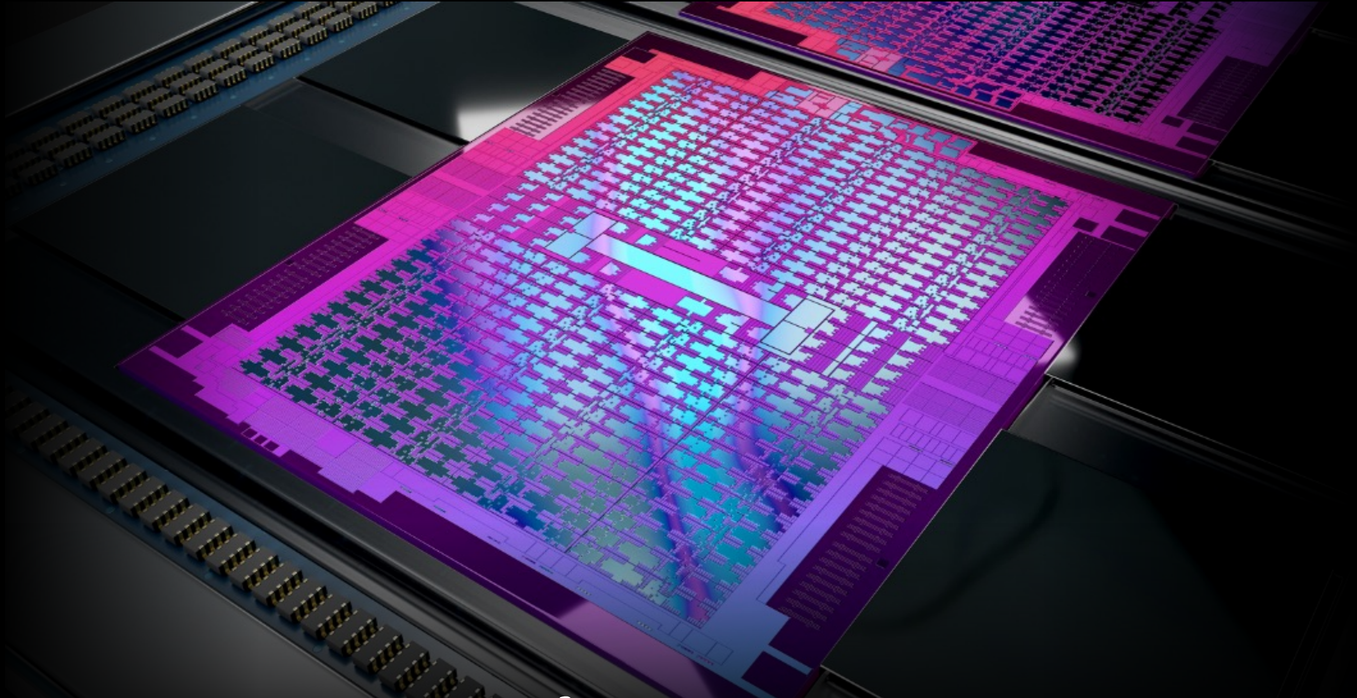
TRAINING MASSIVE-SCALE AI FOUNDATIONAL MODELS ON FRONTIER

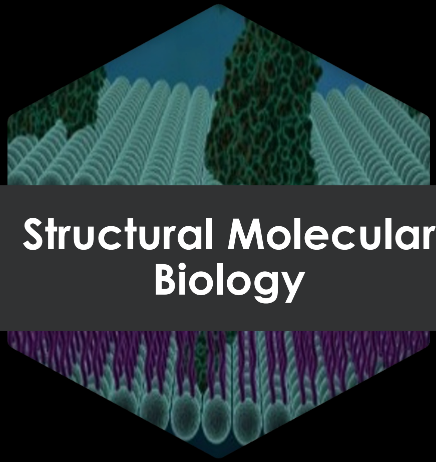
Sumanth Gudaparthi
AMD Research and Advanced Development



THE WORLD CAN'T ACCELERATE ENOUGH

Inelastic Neutron Spectroscopy (INS)

3D Additively Manufactured Microstructures



Structural Molecular Biology

High Flux Isotope Reactor

AMD
EPYC | INSTINCT

LEADING THE EXASCALE ERA

- Powering World's #1 Supercomputer
  First to break Exascale barrier

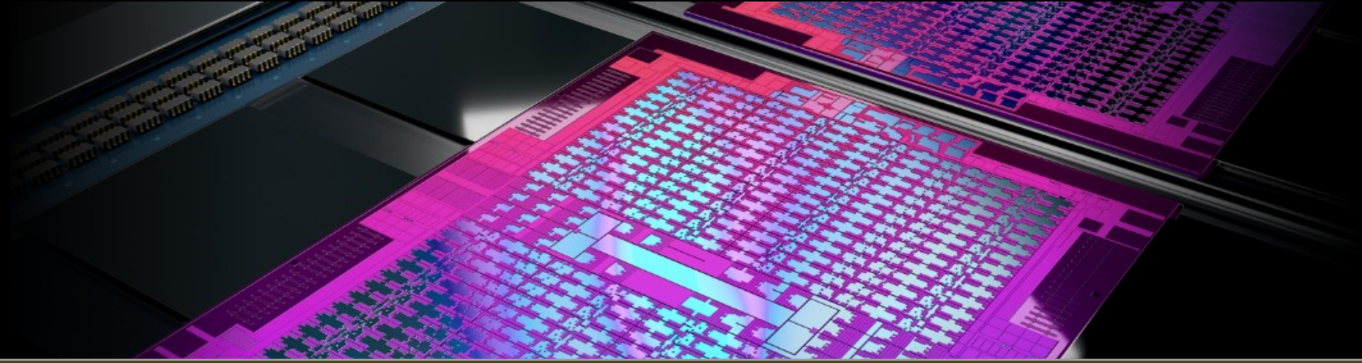
- 9,408 Compute Nodes each with
  One 64-core AMD "Trento" CPU, four AMD Radeon Instinct MI250X GPUs, and 512GB of DDR4 memory

- 37,888 Instinct MI250X GPUs
  8,138,240 cores and 4.6 petabytes of HBM memory (128GB/GPU)

- 700 Petabytes of Storage
  9.95 Eflops on HPL-MxP Mixed-Precision Benchmark

Source: TOP500, Green500 and HPL-MxP lists, as of June 2023

AMD
together we advance_

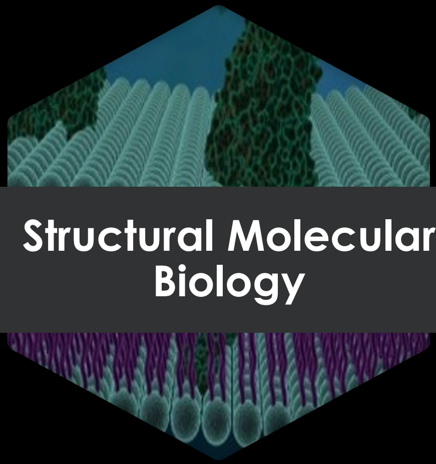


Foundational Models

- Large Language Models
- Vision Transformer
- Graph Neural Networks

Inelastic Neutron Spectroscopy (INS)

