Centauri

Enabling Efficient Scheduling for Communication-Computation Overlap in Large Model Training via Communication Partitioning

Chen, Chang (Peking University) and Li, Xiuhong* (Peking University) and Zhu, Qianchao and Duan, Jiangfei and Sun, Peng and Zhang, Xingcheng and Yang, Chao

PKU, CUHK, SHANGHAI AI LAB

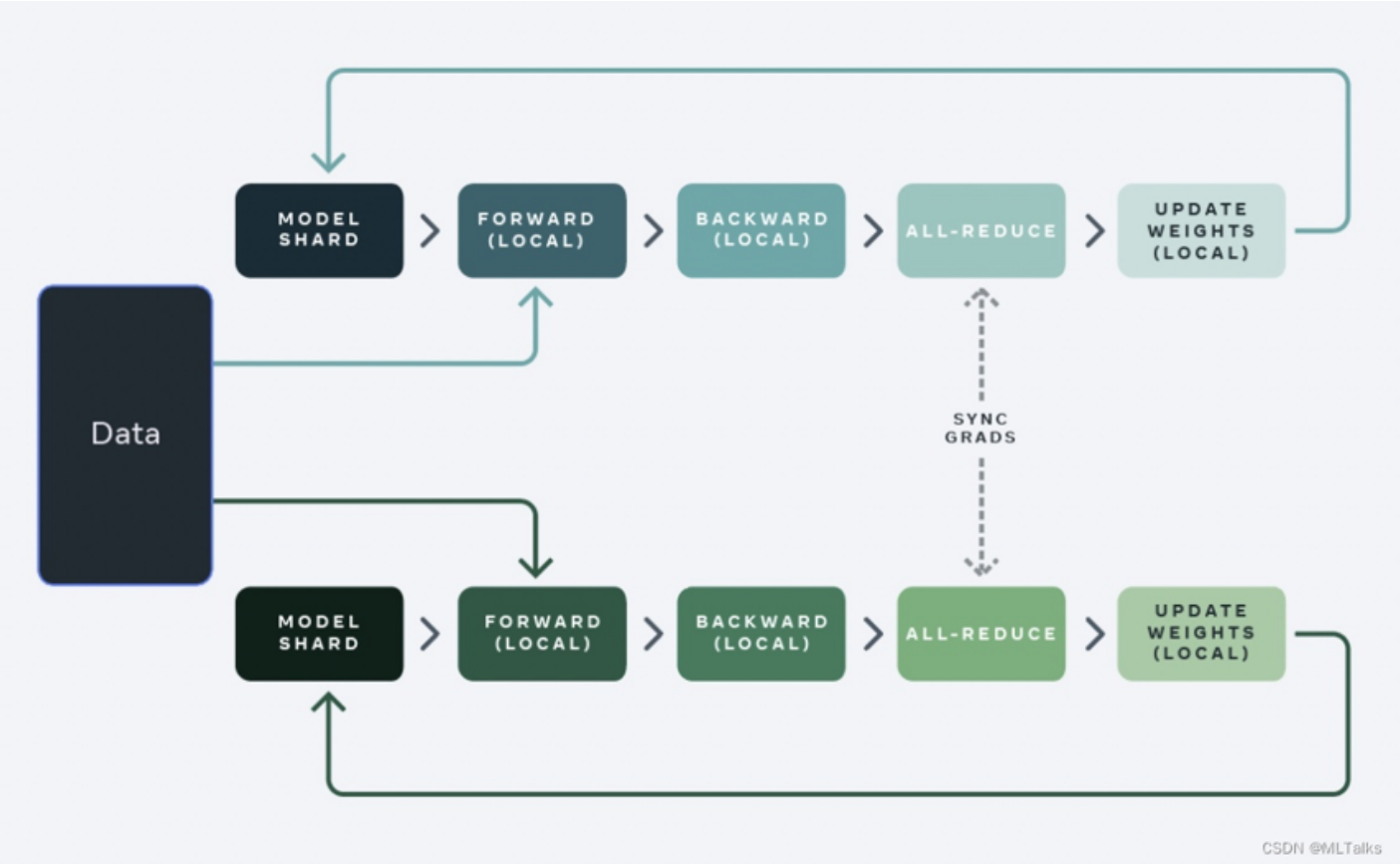
December, 5th

Background-Parallelism Paradigms

- Data Parallelism (DP)
- Tensor Parallelism (TP)
- Pipeline Parallelism (PP)
- Fully Sharded Data Parallelism (FSDP)
- Zero

