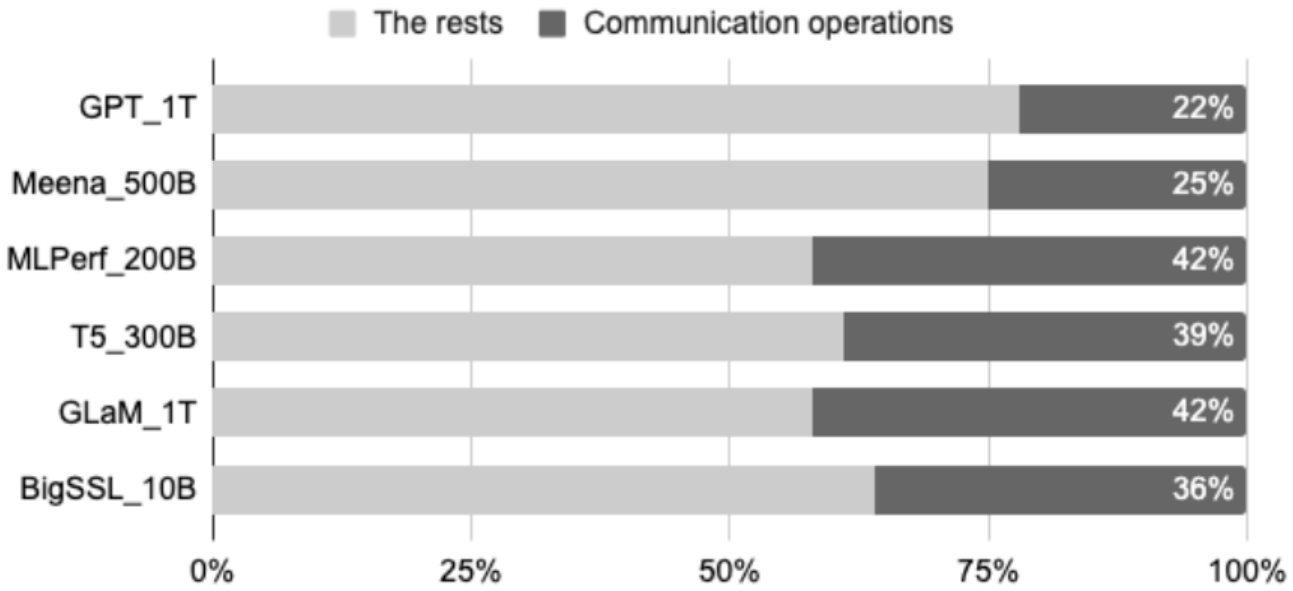
Communication is the Bottleneck

NVIDIA GPU Cluster



Source: Microsoft DeepSpeed, Communication ratio on 8 – 32 H100 GPUs

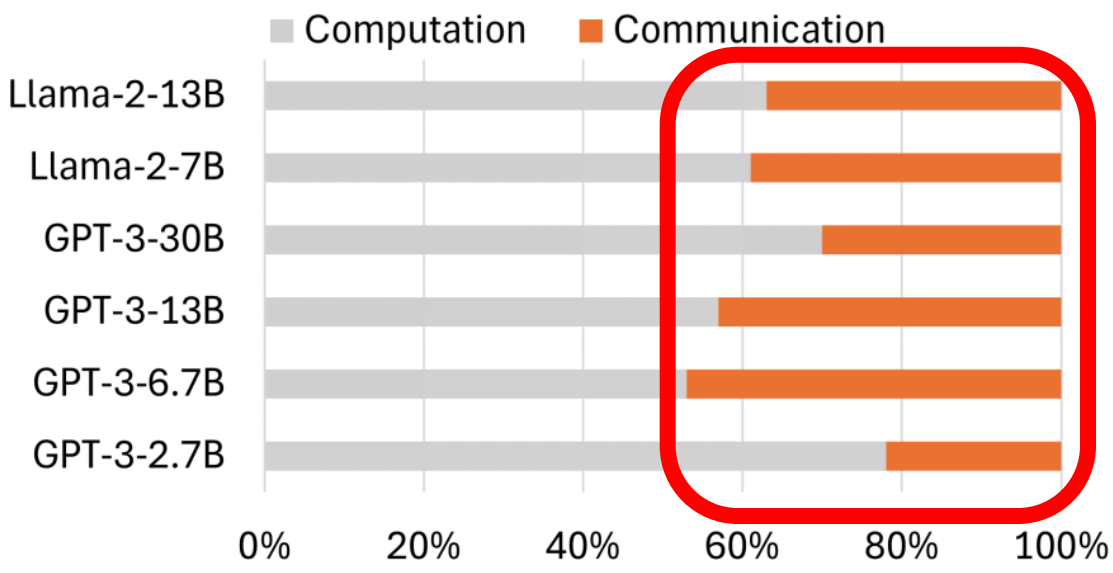
Google TPU Cluster



Source: Google, Communication ratio on 128 – 2048 TPUs

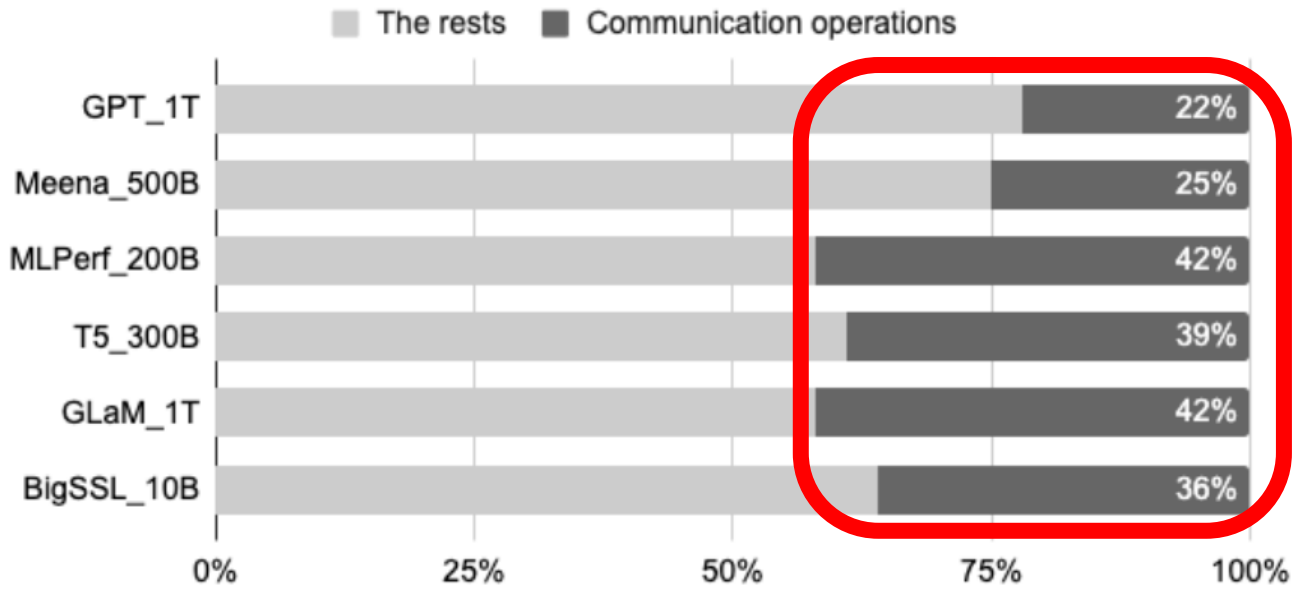
Communication is the Bottleneck

NVIDIA GPU Cluster



Source: Microsoft DeepSpeed, Communication ratio on 8 – 32 H100 GPUs

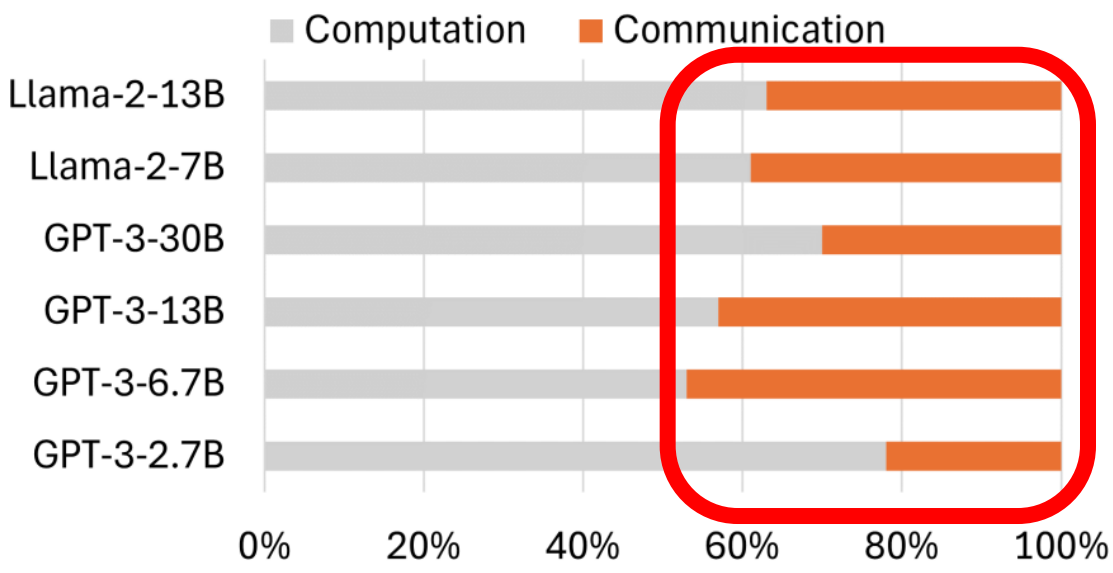
Google TPU Cluster



Source: Google, Communication ratio on 128 – 2048 TPUs

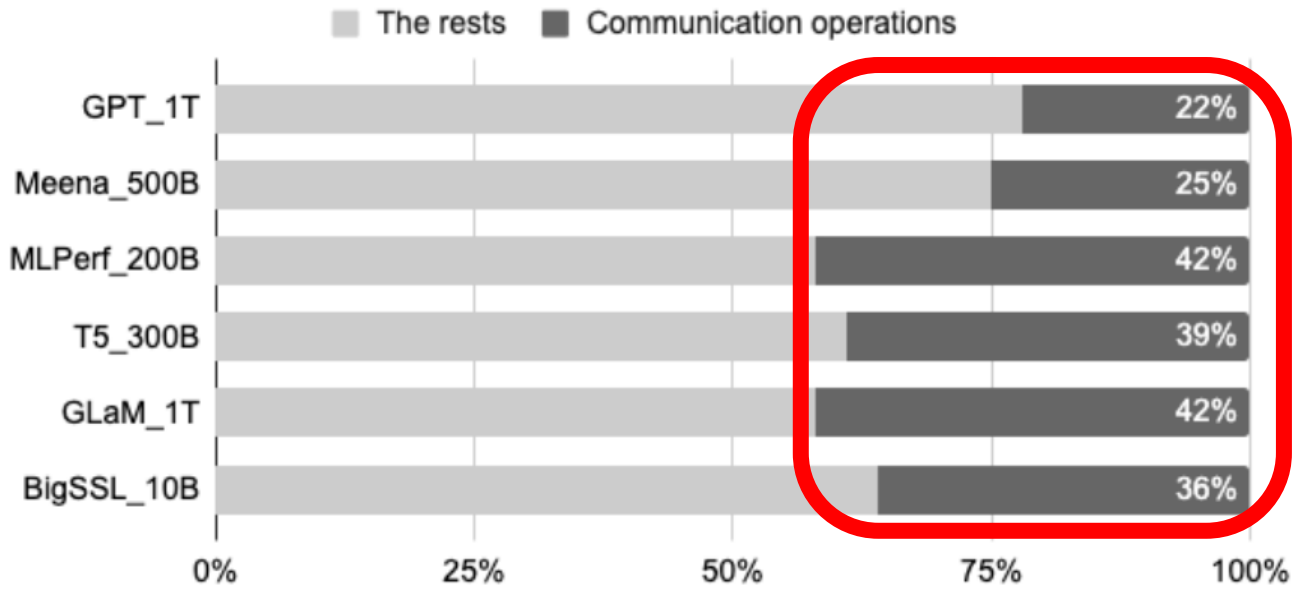
Communication is the Bottleneck

NVIDIA GPU Cluster



Source: Microsoft DeepSpeed, Communication ratio on 8 – 32 H100 GPUs

Google TPU Cluster

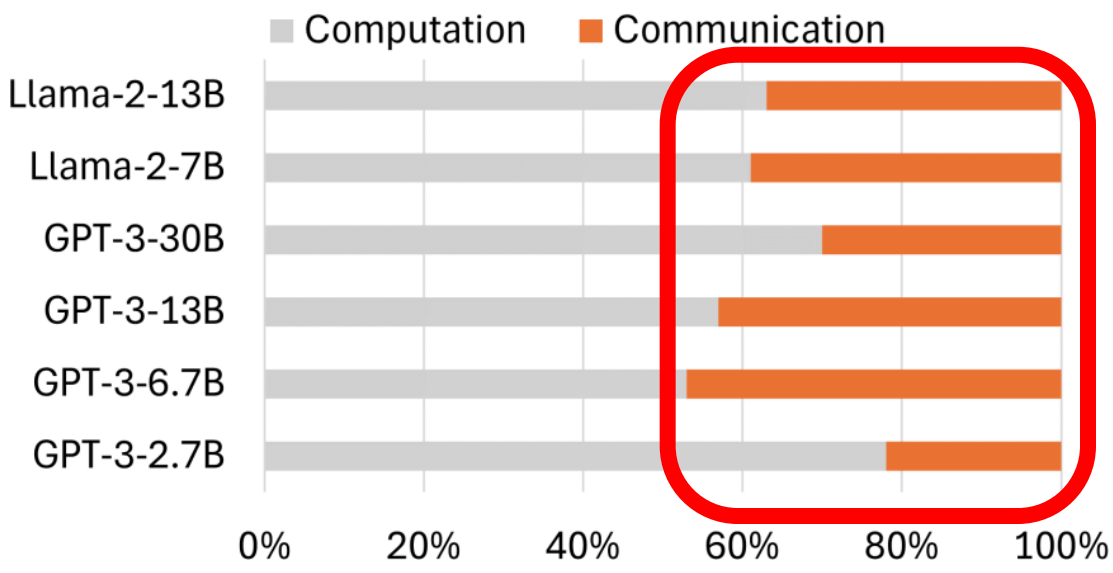


Source: Google, Communication ratio on 128 – 2048 TPUs

High communication overhead in large-scale distributed deep learning

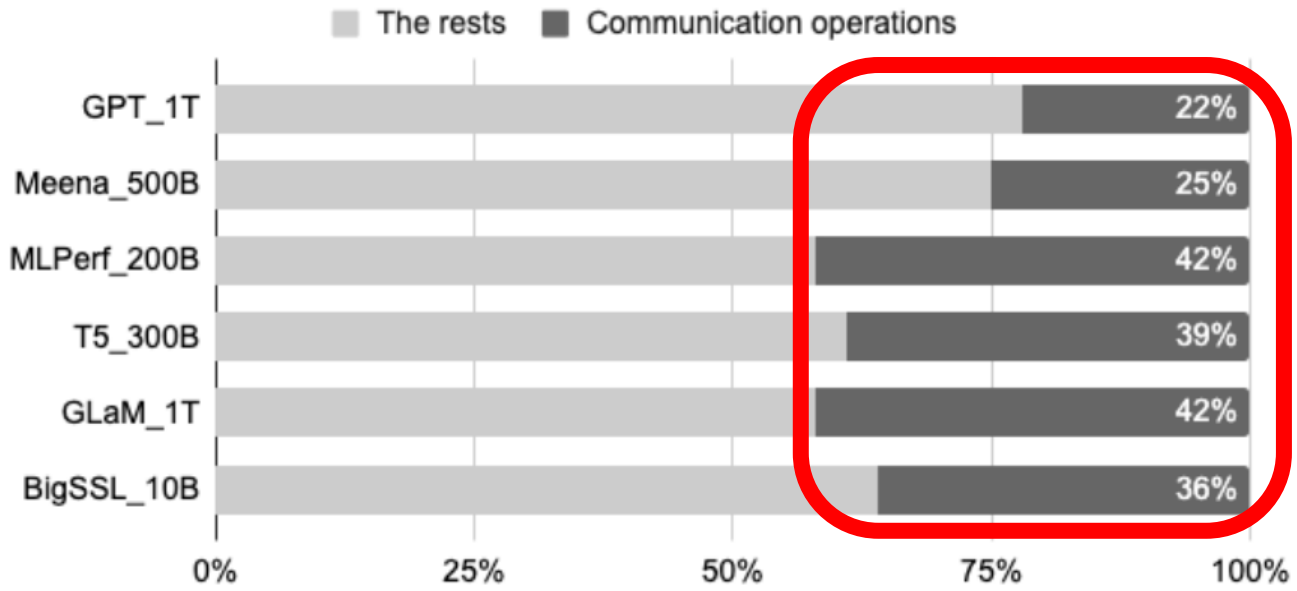
Communication is the Bottleneck

NVIDIA GPU Cluster



Source: Microsoft DeepSpeed, Communication ratio on 8 – 32 H100 GPUs

Google TPU Cluster