3D Additively Manufactured Microstructures



Structural Molecular Biology

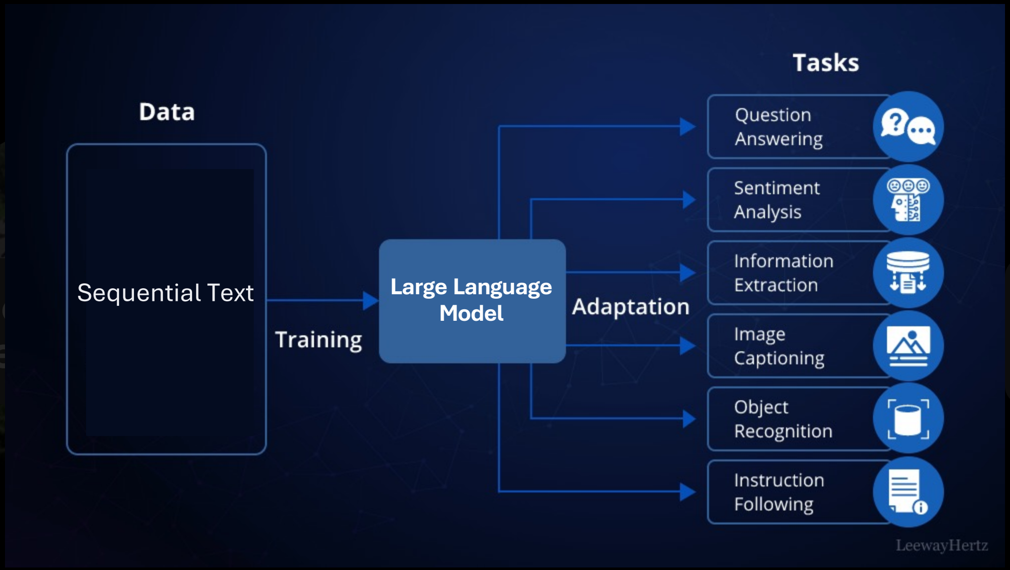
High Flux Isotope Reactor

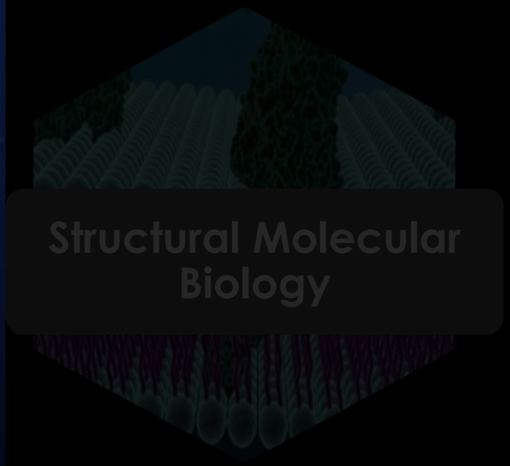
AMD
together we advance_

THE WORLD CAN'T ACCELERATE ENOUGH

Foundational Models

- Large Language Models
- Vision Transformer
- Graph Neural Networks





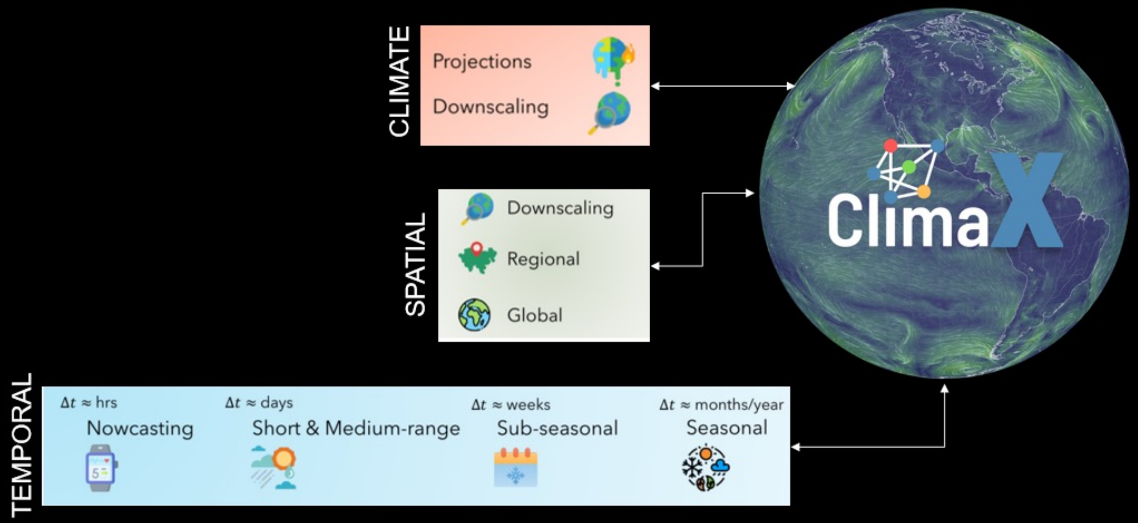
AMD together we advance_

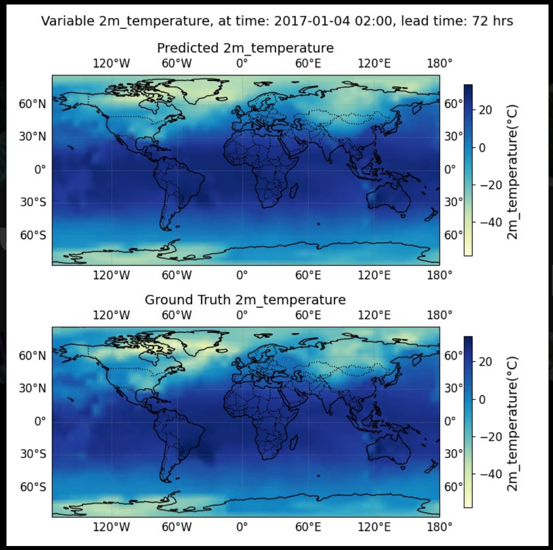
THE WORLD CAN'T ACCELERATE ENOUGH

Foundational Models

- Large Language Models
- Vision Transformer
- Graph Neural Networks

Weather and Climate Predictions





High Flux Isotope Reactor

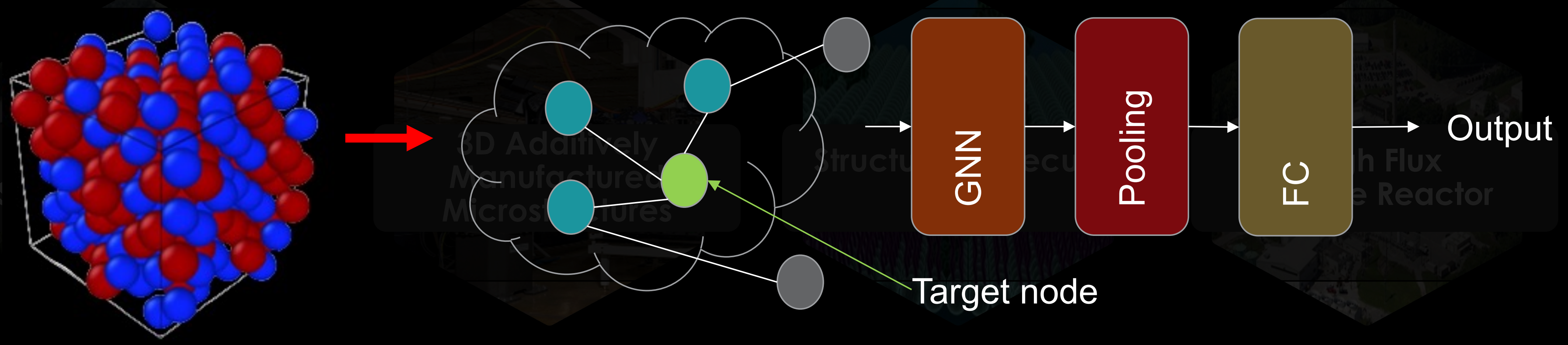
AMD
together we advance_

THE WORLD CAN'T ACCELERATE ENOUGH

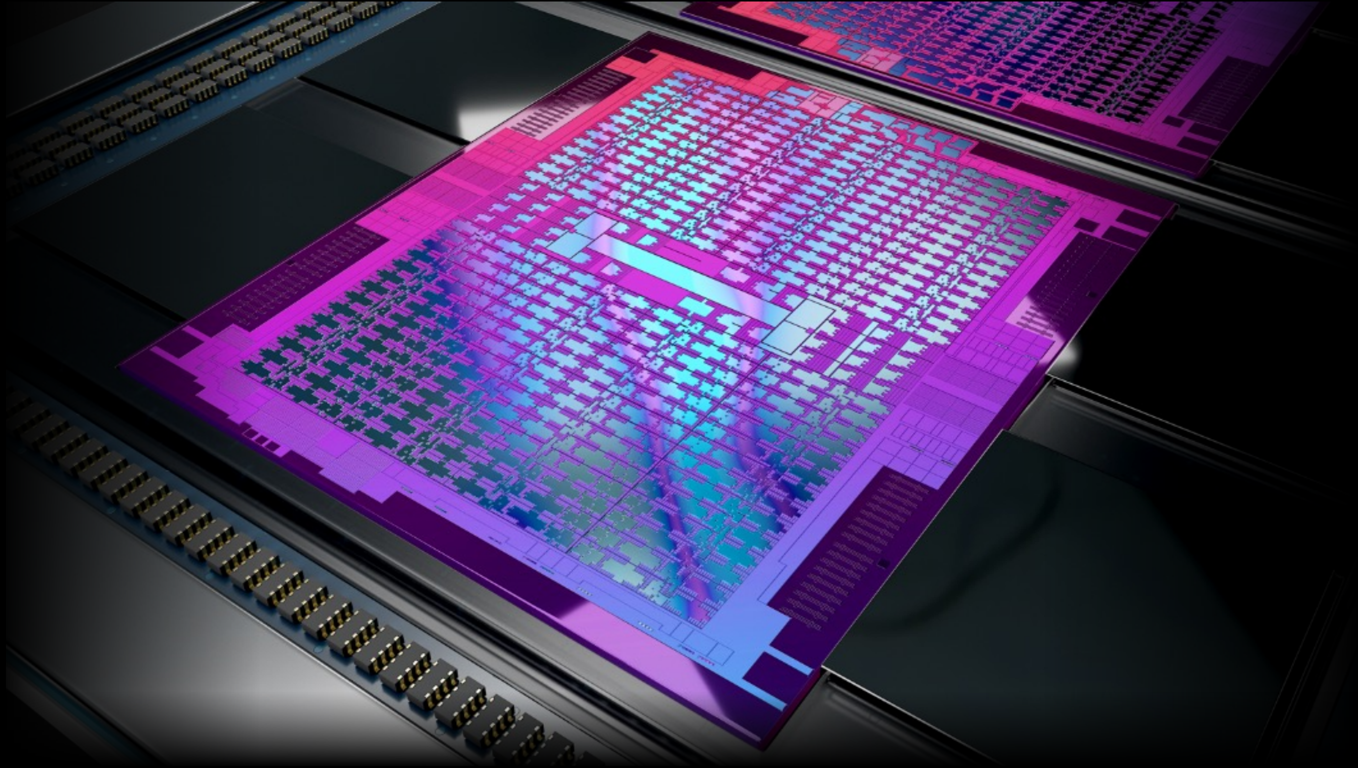
Foundational Models

- Large Language Models
- Vision Transformer
- Graph Neural Networks

<u>Discovery of new materials and new chemical processes</u>



together we advance_



DISTRIBUTED TRAINING OF LLMS ON FRONTIER

WHAT DO YOU NEED TO TRAIN A TRILLION PARAMETER MODEL?

- Training Data Required* = (20 – 200) × # parameters
  = (20 – 200) × 1 Trillion
  = (20 – 200) Terabytes
- Total Compute Required = 6 × #parameters × # training_datapoints
  = 6 × (1 Trillion) × (20 – 200) Trillion
  = (120 – 1200) Million ExaFLOPS

Modern GPUs have < 200 GB Memory and operate at a few hundred of TFLOPS

Solution:

Distribute the training workload across thousands of GPUs on an ExaFLOP machine

*Source: Junqi Yin et. al, "FORGE: Pre-Training Open Foundation Models for Science". SC '23).

FRONTIER FOR THE RESCUE!

Frontier Specs

- 37,888 AMD Instinct™ MI250X GPUs
- Each MI250X has
  - 128 GB memory
  - 383 TFLOPS peak throughput (FP16)
- Each MI250X is split across 2 GCDs (Graphic Compute Die)

Challenges:

- How to distribute the training workload across thousands of MI250X GPUs?
- How to take advantage of existing software developed for accelerating LLMs on these massive clusters?