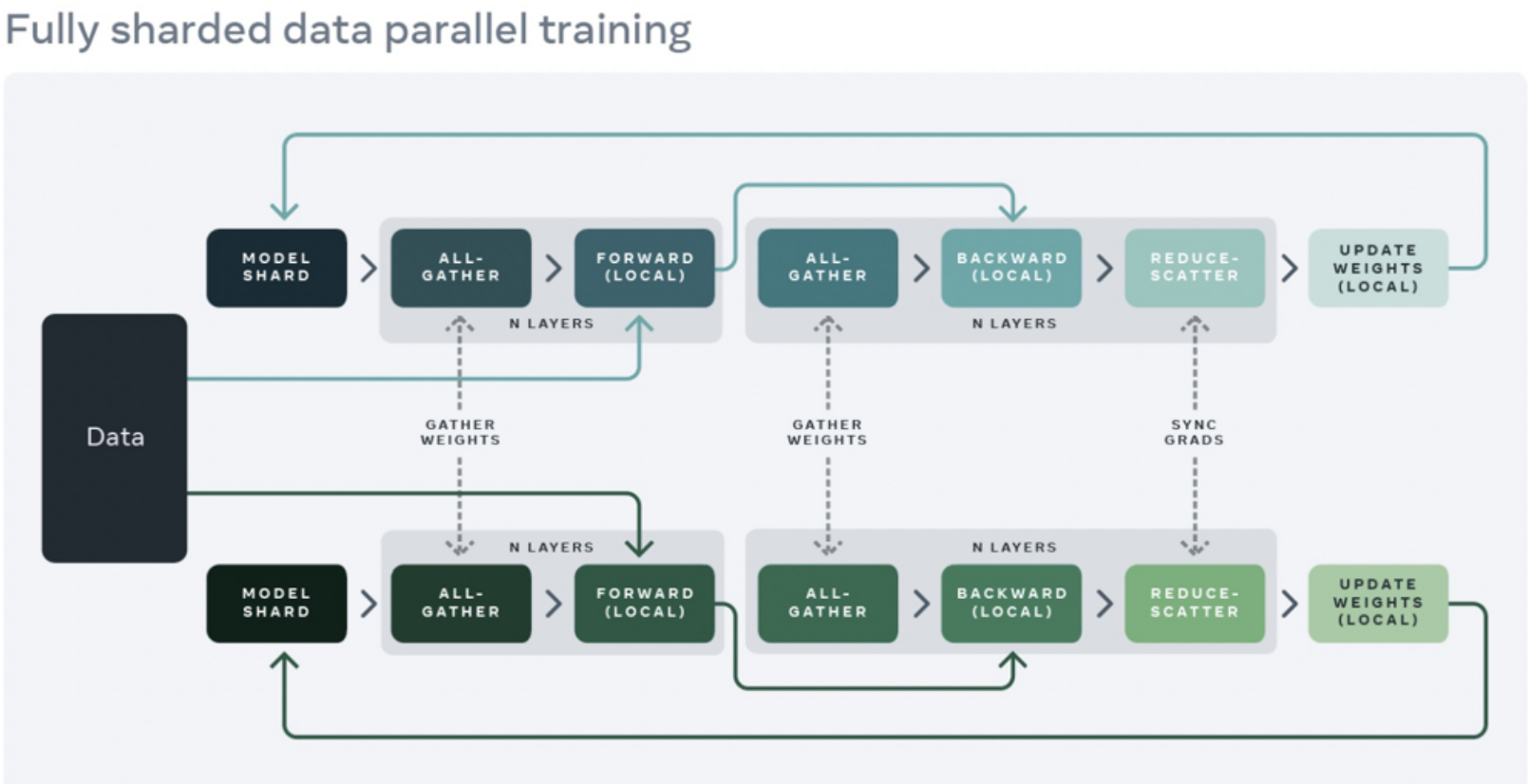
Background-Parallelism Paradigms-DDP

- Distributed Data Parallelism (DDP)
  - The entire model's parameters/gradients/optimizer states are stored on each GPU



Background-Parallelism Paradigms-FSDP

- Fully Sharded Data Parallelism (FSDP)
  - Each GPU only stores a portion of the model's parameters/gradients/optimizer states
  - High communication cost but overlap-friendly



Collective Communication

P devices for each volume D

- **Allreduce** aggregates data from different ranks by applying a reduction operation for a result of volume D.
- **Reduce-Scatter** reduces input values across ranks, with each rank receiving a subpart of the result corresponding to a volume of $\frac{D}{P}$. This is typically employed after intensive computation operations.
- **Allgather** gathers values from all ranks and distributes the result of volume PD to all ranks. This is typically followed by intensive computation operations.

Background-Different Strategies

- For scalable SPMD (Single Program, Multiple Data) computation patterns, having communication of minimal cost is good.
- But having unperceived communication is even better!