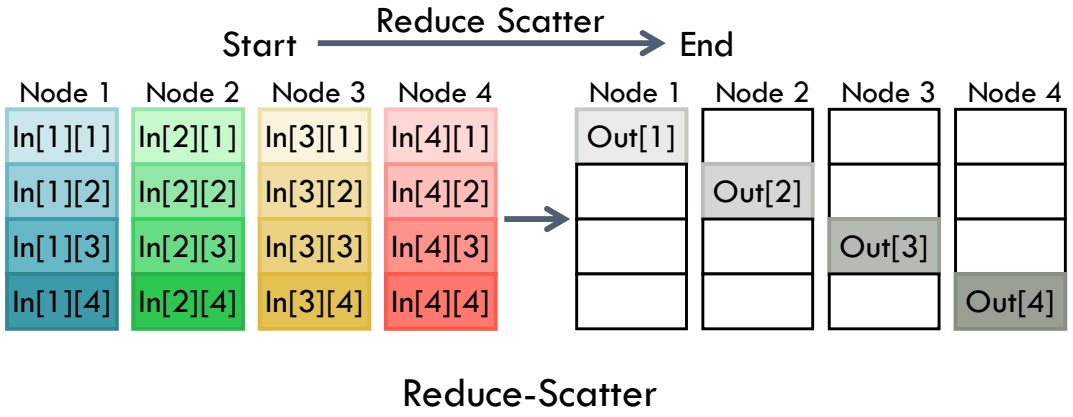


Source: Google, Communication ratio on 128 – 2048 TPUs

High communication overhead in large-scale distributed deep learning

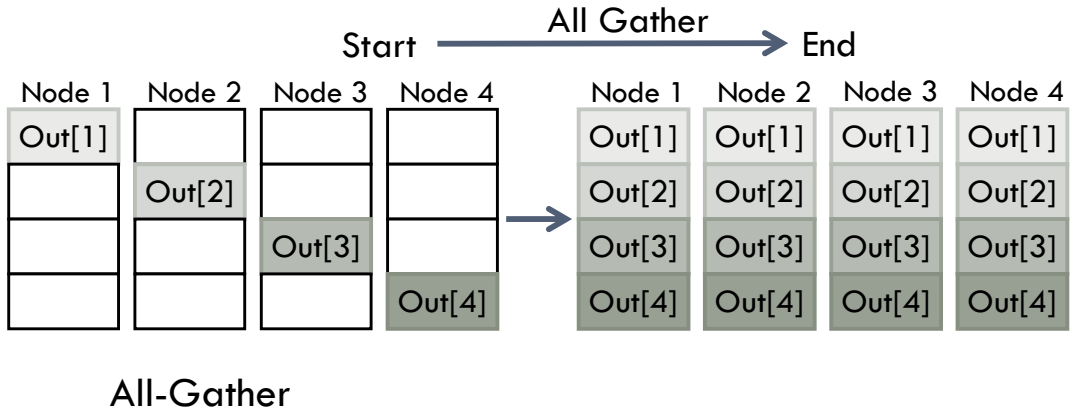
Require further communication optimization

Collective Communications in Distributed DL



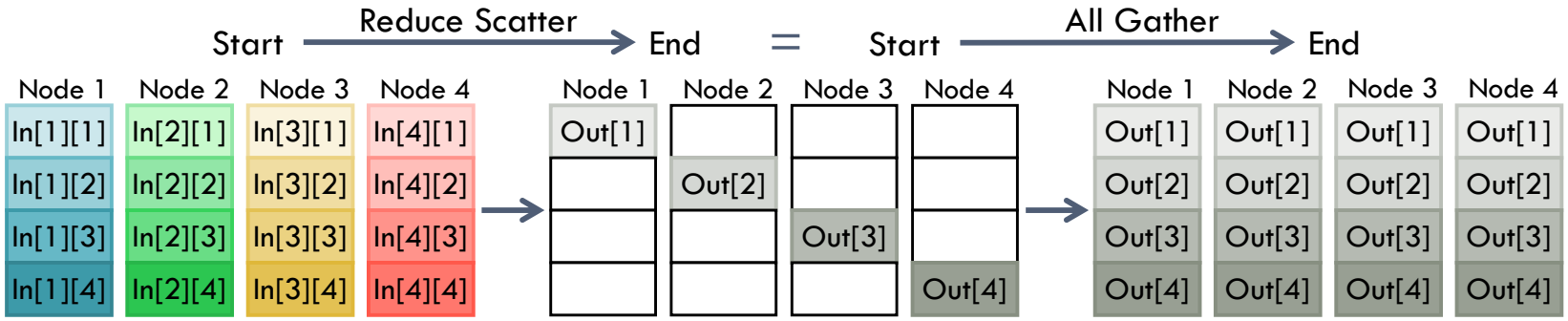
Reduce-Scatter

Collective Communications in Distributed DL



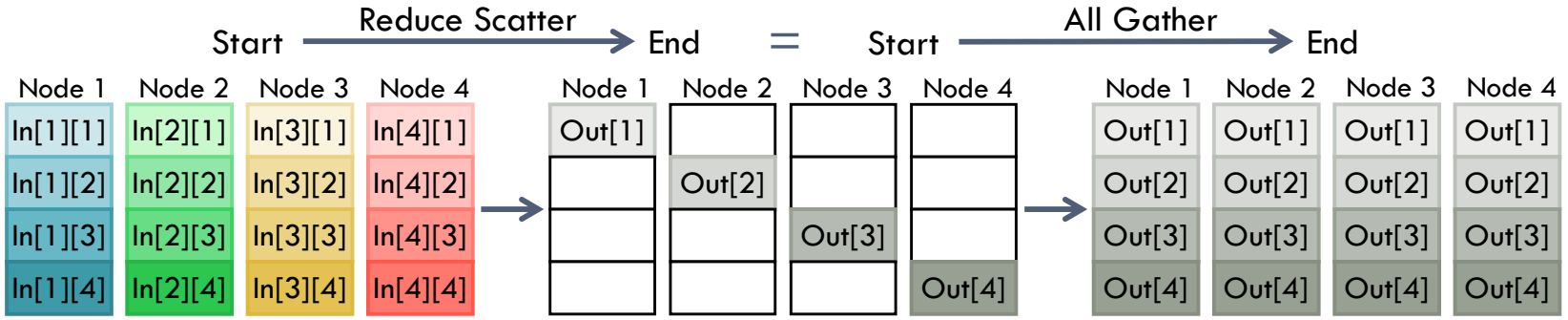
All-Gather

Collective Communications in Distributed DL

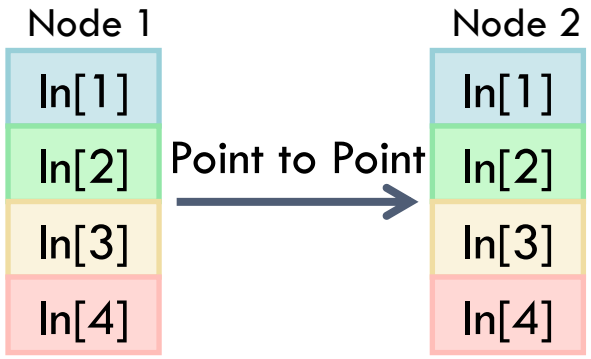


Reduce-Scatter, All-Gather, and All-Reduce

Collective Communications in Distributed DL

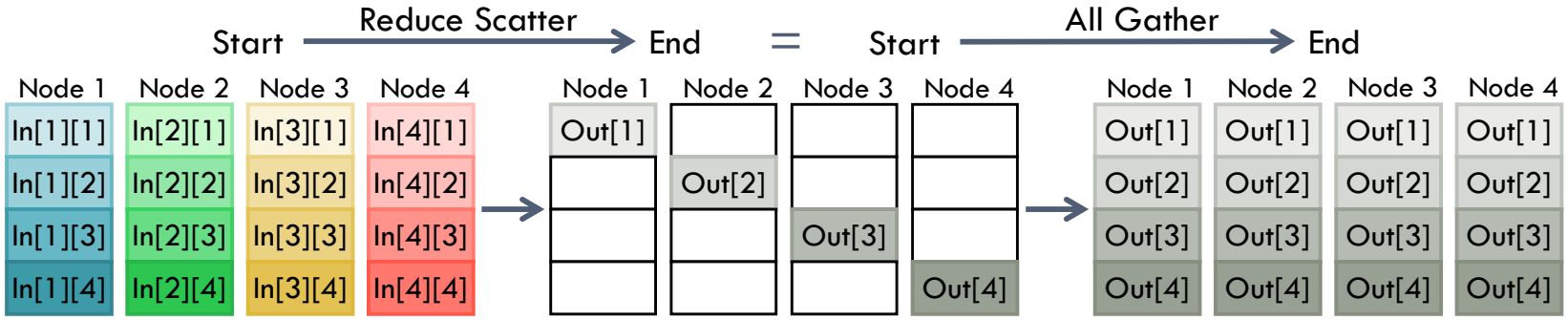


Reduce-Scatter, All-Gather, and All-Reduce

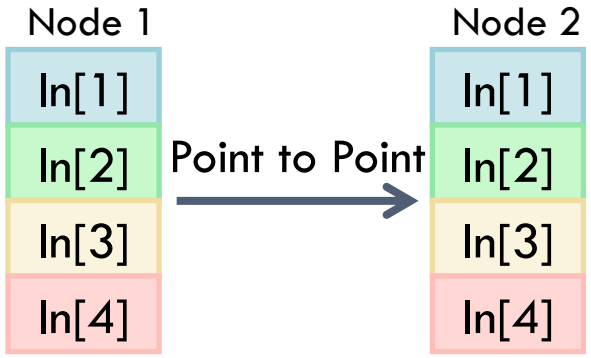


Point-to-Point

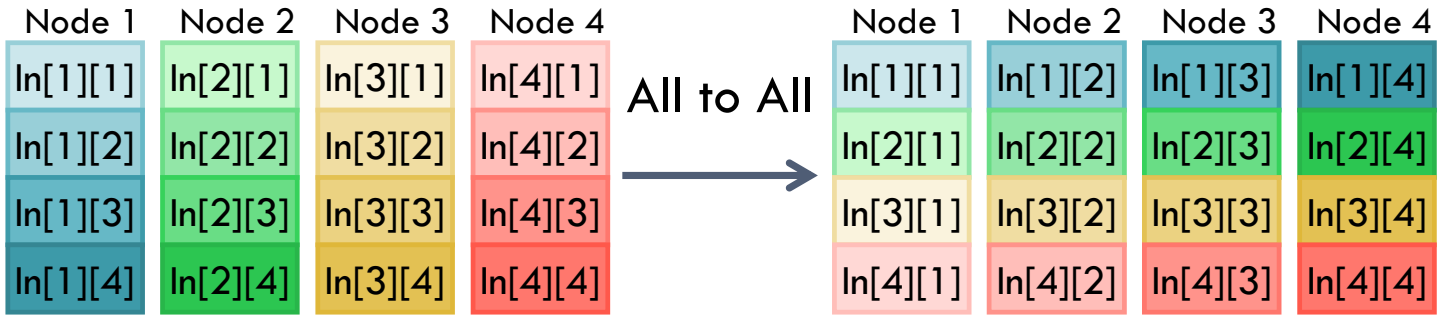
Collective Communications in Distributed DL



Reduce-Scatter, All-Gather, and All-Reduce

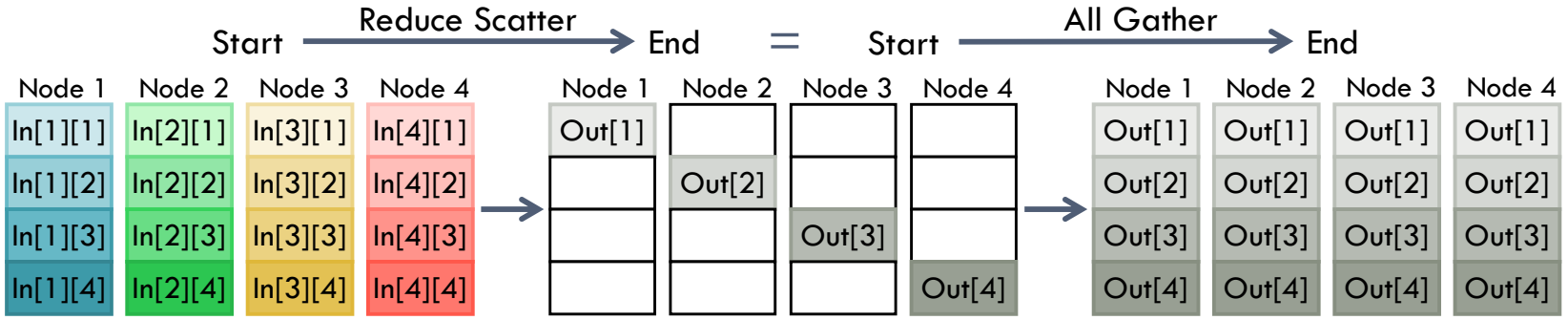


Point-to-Point

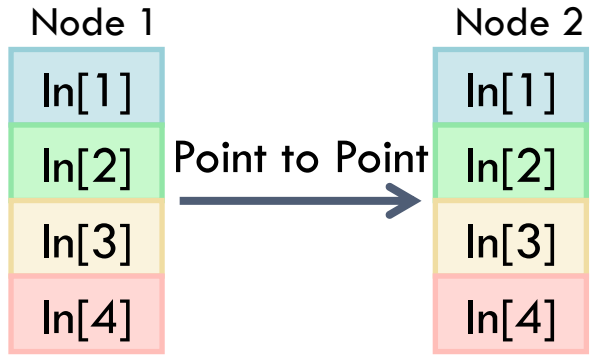


All-to-All

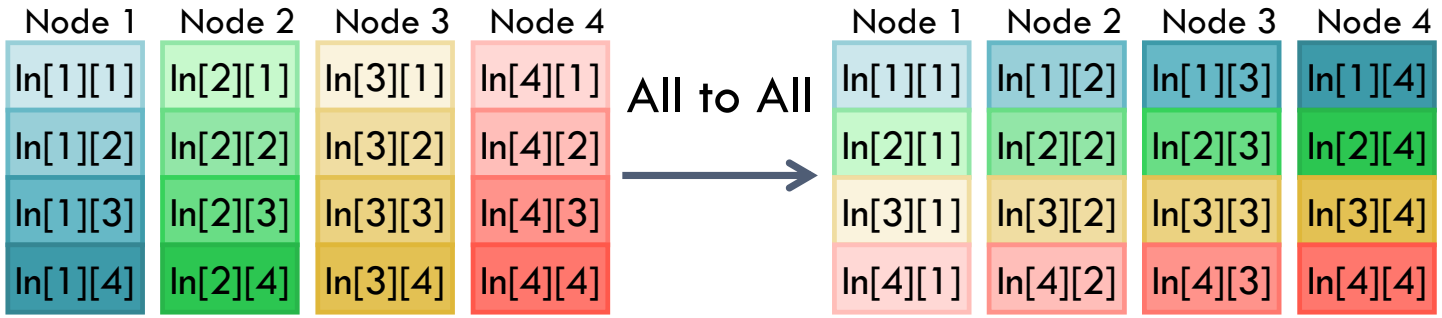
Collective Communications in Distributed DL