3D PARALLELISM USING MEGATRON-DEEPSPEED

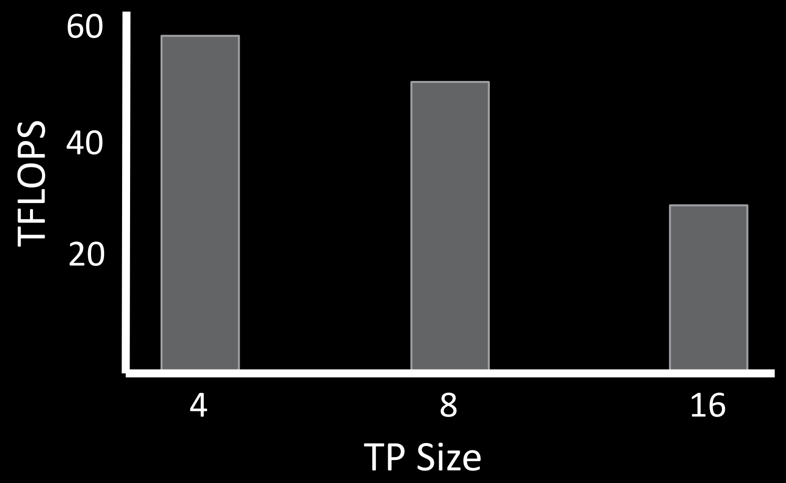
- A combination of Tensor, Pipeline, and Data parallelism ported to Frontier
- Determine how many GPUs (world-size) you need to fit the model
- Factorize world-size into TP (Tensor parallel size) and PP (Pipeline parallel size)

Distribution Strategy	Tunable Parameters
Tensor Parallelism	Tensor Parallel Size (TP)
Pipeline Parallelism	Pipeline Parallel Size (PP), #Microbatches (m)
Sharded Data Parallelism	ZeRO-1
Common	Micro Batch Size
Mixed Precision Training	FP16, BF16

Table: Distribution Strategies and Relevant Tunable Parameters

BEST PRACTICES WITH PARALLELISM PARADIGMS

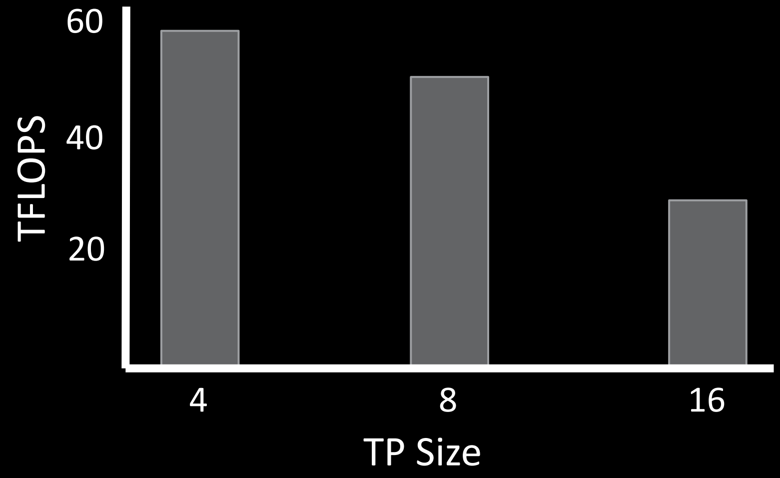
- Tensor Parallelism (TP)
  - Keep it within the node (TP < 8)
- Pipeline Parallelism (PP)
  - Use large number of micro-batches (but that can increase the global batch-size)
- Data Parallelism (DP)
  - Can’t use too much data parallelism. A large global batch size will make the model divergent

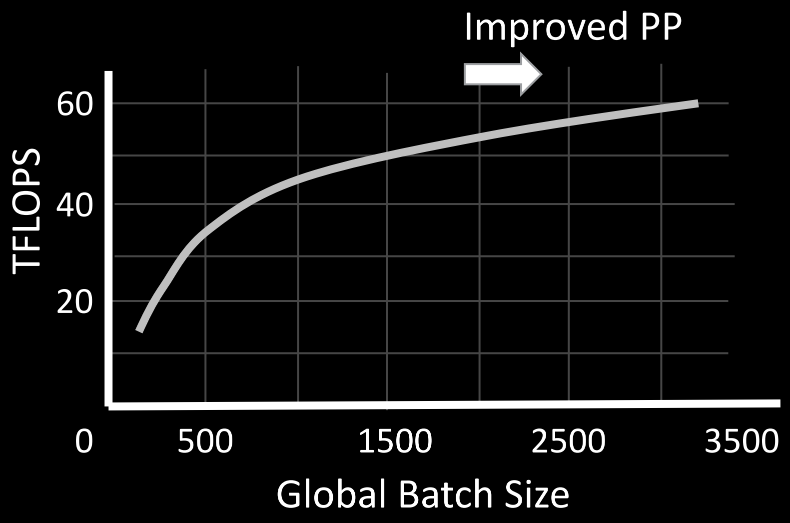


AMD
together we advance_

BEST PRACTICES WITH PARALLELISM PARADIGMS

- Tensor Parallelism (TP)
  - Keep it within the node (TP < 8)
- Pipeline Parallelism (PP)
  - Use large number of micro-batches (But that can increase the global batch-size)
- Data Parallelism (DP)
  - Can't use too much data parallelism. A large global batch size will make the model divergent.