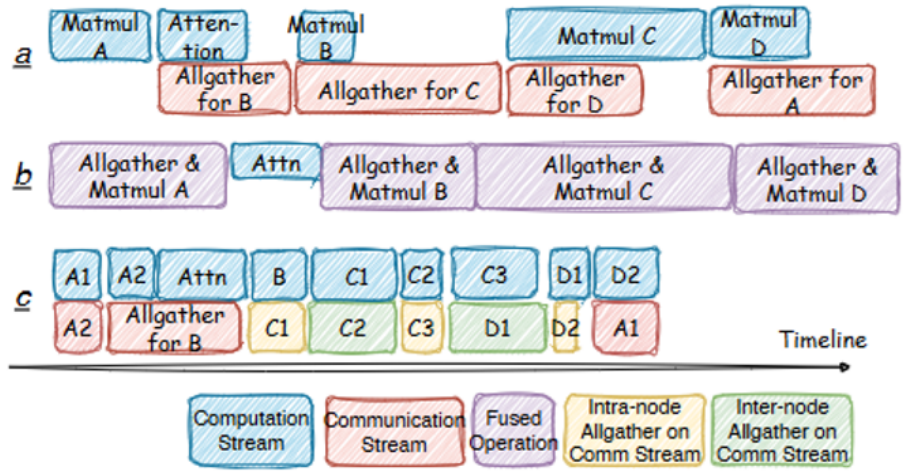


Figure 1. Different overlapping strategies on FSDP training of a simplified Transformer structure: *a* is direct scheduling that weights are gathered before MatMuls; *b* is MatMul and Allgather kernel fusion; *c* is Centauri scheduling that communication is partitioned from group and workload dimensions for better overlapping

Background-Different Strategies (Cont'd)

- Key insight:
  - Partitioning: Communication inherently is a mapping transformation (primitive) of workload across a group of devices

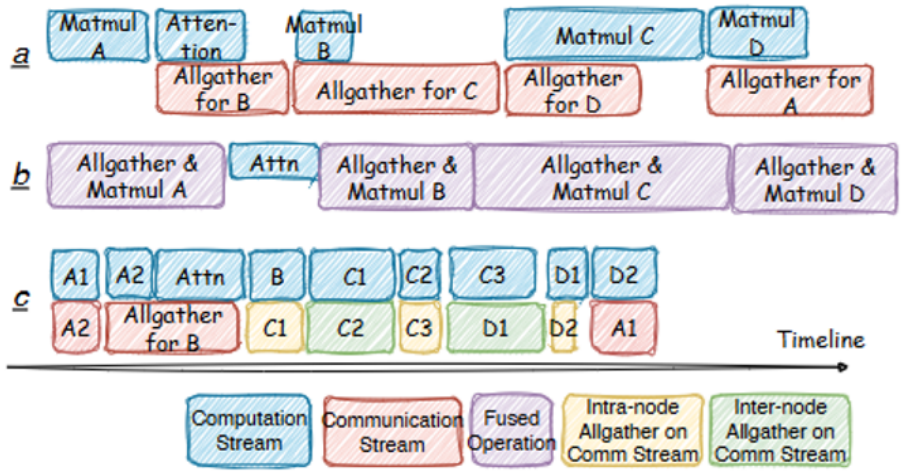


Figure 1. Different overlapping strategies on FSDP training of a simplified Transformer structure: *a* is direct scheduling that weights are gathered before MatMuls; *b* is MatMul and Allgather kernel fusion; *c* is Centauri scheduling that communication is partitioned from group and workload dimensions for better overlapping

Background-Different Strategies (Cont'd)

- Key insight:
  - Partitioning: Communication inherently is a mapping transformation (primitive) of workload across a group of devices

Table 1. Overlapping Capability of Popular Frameworks

Works	Primitive	Group	Workload	Scheduling
Better Together [20]	✓	-	-	✓
Breadth-First [17]	-	-	-	✓
CoCoNet [11]	✓	-	✓	-
Dist-Einsum [40]	-	-	✓	-
DeepSpeed Zeros [29, 30]	✓	-	✓	-
Megatron-LM [21, 33]	-	-	-	✓
OOO-Backprop [27]	-	-	-	✓
Torch DDP, FSDP [18, 42]	-	-	✓	✓

Background-Different Strategies (Cont'd)

- Key insight:
  - Scheduling: Training computation patterns can be simplified into three hierarchical tiers: 1) operation, 2) layer, and 3) model

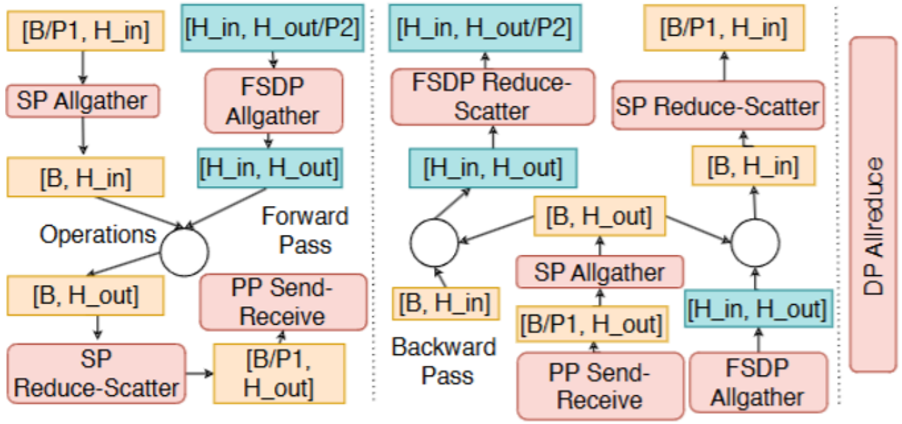


Figure 2. Simplified forward and backward workflows of hybrid parallelism, encompassing sequence parallelism, fully sharded data parallelism, pipeline parallelism, and data parallelism.