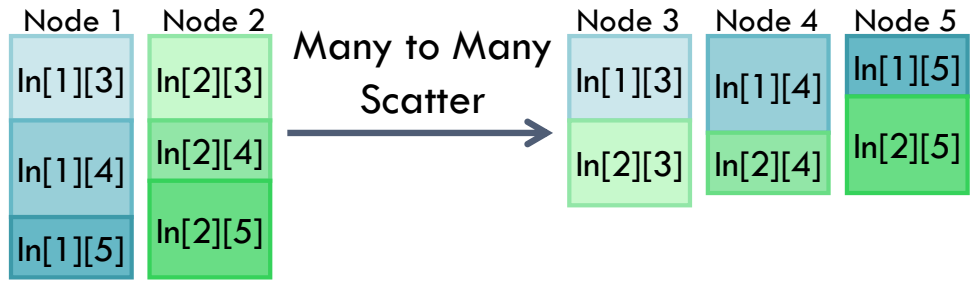
Reduce-Scatter, All-Gather, and All-Reduce



Point-to-Point

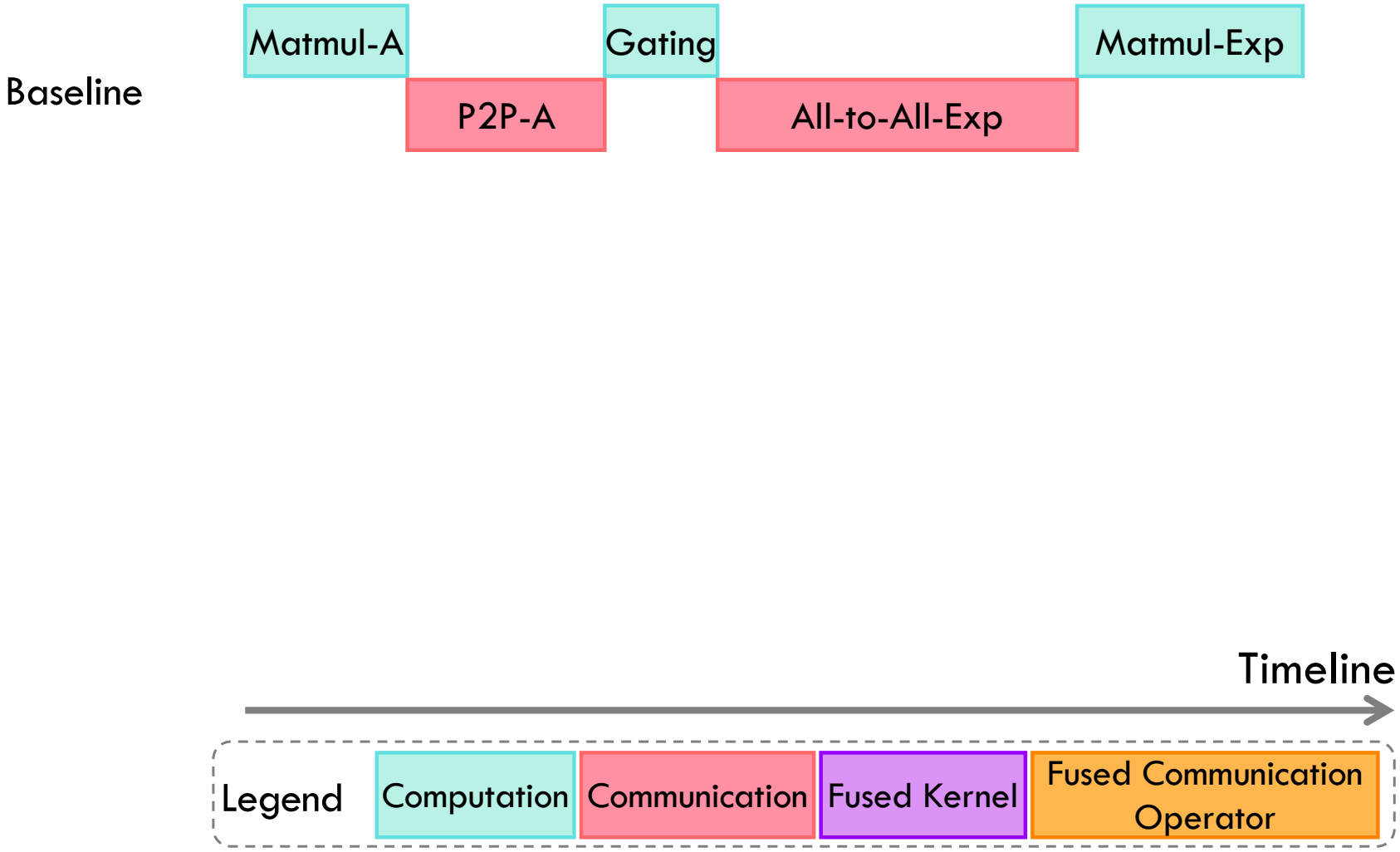


All-to-All

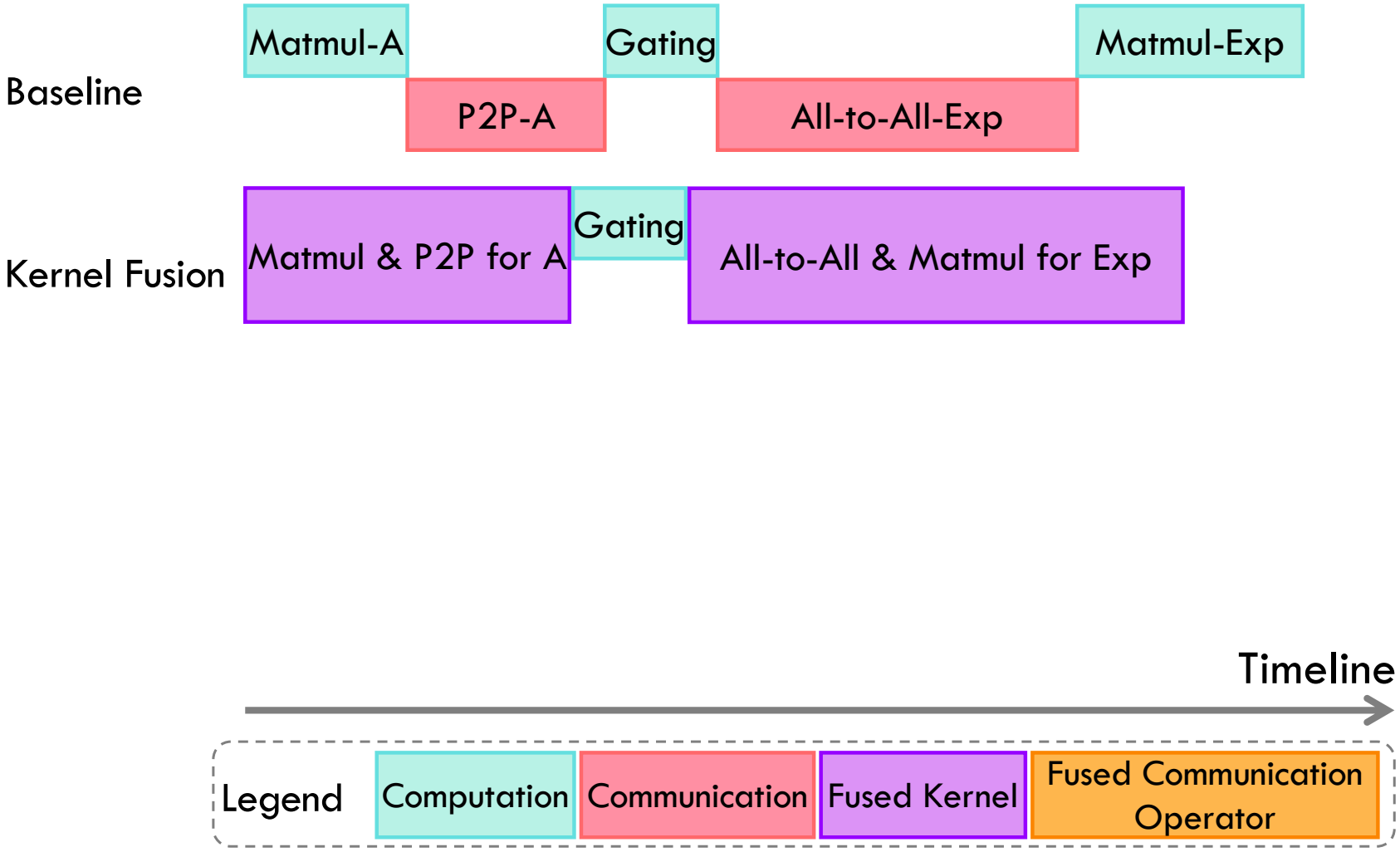


Many-to-Many Scatter

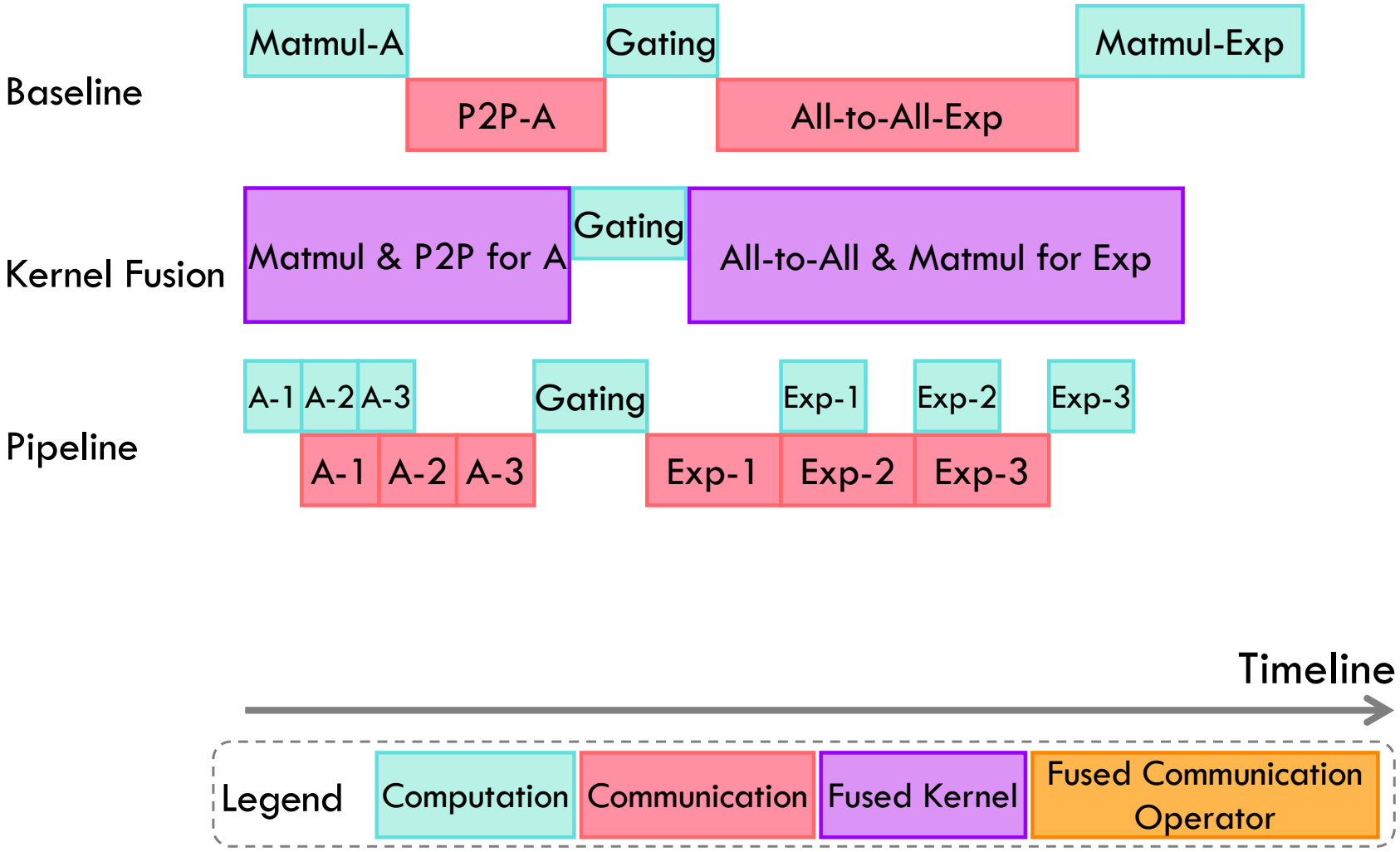
Communication for PP+EP



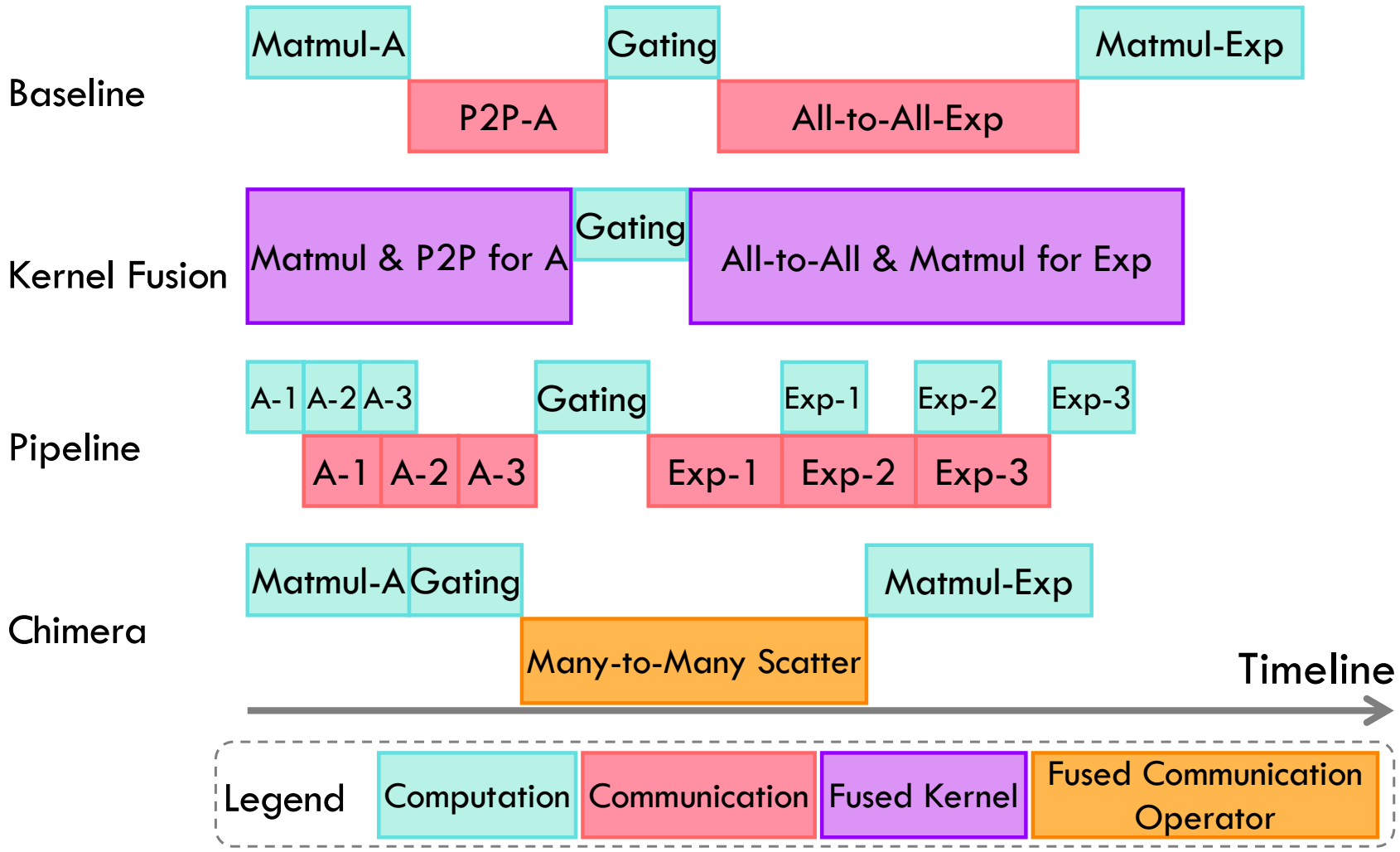
Communication for PP+EP



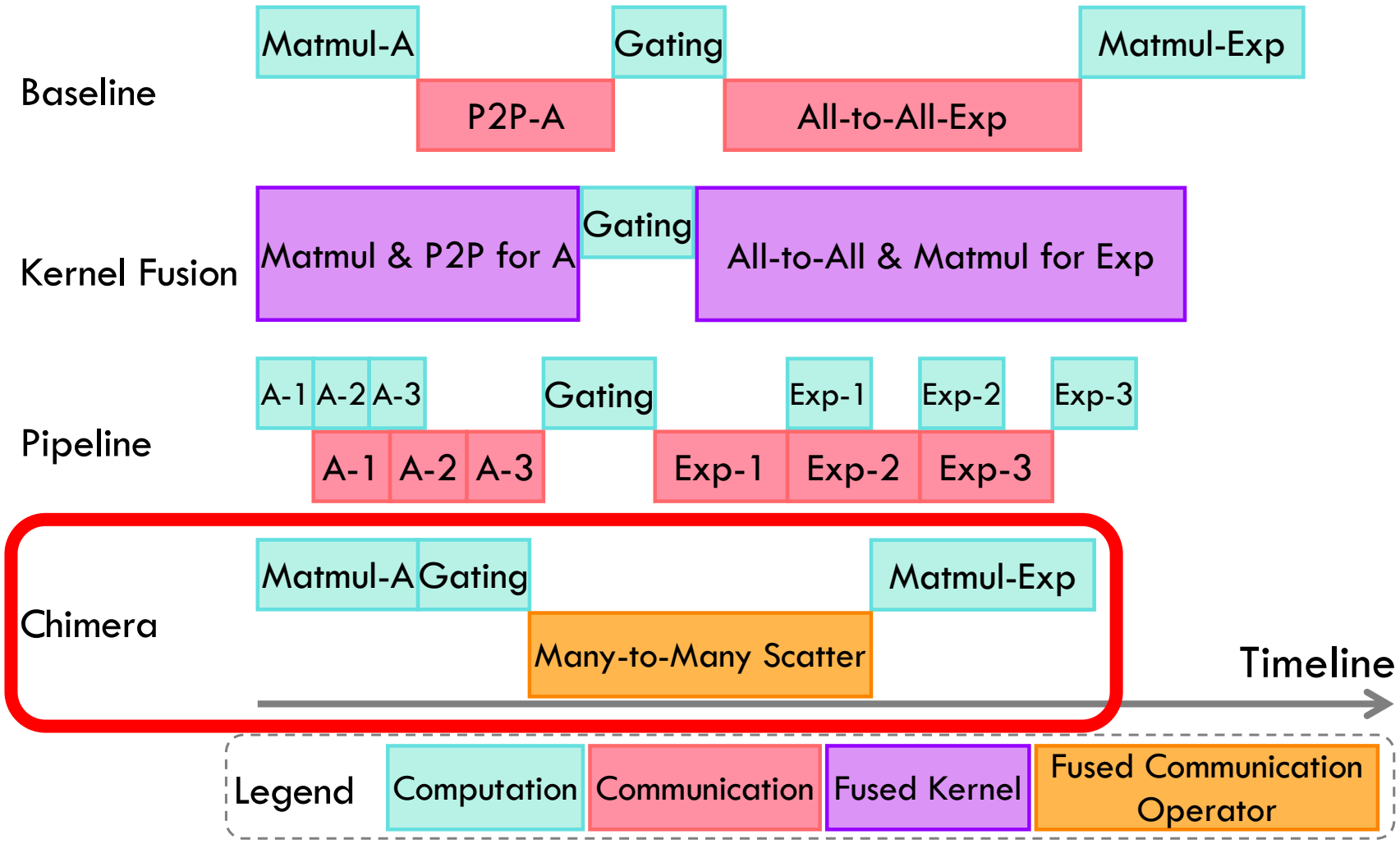
Communication for PP+EP



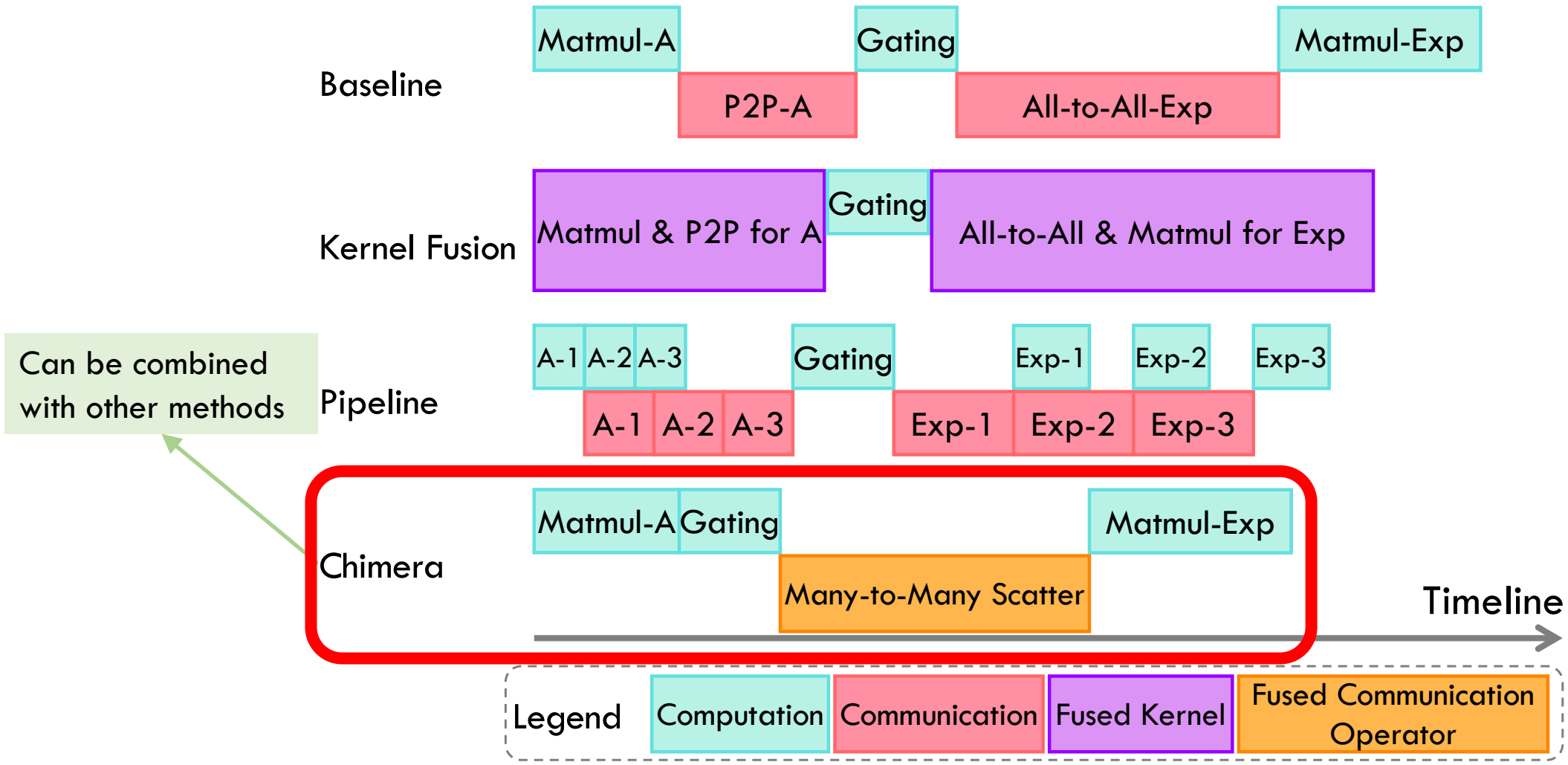