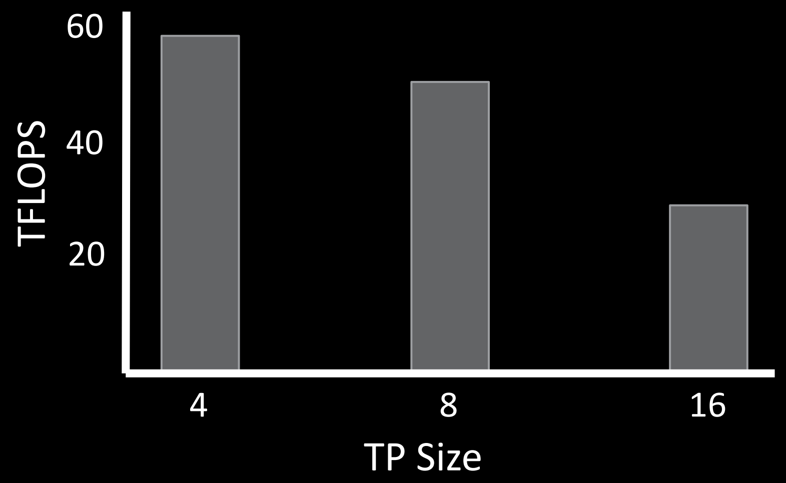


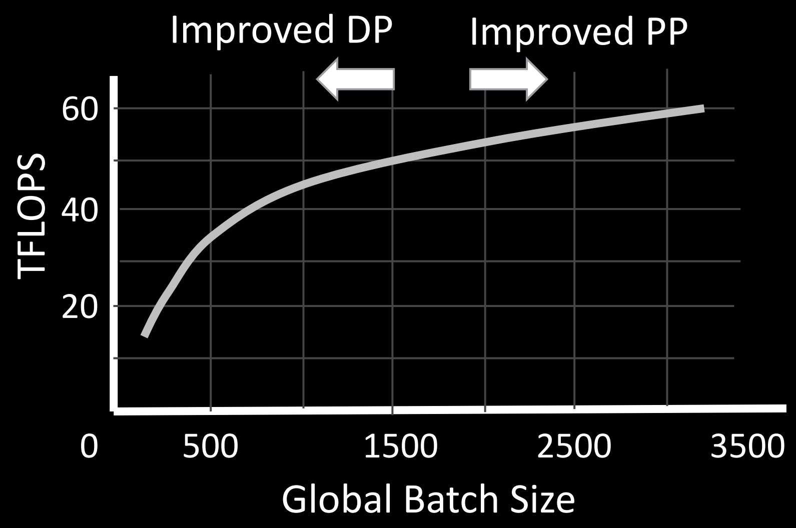


AMD
together we advance_

BEST PRACTICES WITH PARALLELISM PARADIGMS

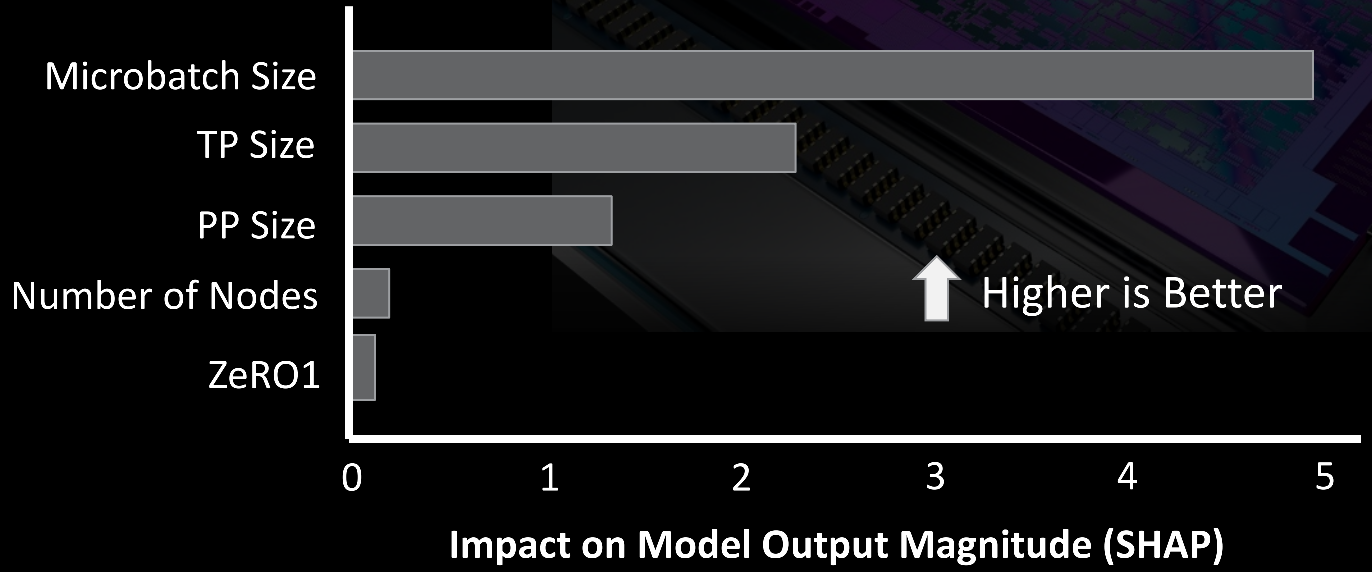
- Tensor Parallelism (TP)
  - Keep it within the node (TP < 8)
- Pipeline Parallelism (PP)
  - Use large number of micro-batches (But that can increase the global batch-size)
- Data Parallelism (DP)
  - Can’t use too much data parallelism. A large global batch size will make the model divergent.





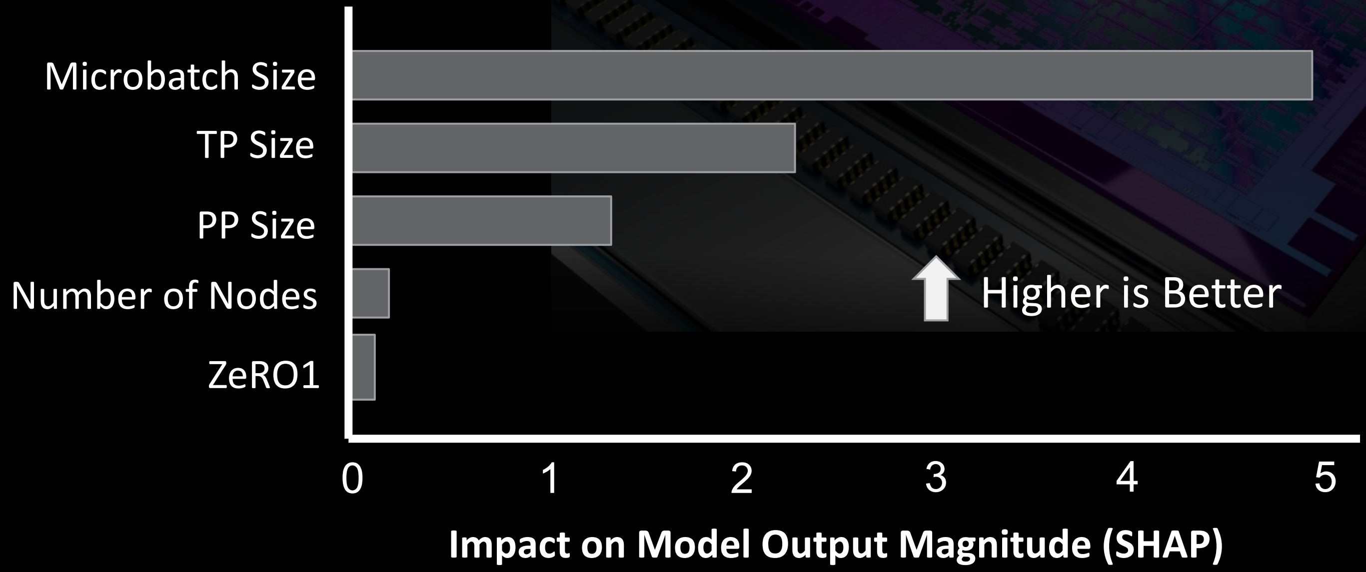
AMD
together we advance_

SEARCHING A 3D PARALLELISM STRATEGY USING DEEPHYPER



- DeepHyper: A Bayesian search algorithm for hyperparameter search.
- SHAP (SHapley Additive exPlanations) sensitivity analysis to assess the impact of hyperparameters on performance.
- Microbatch size is the most important parameter to tune followed by TP.