Overview

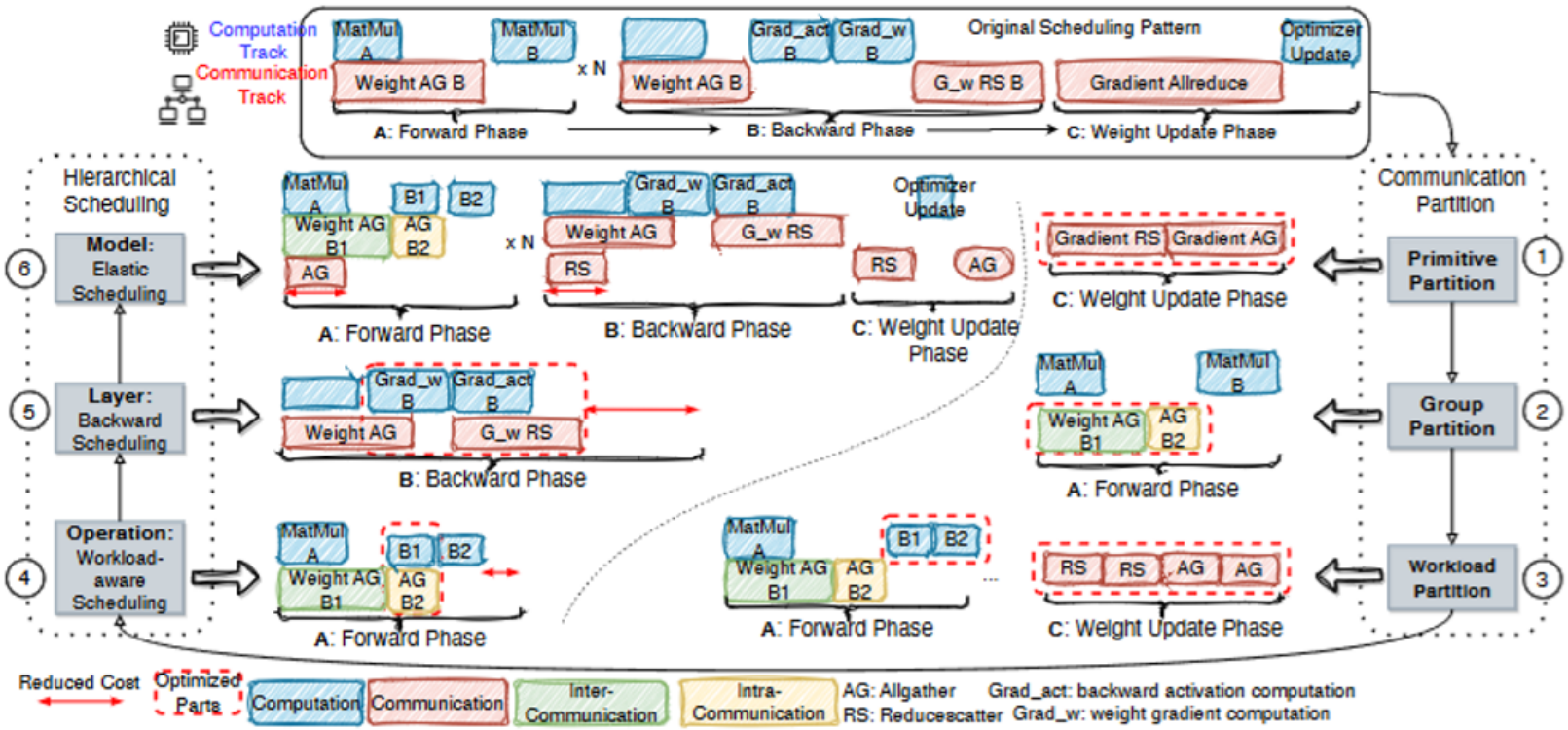


Figure 3. *Centauri* workflow overview for a hybrid parallel training example of DP and FSDP. ① Primitive substitution: Allreduce is split into reduce-scatter and allgather. ② Group partition: Allgather in the forward phase is split into inter-node group and intra-node group communication. ③ Workload partition: This step focuses on splitting collective and computation tasks with proper granularity. ④ Operation scheduling: overlap between two split collective and computation operations. ⑤ Layer scheduling: The execution is adjusted according to the critical path within a layer. ⑥ Model scheduling: Overlapping between different phases enhances overall training efficiency.

Centauri-Communication Partition-(1) Primitive Partition

Principles:

- Primitive substitution is the first partition dim.
- Ensuring performance scalability of collective partition
- Rooted collectives may result in contention in root ranks
- Following the substitution of allreduce with reduce-scatter and allgather, TP and DP evolves into SP and Zero-1,2.