Communication for PP+EP



Communication for PP+EP



Communication for PP+EP



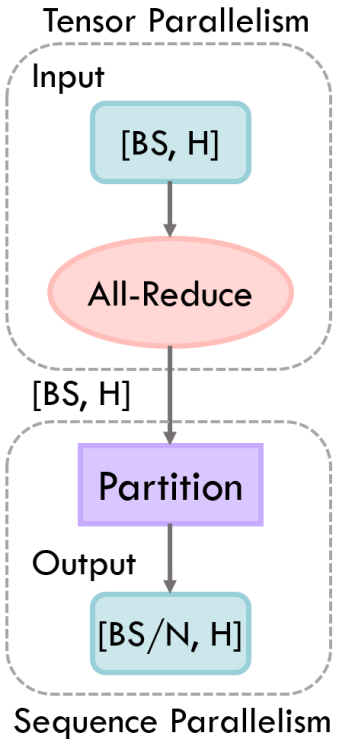
Communication Redundancy

- Some data transmissions are redundant

Communication Redundancy

- Some data transmissions are redundant

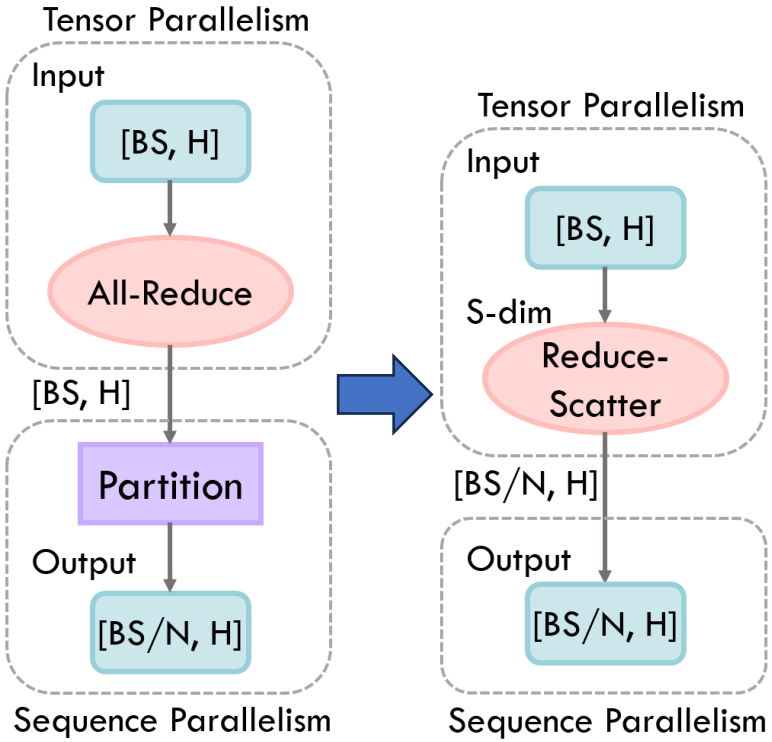
Case 1: TP+SP [Korthikanti+ MLSys '23]



Communication Redundancy

- Some data transmissions are redundant

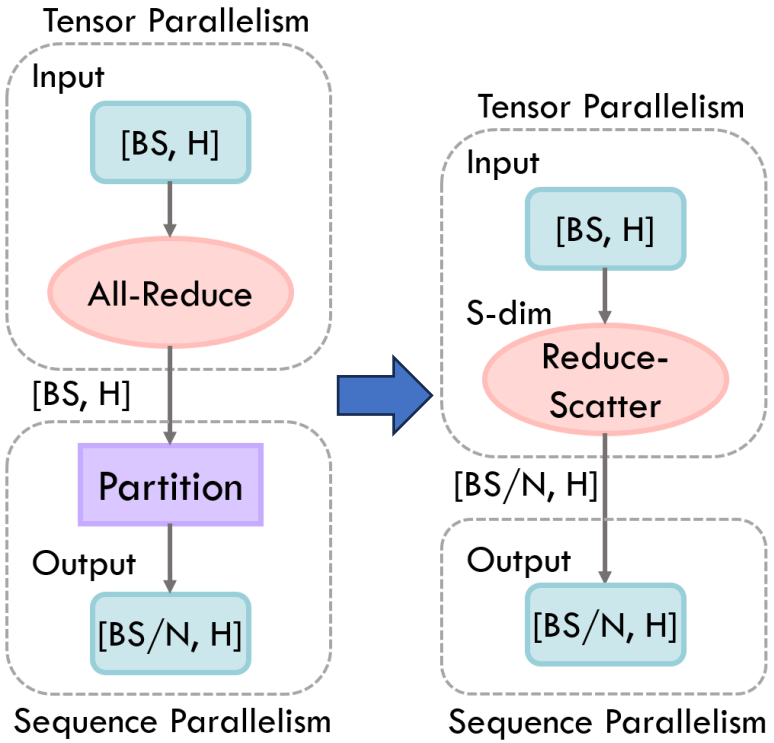
Case 1: TP+SP [Korthikanti+ MLSys '23]