SEARCHING A 3D PARALLELISM STRATEGY USING DEEPHYPER



100% Weak Scaling*

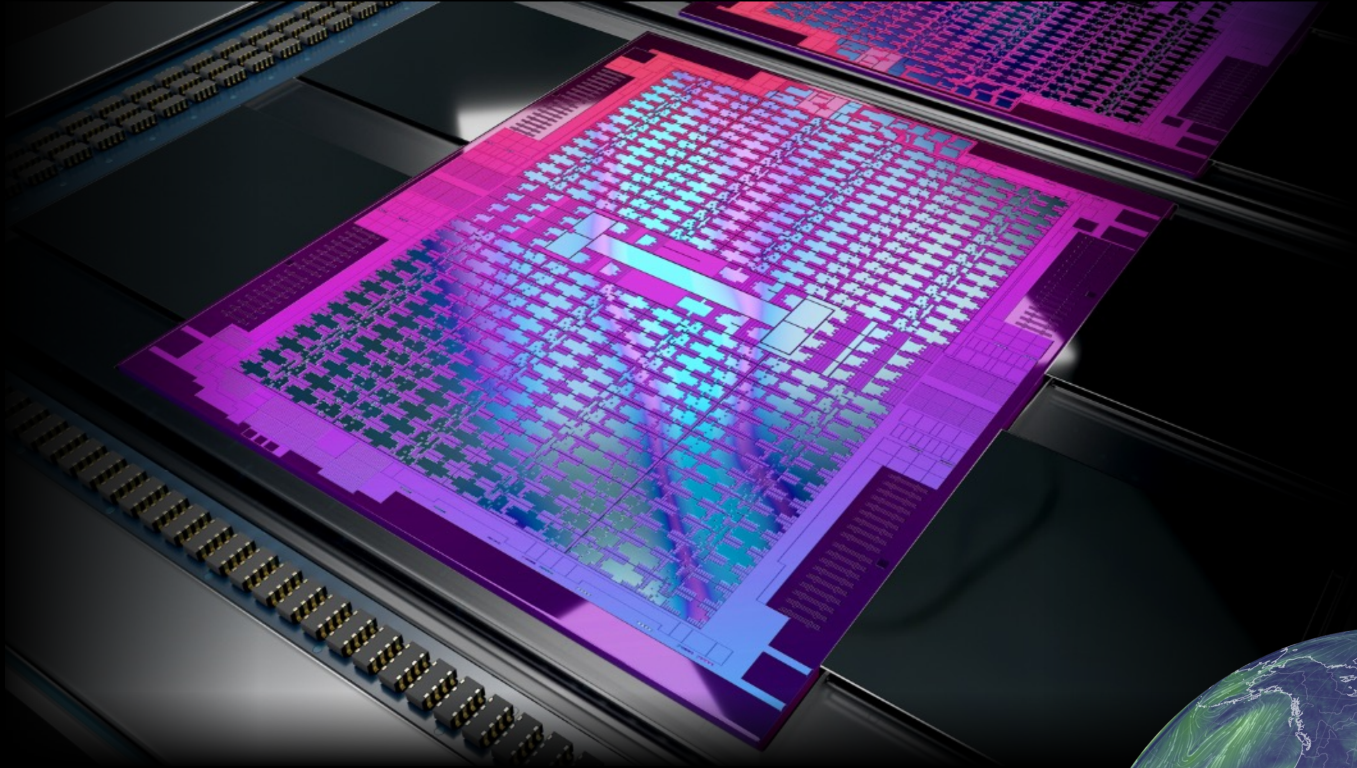
87% Strong Scaling*

- DeepHyper: A Bayesian search algorithm for hyperparameter search.
- SHAP (SHapley Additive exPlanations) sensitivity analysis to assess the impact of hyperparameters on performance.
- Microbatch size is the most important parameter to tune followed by TP.