Table 1: Primitive Substitution List

Primitive	Sub-primitives	Scalability
AllReduce	Reduce + Broadcast	x $(2 * N\log P)$
	Reduce-scatter + AllGather	✓ $(2N)$
Reduce-Scatter	Reduce + Scatter	x $(N\log P + N)$
	Reduces of distinct roots	✓ $(N\log P)$
AllGather	Gather + Broadcast	x $(N\log P + N)$
	Broadcasts of distinct roots	✓ $(N\log P)$

Centauri-Communication Partition-(2) Group Partition

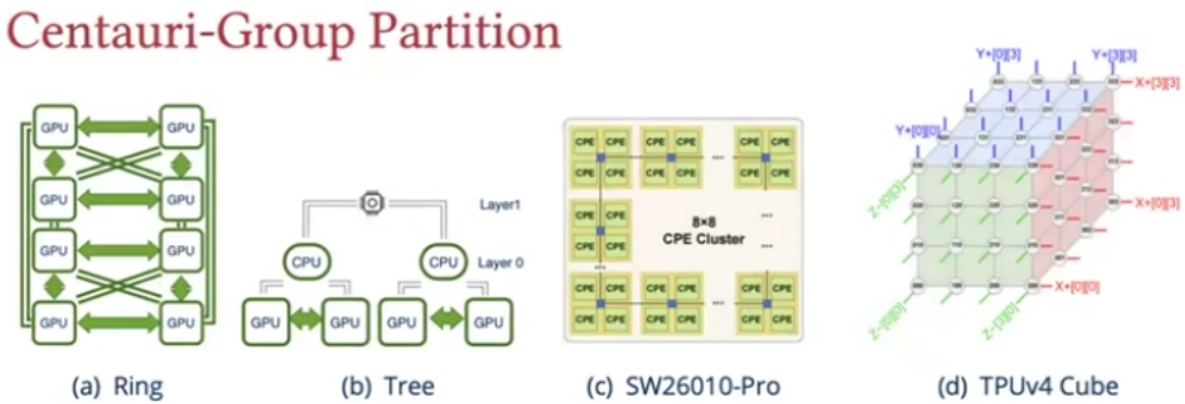


Figure 6: Typical topologies on modern clusters.

Principles:

- No heterogeneous bandwidths within a group.
- Do not break up to hardware-algorithm co-design abstraction.
- Topology awareness is important.

Centauri-Communication Partition-(3) Workload Partition

Table 3. Computation Dimension Types

Operation	Workload	Dims Types	Partition Dims
MatMul	$[A, B]x[B, C]$	OD, CD, OD	A, B, C
+, −, *, /, Dropout, ReLU, GeLU	$[A, B]$	OD, OD	A, B
SwiGLU, Softmax, LayerNorm	$[A, B]$	OD, ND	A

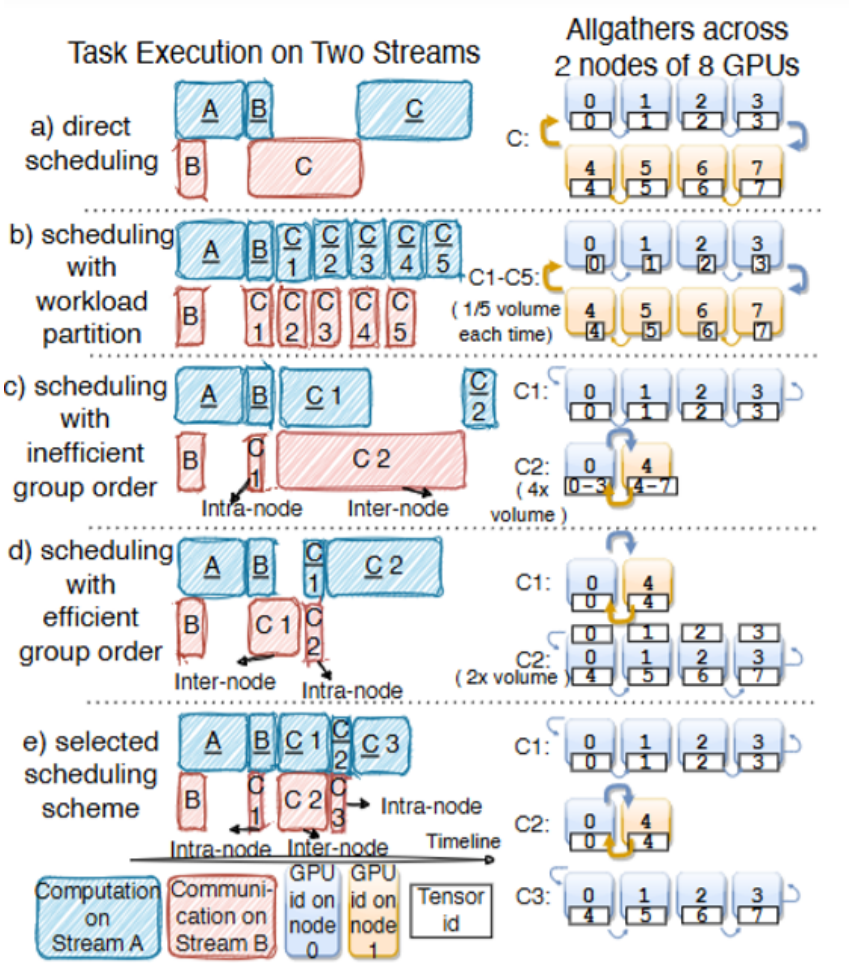
Principles:

- Longer chain of operations for better overlapping.
- Compatible partition dimensions within the chain.