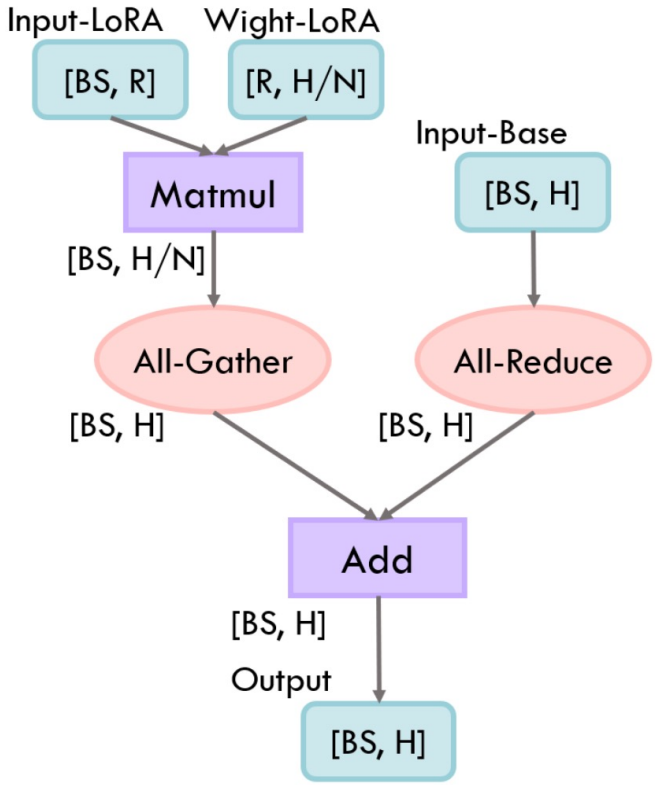
Communication Redundancy

- Some data transmissions are redundant

Case 1: TP+SP [Korthikanti+ MLSys '23]



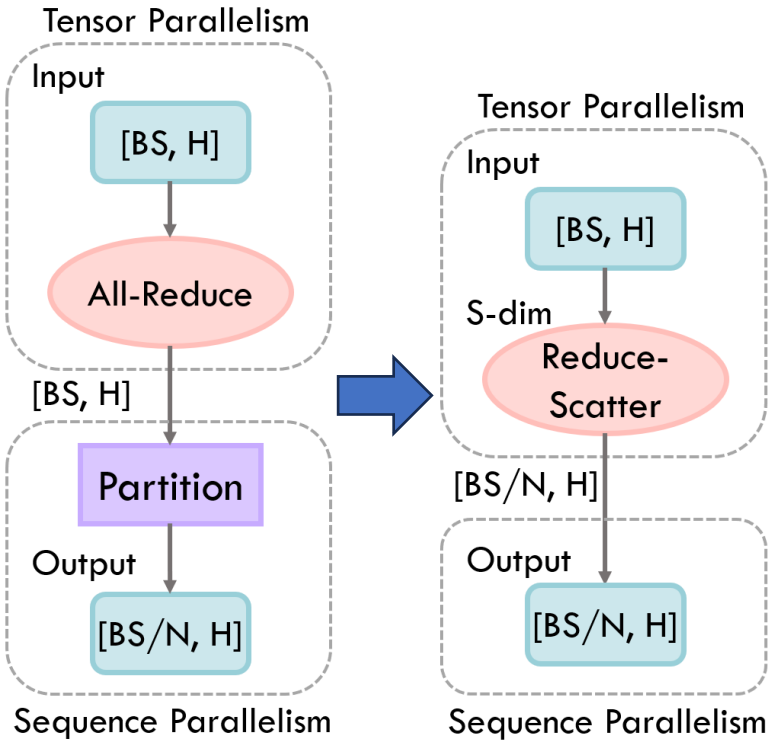
Case 2: S-LoRA [Sheng+ MLSys '23]



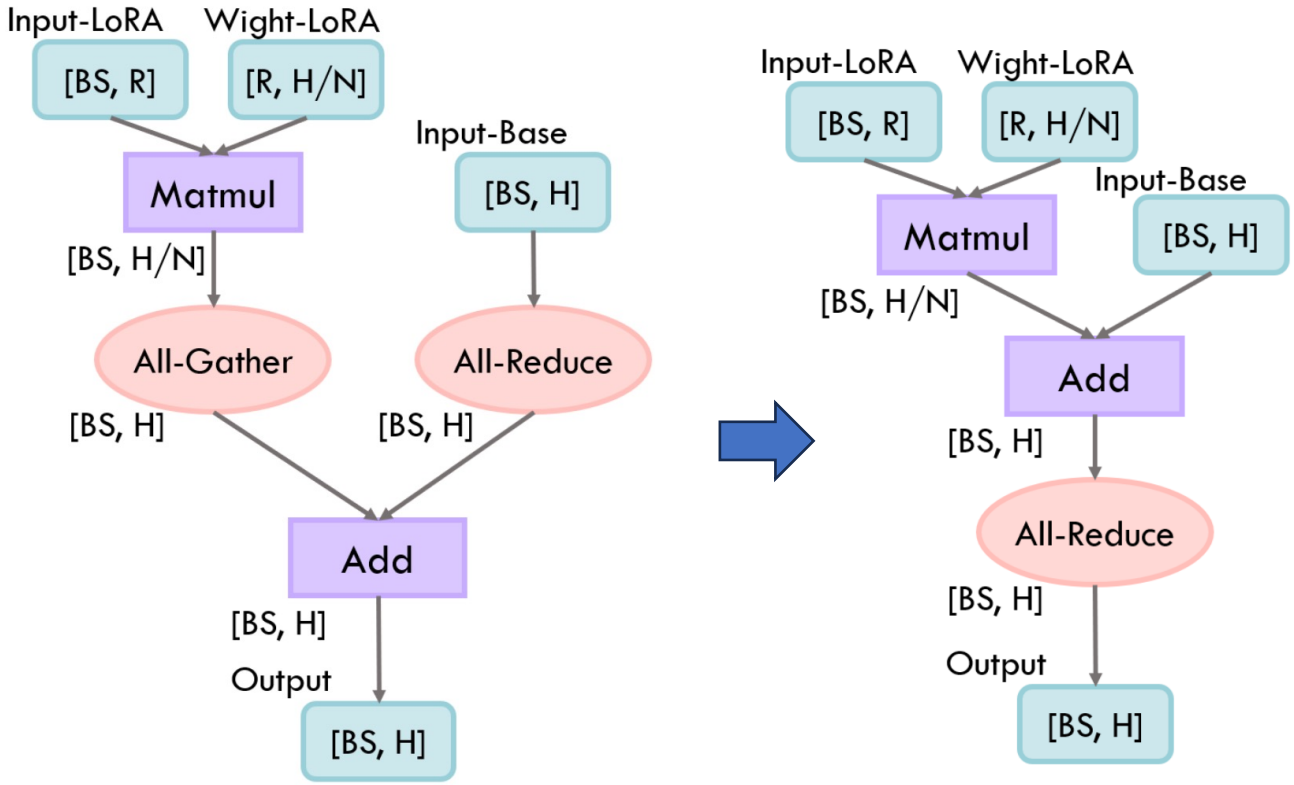
Communication Redundancy

- Some data transmissions are redundant

Case 1: TP+SP [Korthikanti+ MLSys '23]



Case 2: S-LoRA [Sheng+ MLSys '23]



Communication Modeling for Hybrid Parallelism

- Communication overhead
  - N: nodes number
  - B: batch size
  - S: sequence length
  - H: hidden dimension

Parallelism	Communication	Overhead
TP	All-Reduce	$2(N-1) \times \frac{BSH}{N}$
PP	P2P	BSH
SP	All-Gather	$(N-1) \times \frac{BSH}{N}$
EP	All-to-All	$(N-1) \times \frac{BSHk}{N}$
/	Reduce-Scatter	$(N-1) \times \frac{BSH}{N}$
/	Many-to-Many Scatter (M2MS)	BSH

Communication Modeling for Hybrid Parallelism

- Communication overhead
  - N: nodes number
  - B: batch size
  - S: sequence length
  - H: hidden dimension

Parallelism	Communication	Overhead
TP	All-Reduce	$2(N-1) \times \frac{BSH}{N}$
PP	P2P	BSH
SP	All-Gather	$(N-1) \times \frac{BSH}{N}$
EP	All-to-All	$(N-1) \times \frac{BSHk}{N}$
/	Reduce-Scatter	$(N-1) \times \frac{BSH}{N}$
/	Many-to-Many Scatter (M2MS)	BSH

Hybrid Parallelism	Communication Size 1	Communication Size 2	Real Demand	Ratio
TP + SP [34]	/	$2(N-1) \times \frac{BSH}{N}$	$(N-1) \times \frac{BSH}{N}$	50%
TP + PP [75]	$2(N_1-1) \times \frac{BSH}{N_1}$	$\frac{BSH}{N_1} + (N_2-1) \times \frac{BSH}{N_2}$	$BSH + (N_2-1) \times \frac{BSH}{N_2}$	60% – 80%
TP + EP	$2(N_1-1) \times \frac{BSH}{N_1}$	$(N_2-1) \times \frac{BSHk}{N_2}$	$(N_1-1) \times \frac{BSH}{N_1} + (N_2-1) \times \frac{BSHk}{N_2}$	66.7% – 100%
PP + EP [18]	BSH	$(N-1) \times \frac{BSHk}{N}$	BSH	0% – 66.7%
SP + PP	$(N-1) \times \frac{BSH}{N}$	BSH	BSH	50% – 66.7%
SP + EP	$(N_1-1) \times \frac{BSH}{N_1}$	$(N_2-1) \times \frac{BSHk}{N_2}$	$(N_2-1) \times \frac{BSHk}{N_2}$	50% – 100%

Communication Modeling for Hybrid Parallelism

- Communication overhead
  - N: nodes number
  - B: batch size
  - S: sequence length
  - H: hidden dimension

Parallelism	Communication	Overhead
TP	All-Reduce	$2(N-1) \times \frac{BSH}{N}$
PP	P2P	BSH
SP	All-Gather	$(N-1) \times \frac{BSH}{N}$
EP	All-to-All	$(N-1) \times \frac{BSHk}{N}$
/	Reduce-Scatter	$(N-1) \times \frac{BSH}{N}$
/	Many-to-Many Scatter (M2MS)	BSH

Hybrid Parallelism	Communication Size 1	Communication Size 2	Real Demand	Ratio
TP + SP [34]	/	$2(N-1) \times \frac{BSH}{N}$	$(N-1) \times \frac{BSH}{N}$	50%
TP + PP [75]	$2(N_1-1) \times \frac{BSH}{N_1}$	$\frac{BSH}{N_1} + (N_2-1) \times \frac{BSH}{N_2}$	$BSH + (N_2-1) \times \frac{BSH}{N_2}$	60% – 80%
TP + EP	$2(N_1-1) \times \frac{BSH}{N_1}$	$(N_2-1) \times \frac{BSHk}{N_2}$	$(N_1-1) \times \frac{BSH}{N_1} + (N_2-1) \times \frac{BSHk}{N_2}$	66.7% – 100%
PP + EP [18]	BSH	$(N-1) \times \frac{BSHk}{N}$	BSH	0% – 66.7%
SP + PP	$(N-1) \times \frac{BSH}{N}$	BSH	BSH	50% – 66.7%
SP + EP	$(N_1-1) \times \frac{BSH}{N_1}$	$(N_2-1) \times \frac{BSHk}{N_2}$	$(N_2-1) \times \frac{BSHk}{N_2}$	50% – 100%

Communication Modeling for Hybrid Parallelism

- Communication overhead
  - N: nodes number
  - B: batch size
  - S:
  - H:

Parallelism	Communication	Overhead
TP	All-Reduce	$2(N-1) \times \frac{BSH}{N}$
PP	P2P	BSH
SP	All-Gather	$(N-1) \times \frac{BSH}{N}$
		$) \times \frac{BSHk}{N}$
		$1) \times \frac{BSH}{N}$
		BSH

Redundancy is introduced during parallelism transitioning

Hyb				Ratio
				50%
TP + PP [75]	$2(N_1-1) \times \frac{}{N_1}$	$\frac{}{N_1} + (N_2-1) \times \frac{}{N_2}$	$BSH + (N_2-1) \times \frac{}{N_2}$	60% – 80%
TP + EP	$2(N_1-1) \times \frac{BSH}{N_1}$	$(N_2-1) \times \frac{BSHk}{N_2}$	$(N_1-1) \times \frac{BSH}{N_1} + (N_2-1) \times \frac{BSHk}{N_2}$	66.7% – 100%
PP + EP [18]	BSH	$(N-1) \times \frac{BSHk}{N}$	BSH	0% – 66.7%
SP + PP	$(N-1) \times \frac{BSH}{N}$	BSH	BSH	50% – 66.7%
SP + EP	$(N_1-1) \times \frac{BSH}{N_1}$	$(N_2-1) \times \frac{BSHk}{N_2}$	$(N_2-1) \times \frac{BSHk}{N_2}$	50% – 100%

Fuse Two Operators to One

- 5 basic communication operators
  - Point-to-Point (P2P)
  - Reduce-Scatter (R-S)
  - All-Gather (A-G)
  - All-to-All (A2A)
  - Many-to-Many Scatter (M2MS)

- Process
  1. Divide All-Reduce to Reduce-Scatter and All-Gather
  2. Replace adjacent operators with one to remove redundancy

Way 1			Way 2		
Adjacent Comm	**Fused Comm**	**Overhead**	**Adjacent Comm**	**Fused Comm**	**Overhead**
R-S+A-G	A-R	equal	A-G+R-S	Zero	lower
R-S+P2P	M2MS	equal	P2P+R-S	M2MS	lower
R-S+M2MS	M2MS	equal	M2MS+R-S	M2MS	lower
R-S+A2A	R-S	lower	A2A+R-S	R-S	lower
A-G+P2P	M2MS	lower ★	P2P+A-G	M2MS	equal
A-G+M2MS	M2MS	lower ★	M2MS+A-G	M2MS	equal
A-G+A2A	A2A	lower ★	A2A+A-G	A2A	lower
P2P+M2MS	M2MS	lower	M2MS+P2P	M2MS	lower
P2P+A2A	M2MS	lower ★	A2A+P2P	M2MS	lower ★
M2MS+A2A	M2MS	lower	A2A+M2MS	M2MS	lower
R-S+R-S	N/A	equal	A-G+A-G	A-G	lower
P2P+P2P	P2P	lower	M2MS+M2MS	M2MS	lower
A2A+A2A	A2A	lower ★			

★ represents communication in real hybrid parallelism

Fuse Two Operators to One

- 5 basic communication operators
  - Point-to-Point (P2P)
  - Red
  - All-
  - All-
  - Ma
- Process
  1. Divide All-Reduce to Reduce-Scatter and All-Gather
  2. Replace adjacent operators with one to remove redundancy

Fuse adjacent communication operators to remove redundancy

Way 1			Way 2		
Adjacent Comm	**Fused Comm**	**Overhead**	**Adjacent Comm**	**Fused Comm**	**Overhead**
R-S+A-G	A-R	equal	A-G+R-S	Zero	lower
				MS	lower
				MS	lower
				-S	lower
				MS	equal
				MS	equal
				2A	lower
P2P+M2MS	M2MS	lower	M2MS+P2P	M2MS	lower
P2P+A2A	M2MS	lower ★	A2A+P2P	M2MS	lower ★
M2MS+A2A	M2MS	lower	A2A+M2MS	M2MS	lower
R-S+R-S	N/A	equal	A-G+A-G	A-G	lower
P2P+P2P	P2P	lower	M2MS+M2MS	M2MS	lower
A2A+A2A	A2A	lower ★			