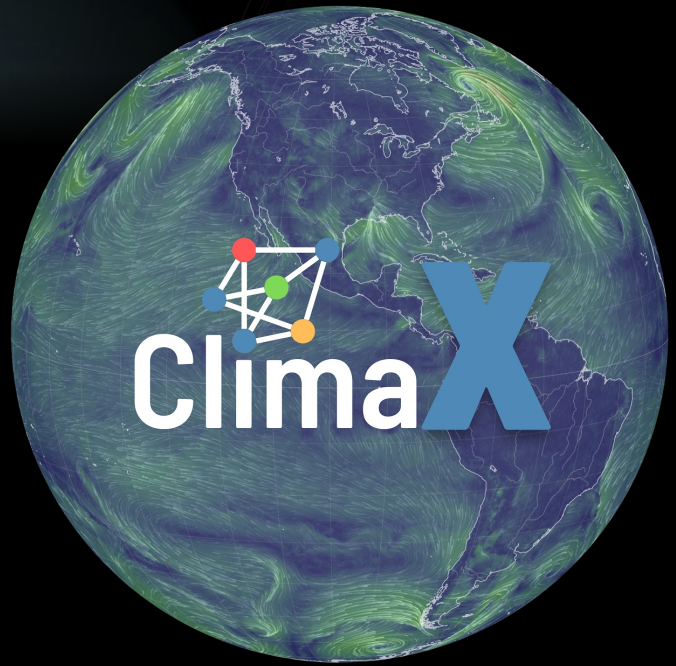
*On 3,072 MI250X GCDs

SCALING LARGE LANGUAGE MODELS SUMMARIZED

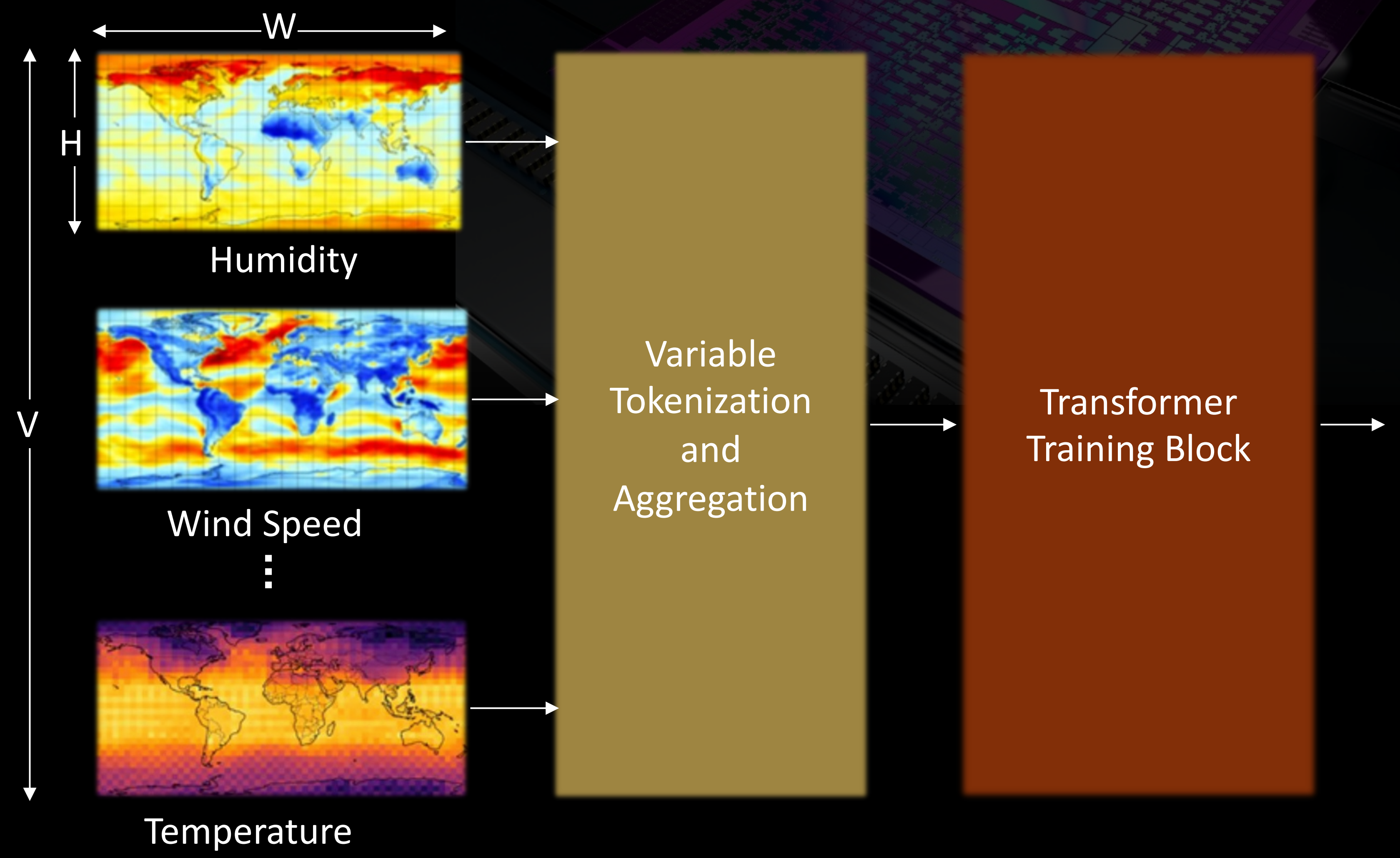
1. Training Challenges
   - Training large language models (LLMs) with billions to trillions of parameters involves overcoming GPU memory and communication challenges.
2. Parallelism Strategies
   - Model parallelism (tensor and pipeline) and data parallelism distribute the load across multiple GPUs to address memory constraints.
3. Software and Frameworks
   - The right combination of parallelization and frameworks like Megatron-DeepSpeed, plus hyperparameter tuning, are important to high throughput on Frontier with AMD ROCm software.
4. Performance Achievements
   - Achieved high GPU throughput and strong scaling efficiencies (up to 100% weak scaling, 89% and 87% strong scaling) for 175 billion and 1 trillion parameter models on thousands of GPUs.