- Contraction dimension (CD): Contraction dimensions of operations like MatMuls and Einsums.
- None-split dimension (ND): Coupled computation along this dimension and is not preferable for splitting. It includes reduced dimensions of normalization functions.
- Other dimension (OD): This type encompasses the remaining workload dimensions, such as the batch dimension and none-contraction dimensions of MatMuls.

Hierarchical Scheduling-(1) Operation Level

- a) scheduling with coarse granularity results in a large idle gap (x)
- b) fine granularity scheduling introduces large overhead due to excessive workload partition (x)
- c) scheduling with a group order of large inter-node communication overheads (x)
- d) group partition with a group order of small inter-node and acceptable intra-node communication overheads (x)
- e) the selected partition and scheduling scheme with no idle gaps (group partition and workload partition) (✓)



Hierarchical Scheduling-(2) Layer Level

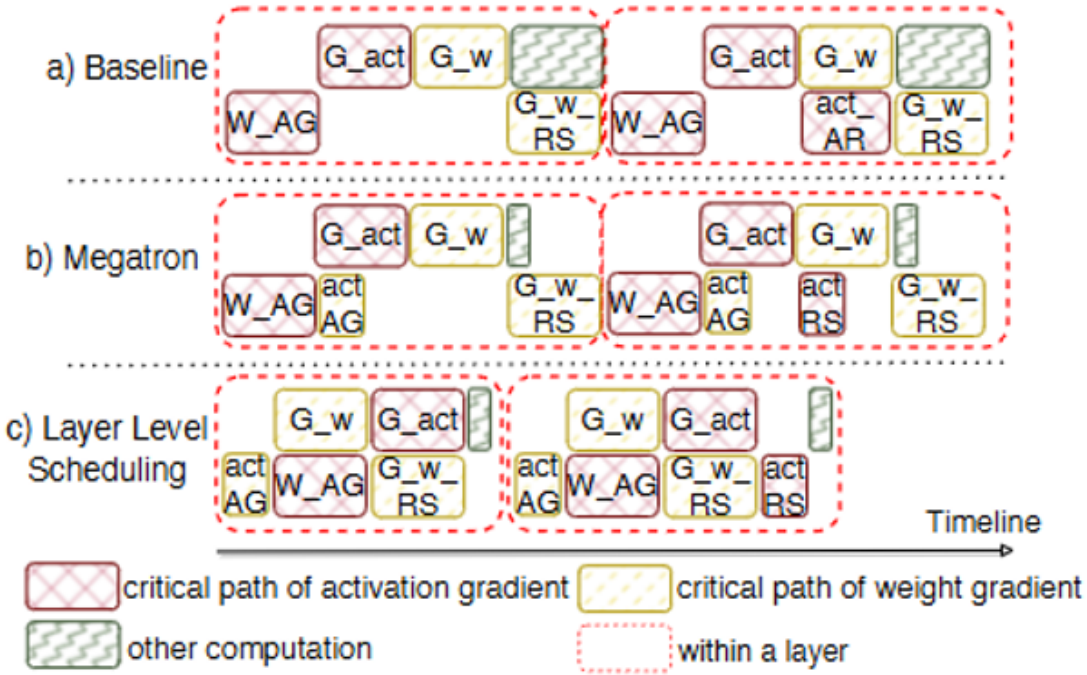


Figure 6. Layer-level Backward Scheduling of simplified TP and FSDP methods. a) OOO propagation [27] that activation computation is scheduled with higher priority. b) Megatron-LM sequence parallel method [15] that aims at overlapping activation communication. c) *Centauri* schedules critical path to maximize overlapping.