★ represents communication in real hybrid parallelism

Fuse Two Operators to One

- 5 basic communication operators
  - Point-to-Point (P2P)
  - Red
  - All-
  - All-
  - Ma
- Process
  1. Divide A
     Scatter
  2. Replace adjacent operators with one to remove redundancy

Fuse adjacent communication operators to remove redundancy

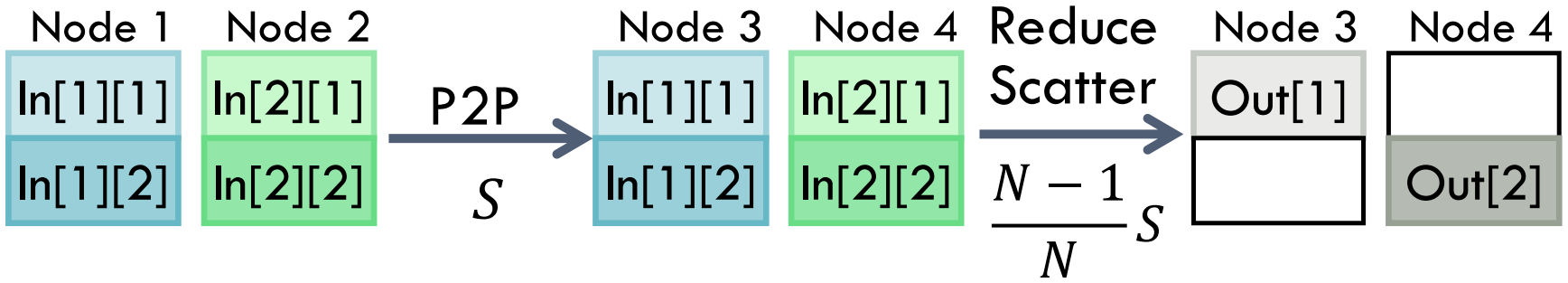
More details in the paper

Way 1			Way 2		
Adjacent Comm	**Fused Comm**	**Overhead**	**Adjacent Comm**	**Fused Comm**	**Overhead**
R-S+A-G	A-R	equal	A-G+R-S	Zero	lower
				MS	lower
				MS	lower
				-S	lower
				MS	equal
				MS	equal
				2A	lower
P2P+M2MS	M2MS	lower	M2MS+P2P	M2MS	lower
			+P2P	M2MS	lower ★
			-M2MS	M2MS	lower
			+A-G	A-G	lower
P2P+P2P	P2P	lower	M2MS+M2MS	M2MS	lower
A2A+A2A	A2A	lower ★			

★ represents communication in real hybrid parallelism

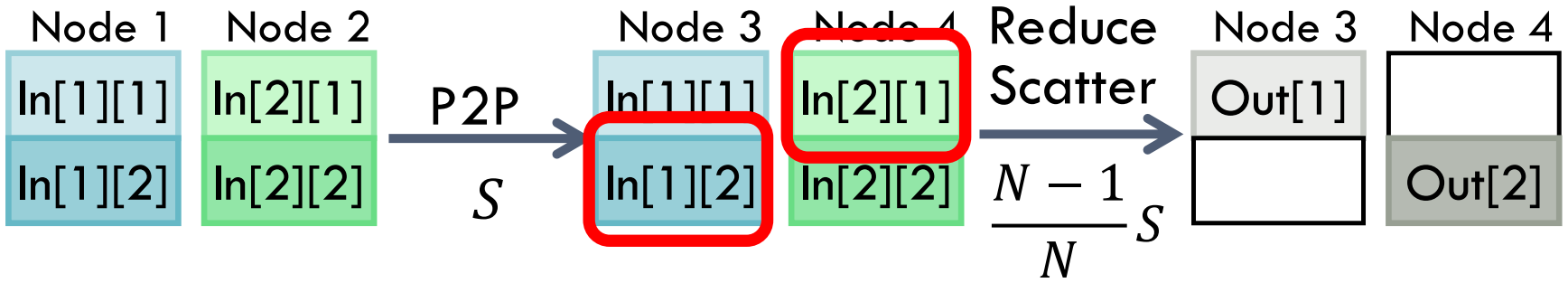
A Fusion Case： P2P + Reduce-Scatter (RS)

- P2P, then Reduce-Scatter
  - In P2P, destination nodes receive unneeded intermediate data



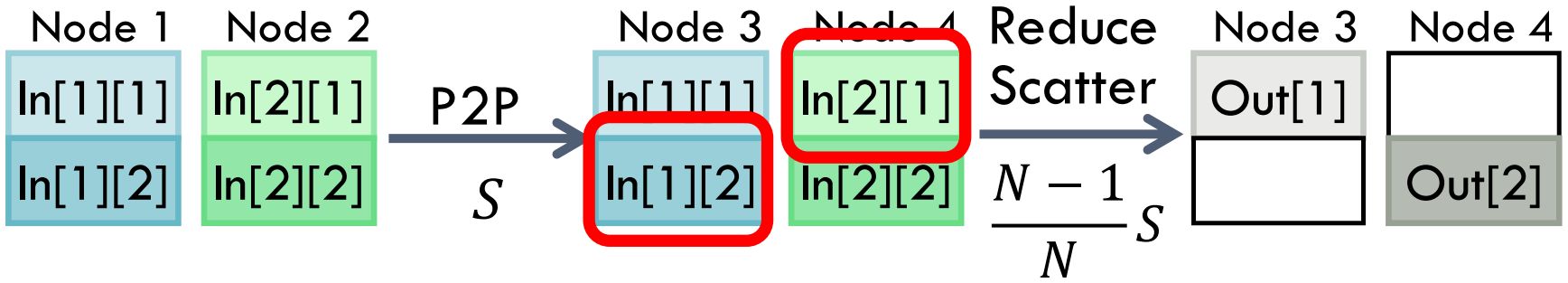
A Fusion Case： P2P + Reduce-Scatter (RS)

- P2P, then Reduce-Scatter
  - In P2P, destination nodes receive unneeded intermediate data

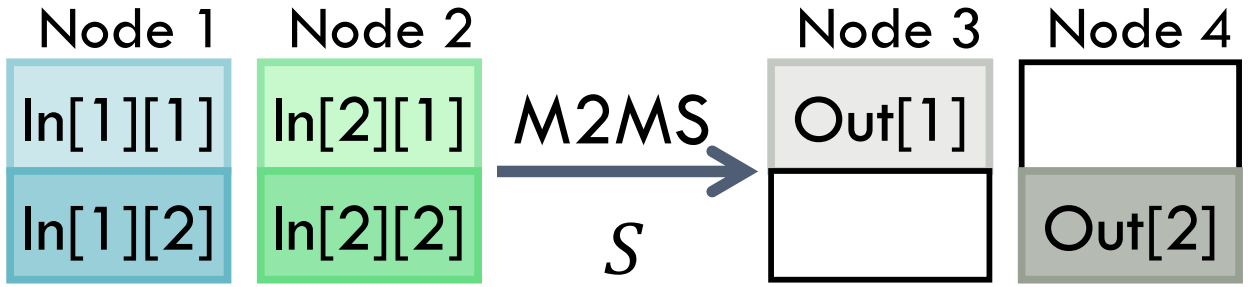


A Fusion Case: P2P + Reduce-Scatter (RS)

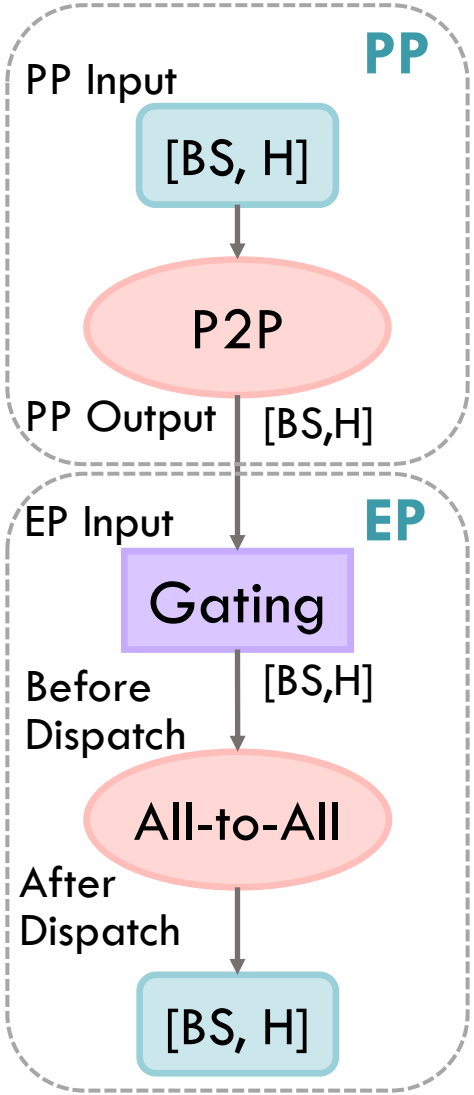
- P2P, then Reduce-Scatter
  - In P2P, destination nodes receive unneeded intermediate data



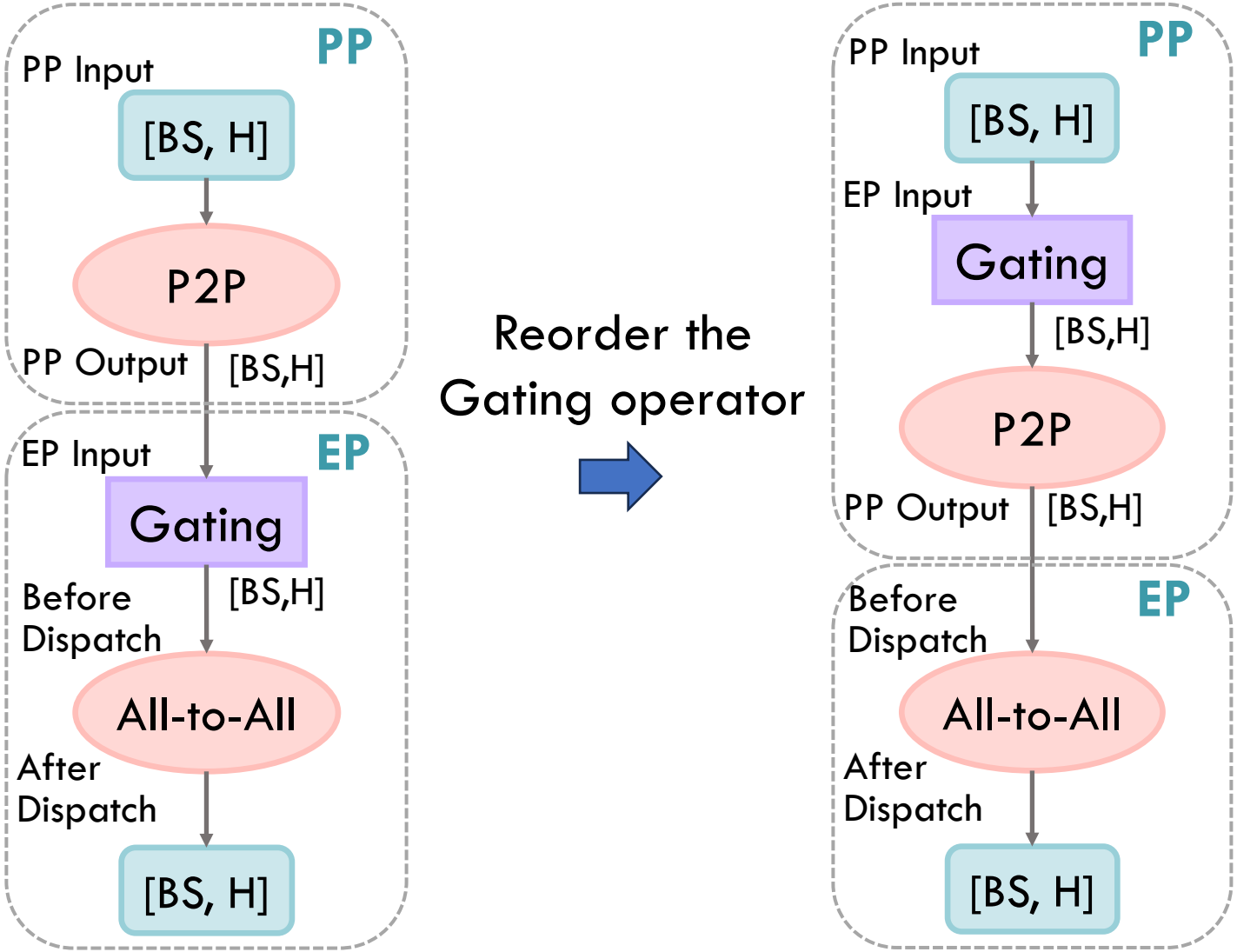
- Use Many-to-Many Scatter (M2MS) to send the data directly



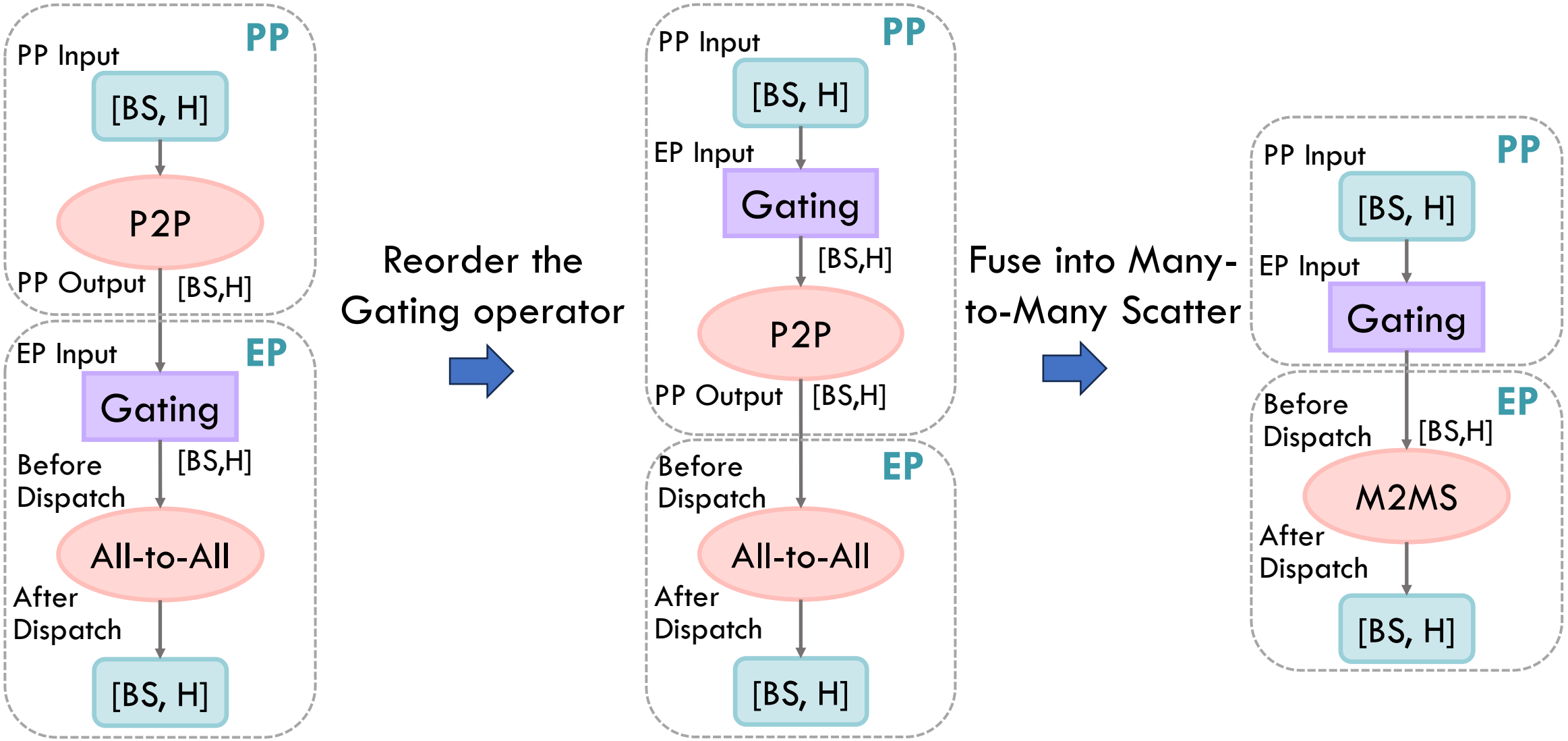
Case Study I: PP+EP



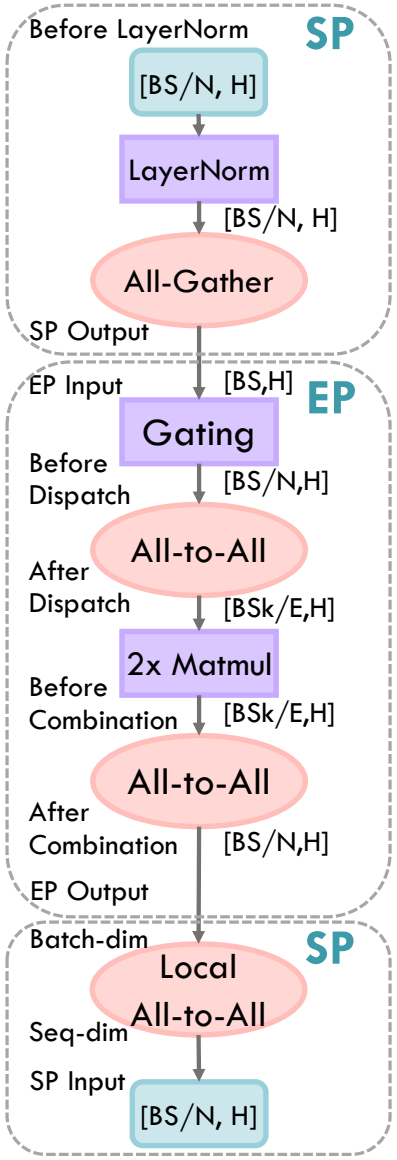
Case Study I: PP+EP



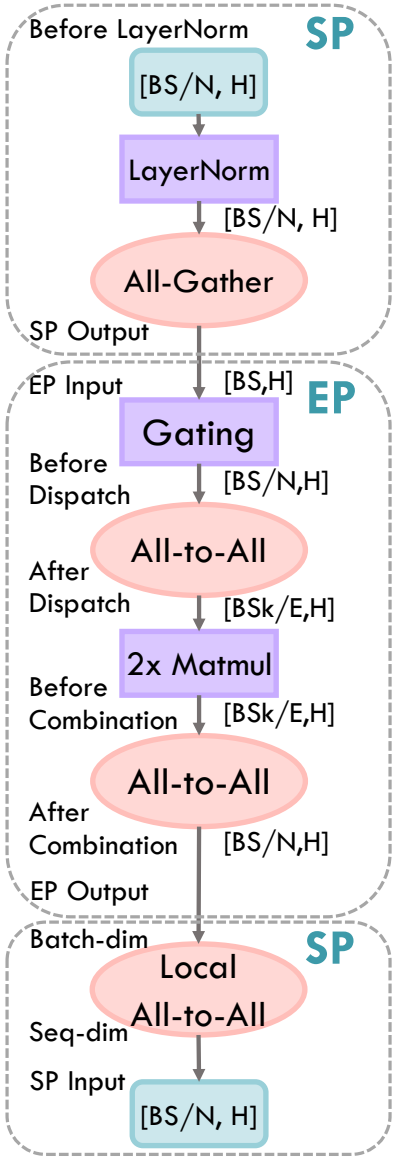
Case Study I: PP+EP



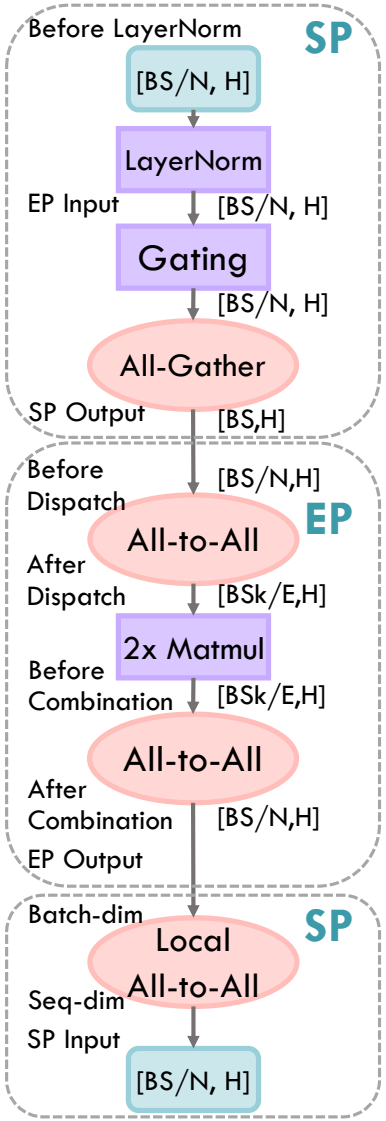
Case Study II: SP+EP



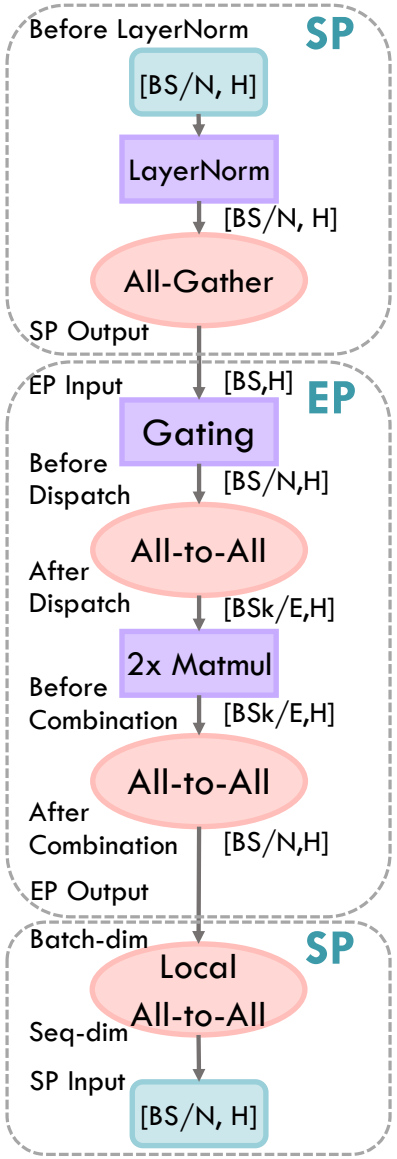
Case Study II: SP+EP



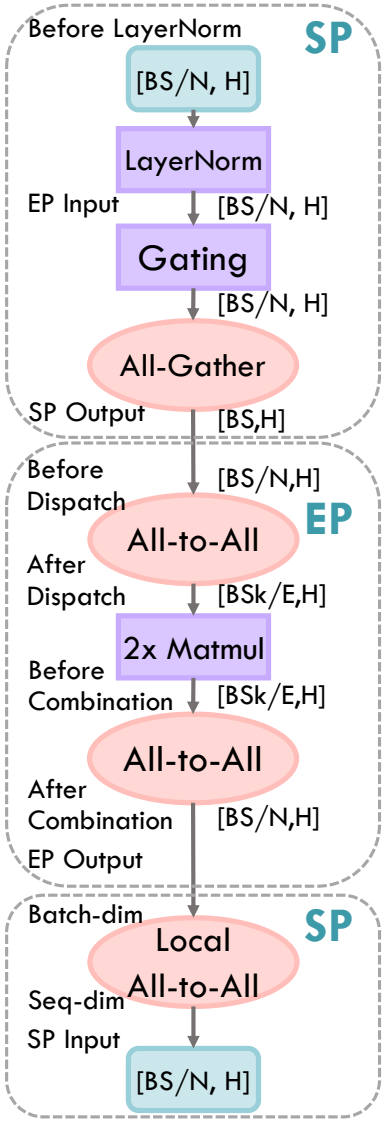
Reorder the Gating operator



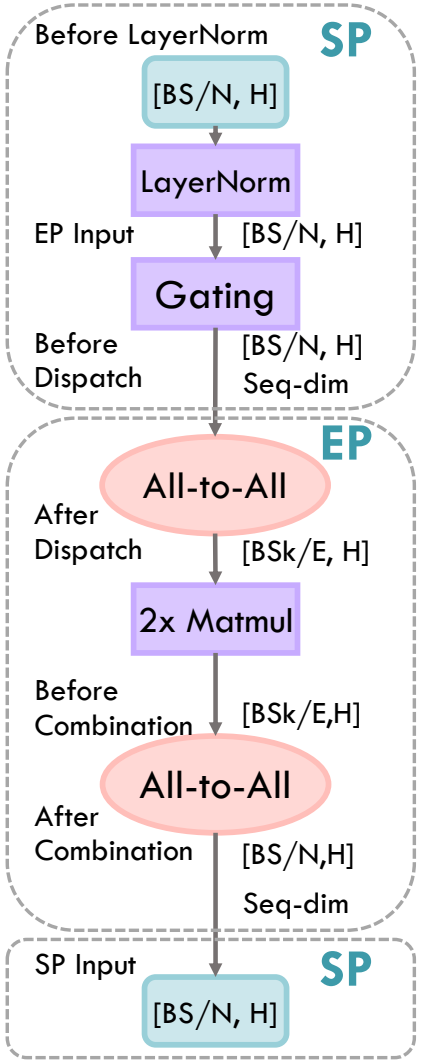
Case Study II: SP+EP



Reorder the Gating operator



Fuse into All-to-All



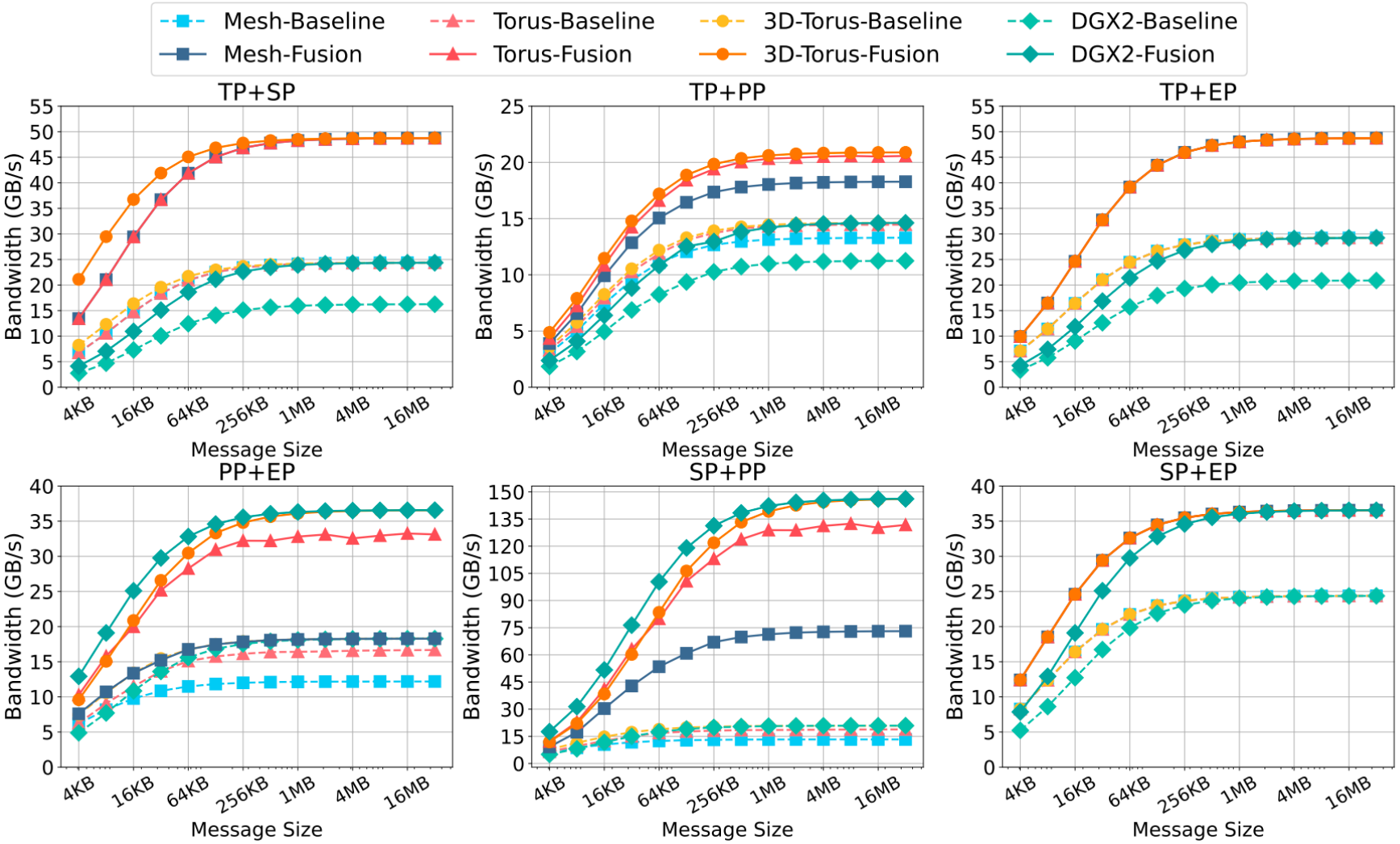
Evaluation

- **Simulation Tools**
  - SCALE-Sim v2
  - BookSim2
- **Real-World Testing**
  - 8 RTX 4080 GPU
  - PCIe 4.0
  - PyTorch 2.5.1
  - CUDA 12.4
  - NCCL 2.21.5

Simulation Configurations

Parameter		Configuration
PE	MAC array	256×256
	Dataflow	Output Stationary
	Precision	32 bits
Accelerator	Number of PEs	16
	Clock	1 GHz
Network	Number of Accelerators	small: 5×5, 4×4, 2×2×2, 8×2 large: 8×8, 8×8, 4×4×4, 32×2
	Topology	2D-Mesh, 2D/3D-Torus, Fat-Tree
	Algorithm	Ring-2D, Halving Doubling
	Flow Control	Virtual Cut-Through
	Router Clock	1 GHz
	Number of VCs	4
	VC Buffer Depth	318 flits
	Data Packet Payload	256 Bytes
	Bandwidth	50 GBps
	Link Latency	100 ns