Hierarchical Scheduling-(3) Model Level

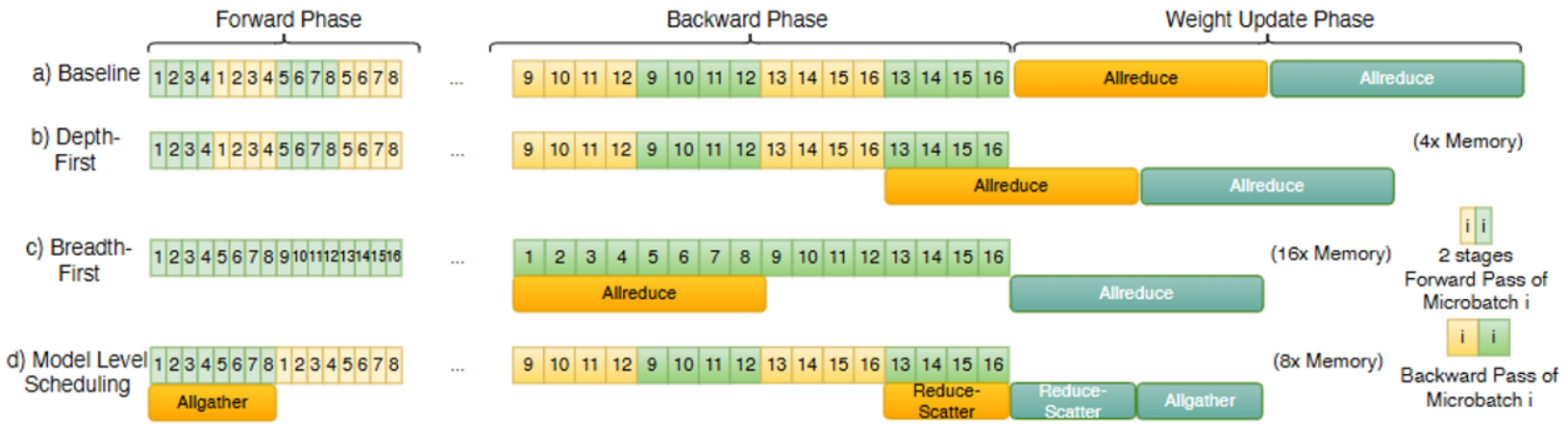


Figure 7. Model level scheduling of DP and PP: each case shows micro-batch scheduling and memory consumption of the first stage of PP group within each iteration. a) A sequential execution of forward, backward, and weight update phases, with an interleaved pipeline of 2 stages for each device, 16 micro-batches per batch, and a depth of 4. b) Direct overlap allreduce of the second stage with backward of the first stage, with minimal memory cost of 4x activation. c) All micro-batches (16) launched together for maximal overlapping, with a maximal memory cost of 16x activation. d) Minimal number (8) of micro-batches launched together for well overlapping, with medium memory cost of 8x activation.

Evaluation-FSDP, DP

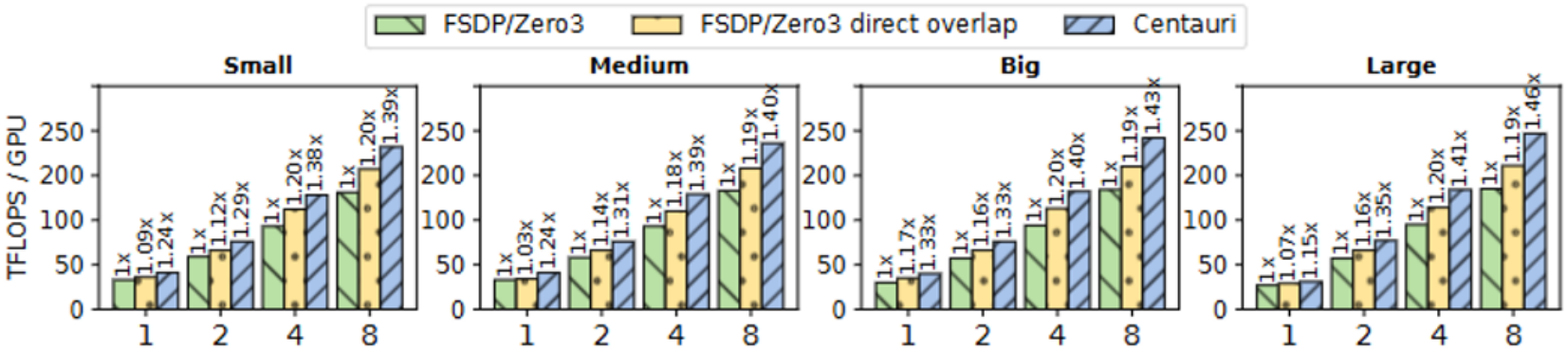


Figure 8. Performance of FSDP/Zero3 tasks on 2 nodes of Cluster A, with FSDP group size 16.

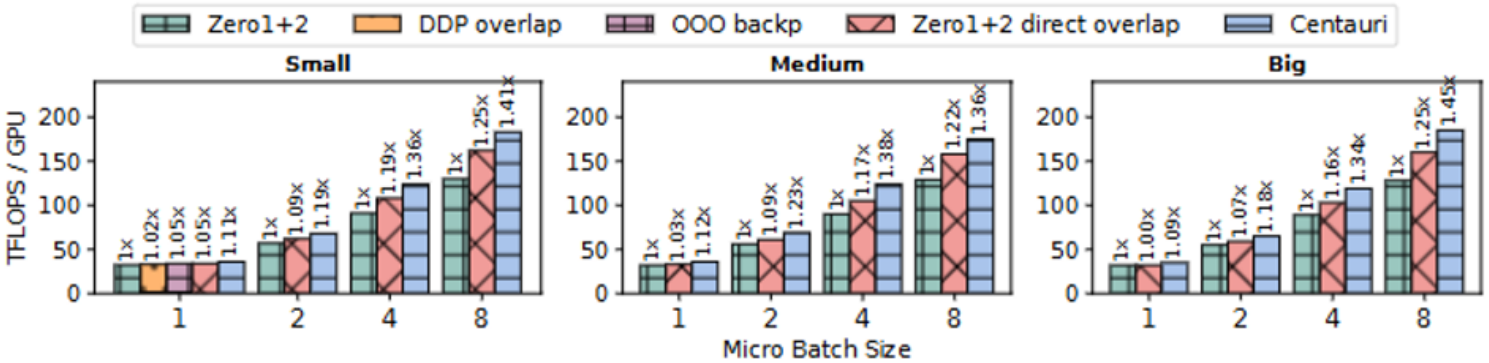


Figure 9. Performance of DP tasks on 2 nodes of Cluster A, with DP group size of 16.