Synthetic Experiment: 1.23x – 7.06x Speedups

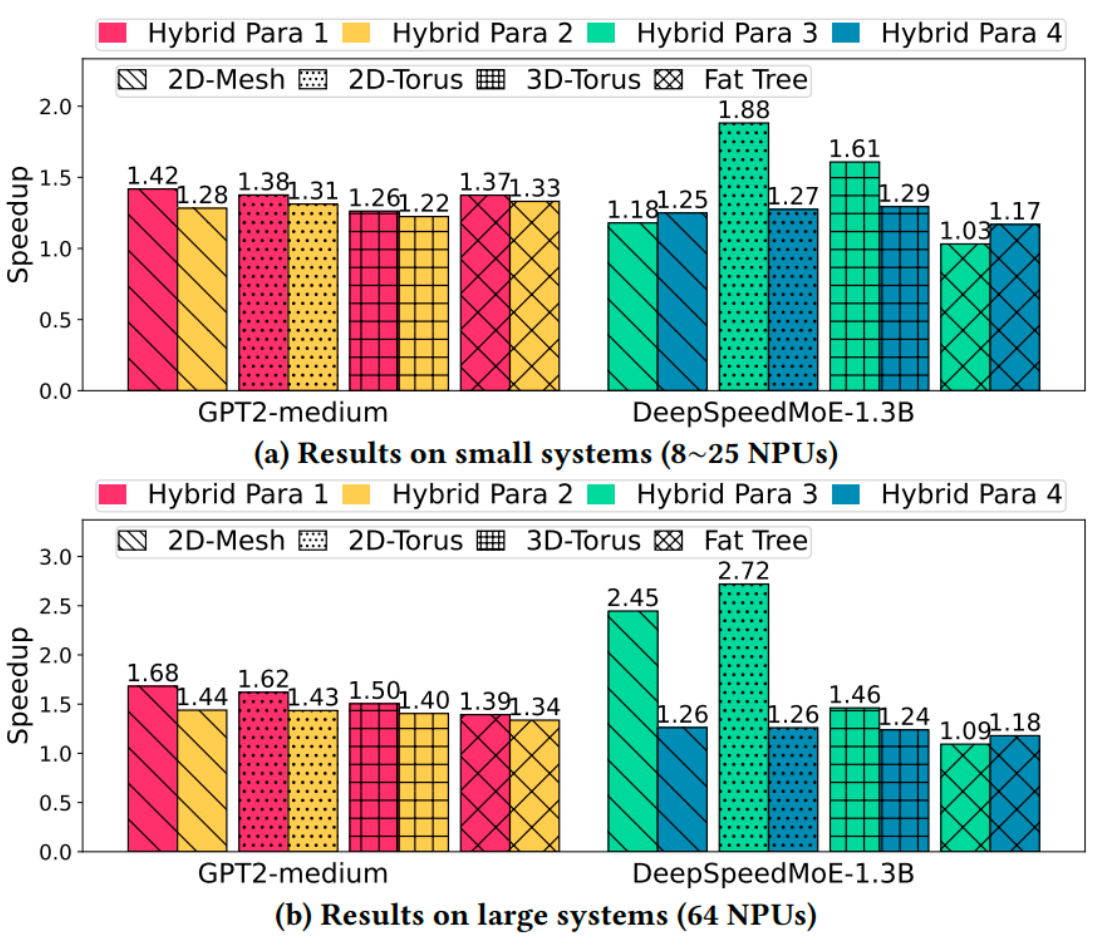


End-to-End Performance Evaluation Settings

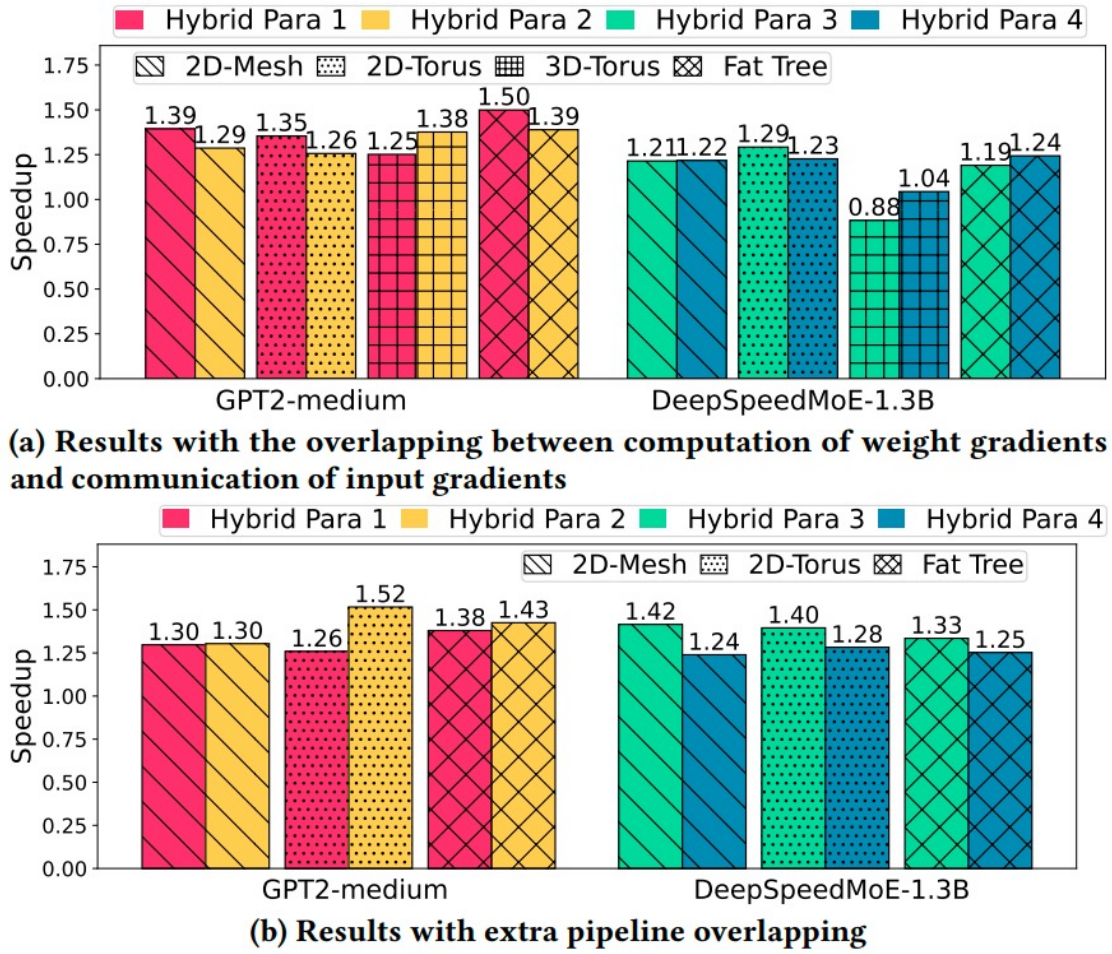
- Models: GPT2, DeepSpeed-MoE
- Topologies: 5×5 mesh, 4×4 torus, 8×2 fat-tree, 2×2×2 torus

Type	Parallelism Transitions	DP	TP	SP	PP	EP
HP 1	TP+SP, SP+PP	1	2	2	12/8/4	0
HP 2	TP+SP, TP+PP	1	2	2	12/8/4	0
HP 3	TP+SP, SP+EP, SP+PP	2	2	2	6/4/2	2
HP 4	TP+SP, TP+EP, TP+PP	2	2	2	6/4/2	2

End-to-End: 1.16x – 1.58x Average Speedups

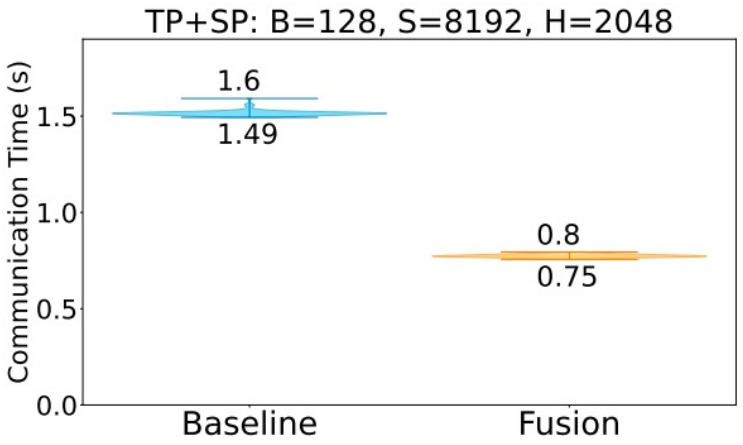


End-to-End Speedups for Forward Pass

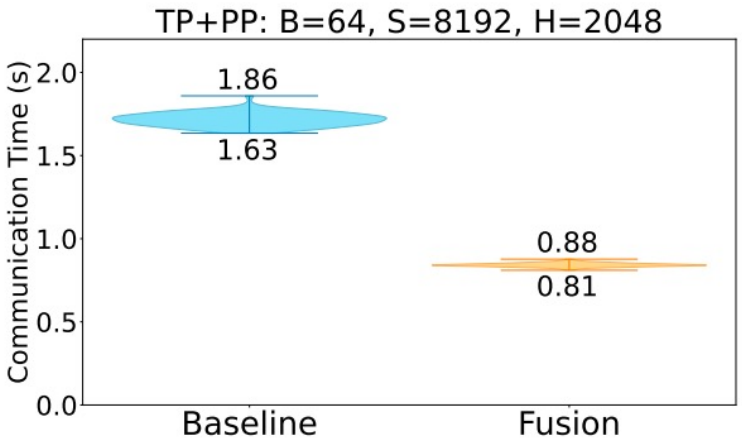


End-to-End Speedups for Backward Pass

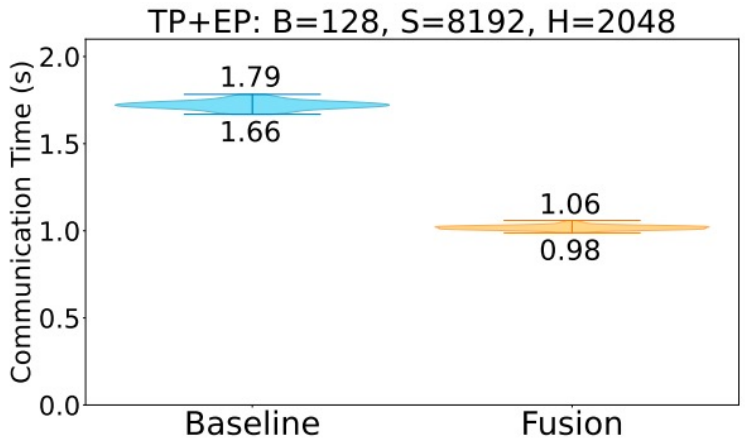
Real Machine Tests: 1.32x – 3.0x Speedups



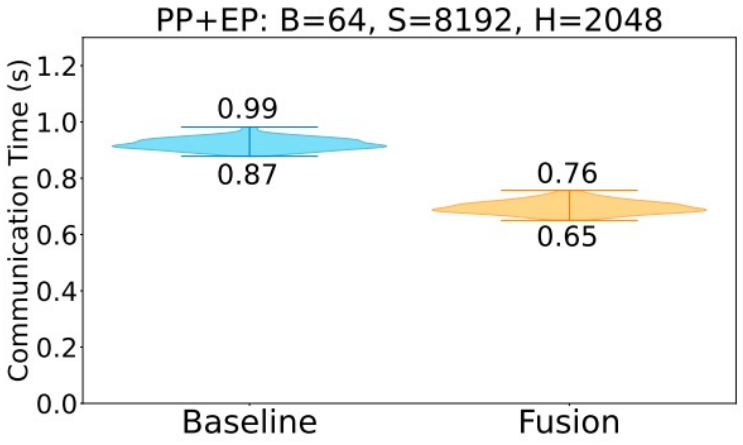
(a) Comparison of TP+SP



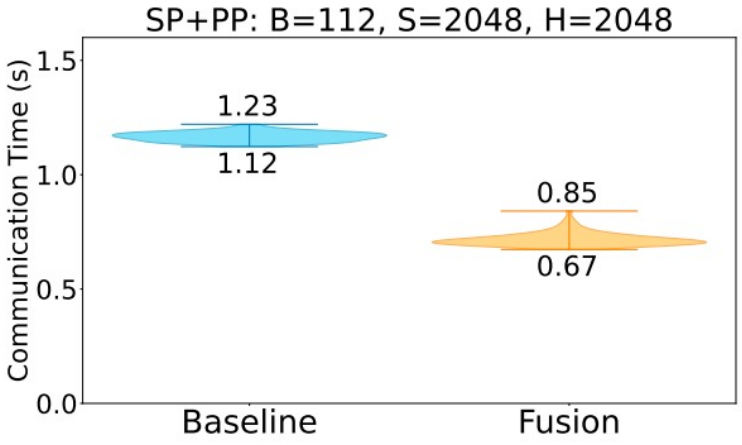
(b) Comparison of TP+PP



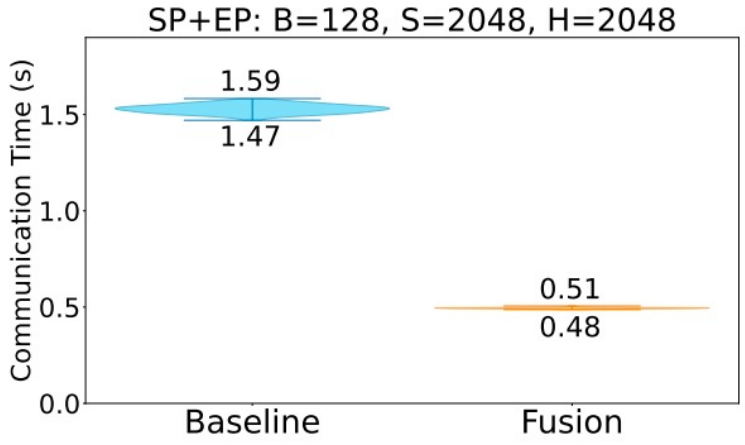
(c) Comparison of TP+EP



(d) Comparison of PP+EP



(e) Comparison of SP+PP



(f) Comparison of SP+EP

Summary

Summary

- **Problem:** We identify the communication bottleneck in current hybrid parallelism of LLM training and inference.

Summary

- **Problem:** We identify the communication bottleneck in current hybrid parallelism of LLM training and inference.
- **Inspiration:** Redundant data movements exist in hybrid parallelism of distributed LLMs.

Summary

- **Problem:** We identify the communication bottleneck in current hybrid parallelism of LLM training and inference.
- **Inspiration:** Redundant data movements exist in hybrid parallelism of distributed LLMs.

- **Chimera:**
  - Analysis of the communication patterns and redundant data movement
  - Communication fusion mechanism for hybrid parallel LLMs
  - Extensive evaluations on common LLM models, multi-NPU systems and hybrid parallelism modes

Thanks and Questions

Email: lqin674@connect.hkust-gz.edu.cn

"Chimera: Communication Fusion for Hybrid Parallelism in Large Language Models"
Le Qin, Junwei Cui, Weilin Cai, Jiayi Huang, ISCA 2025.

香港科技大学(广州)
THE HONG KONG UNIVERSITY OF SCIENCE
AND TECHNOLOGY (GUANGZHOU)