Evaluation-Hybrid Parallel

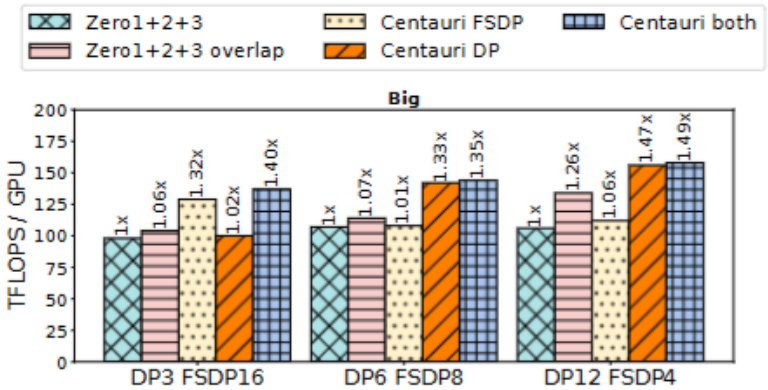


Figure 10. Performance of FSDP+DP tasks on 6 nodes of Cluster A, with a total group size of 48.

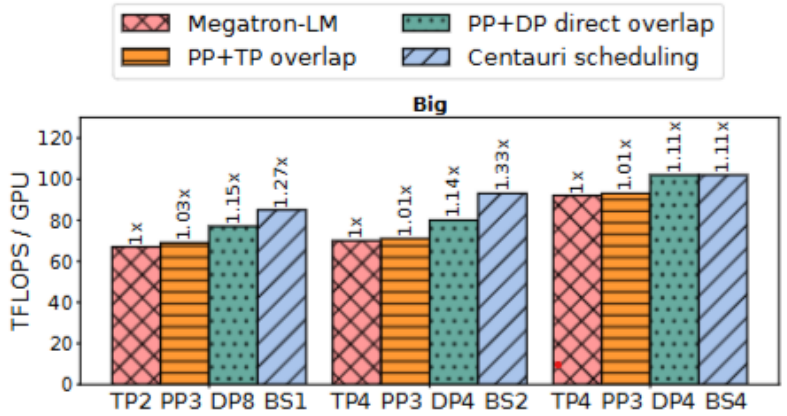


Figure 11. Performance of TP+PP+DP tasks on 6 nodes of Cluster A, with a total group size of 48 and total gradient accumulation steps of 6.

Evaluation-Scalability

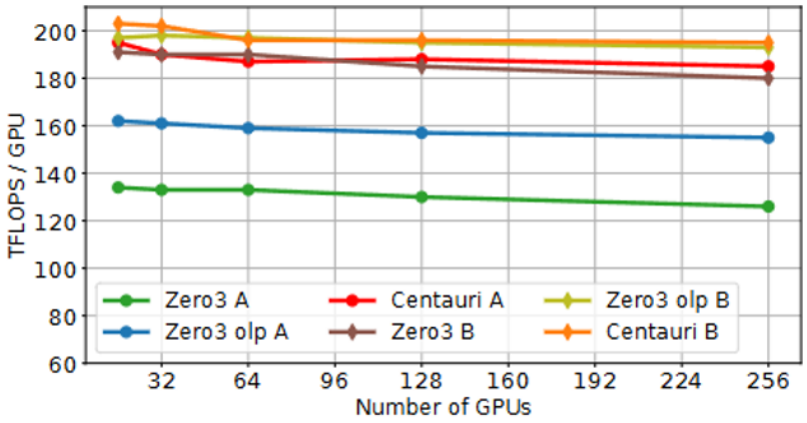


Figure 13. Scalability of FSDP/Zero3 on Cluster A&B

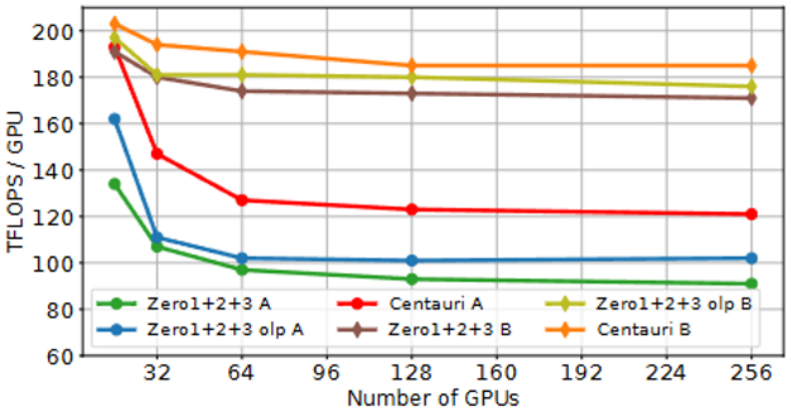


Figure 14. Scalability of FSDP16+DP on Cluster A&B

Future Work

Partitioning:

- Collective primitive analysis
- Group partitioning on modern architecture
- Auto-parallel workload partitioning

Scheduling:

- Whole graph level cost model scheduling.
- Multiple computation and communication streams scheduling
- Other SPMD scientific computation or MoE tasks.

Thanks!