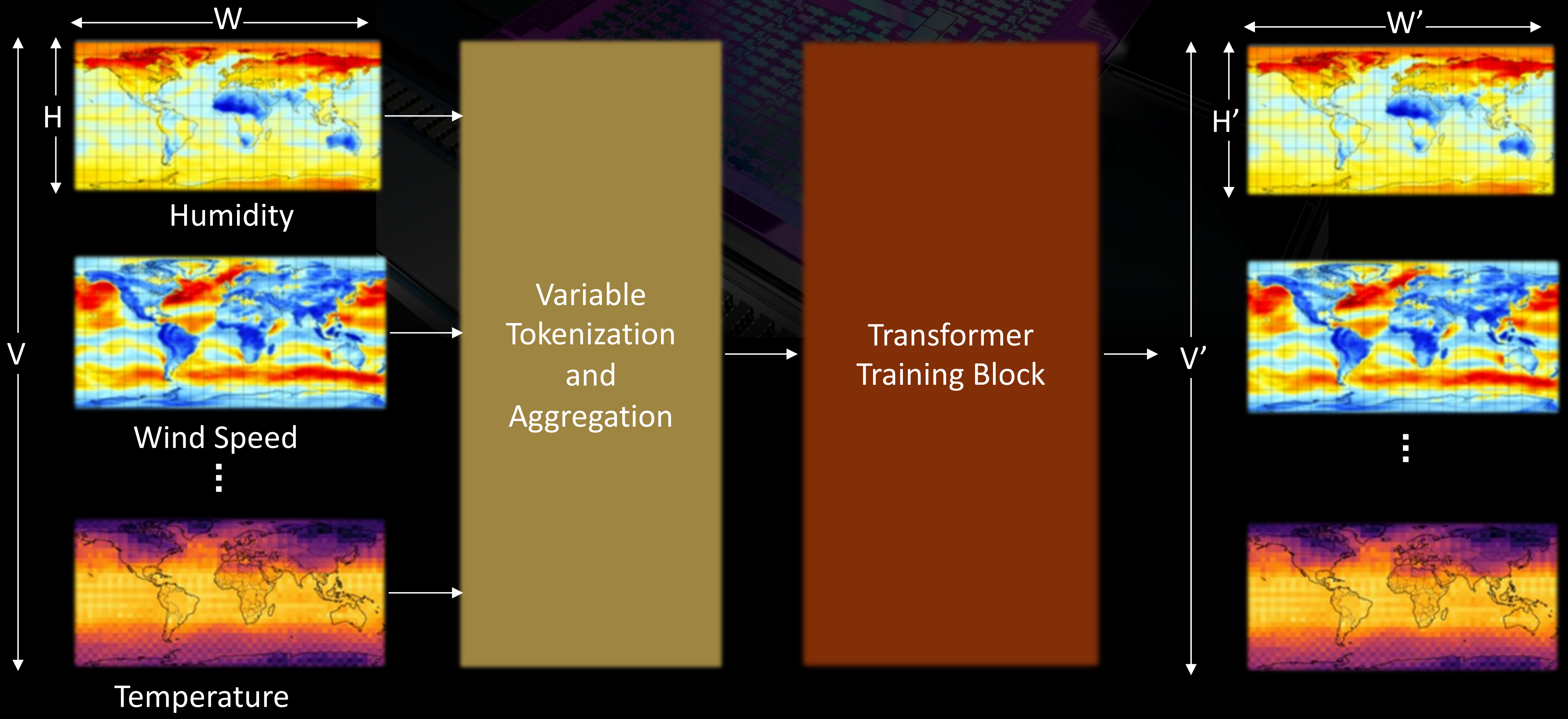
CLIMAX: AI FOUNDATION MODEL FOR BETTER WEATHER AND CLIMATE SOLUTIONS



CLIMAX: VISION TRANSFORMER BACKBONE



CLIMAX: VISION TRANSFORMER BACKBONE



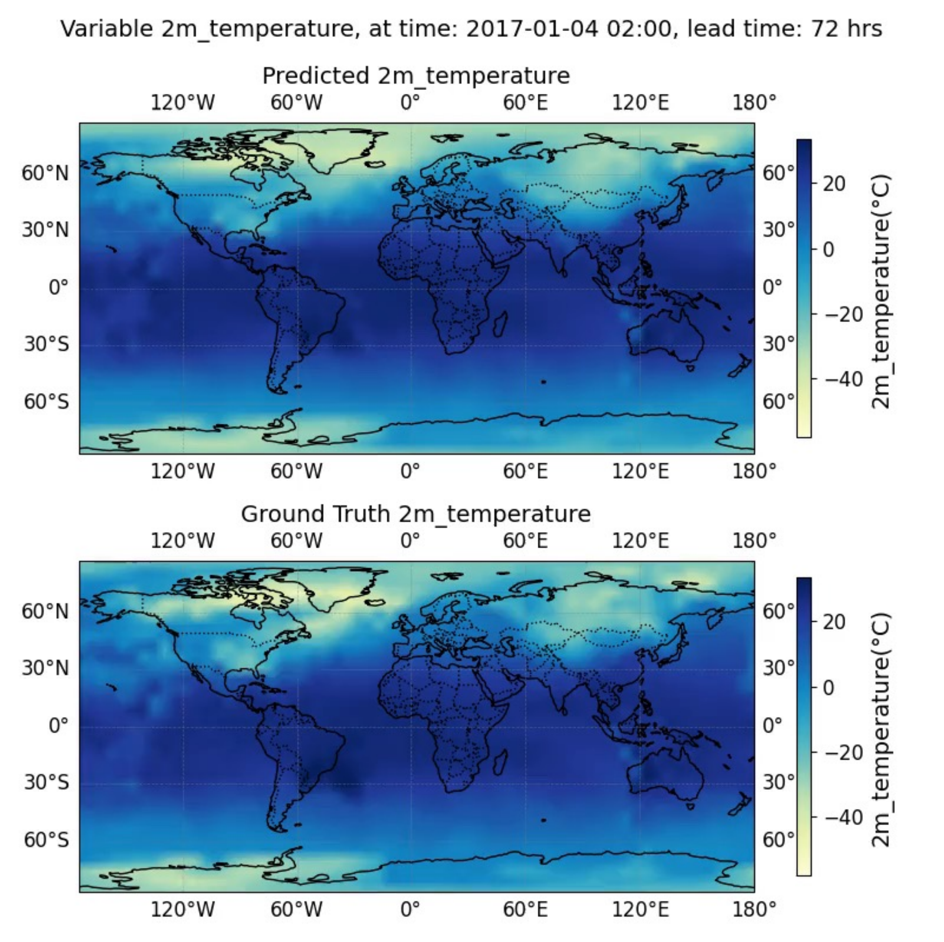
CLIMAX CAN ACCURATELY FORECAST WEATHER 72 HOURS AHEAD

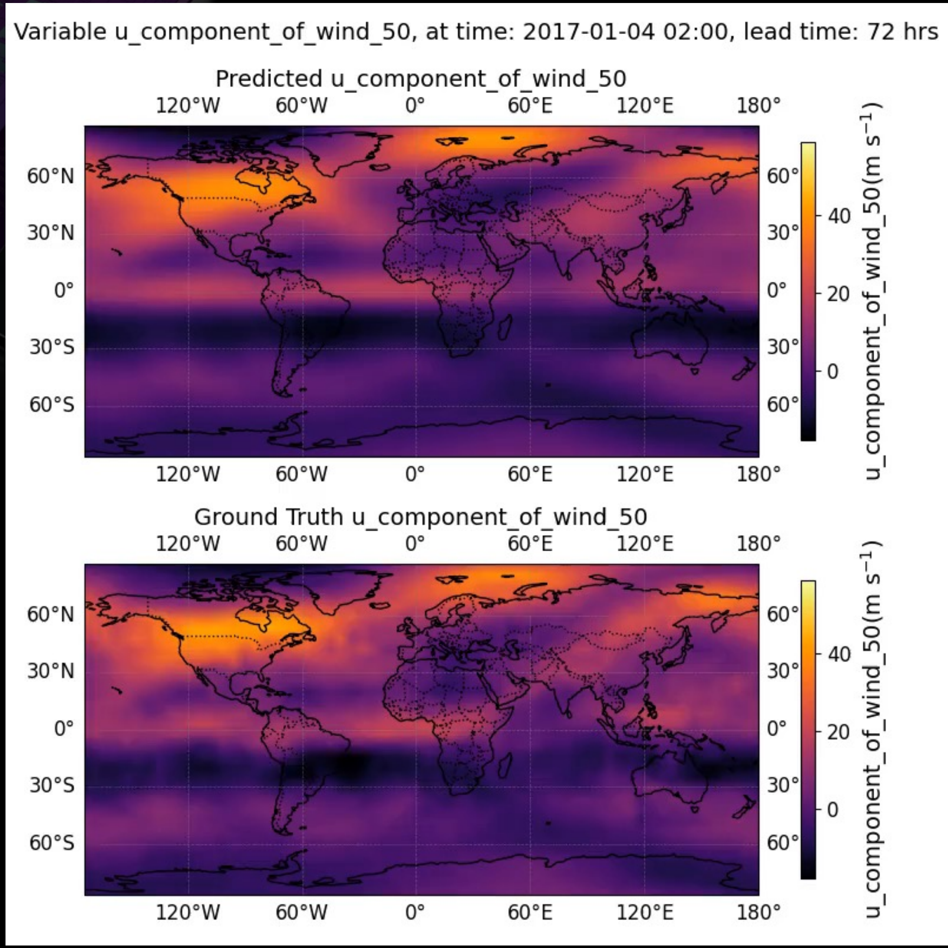
Temperature

Wind

Predicted

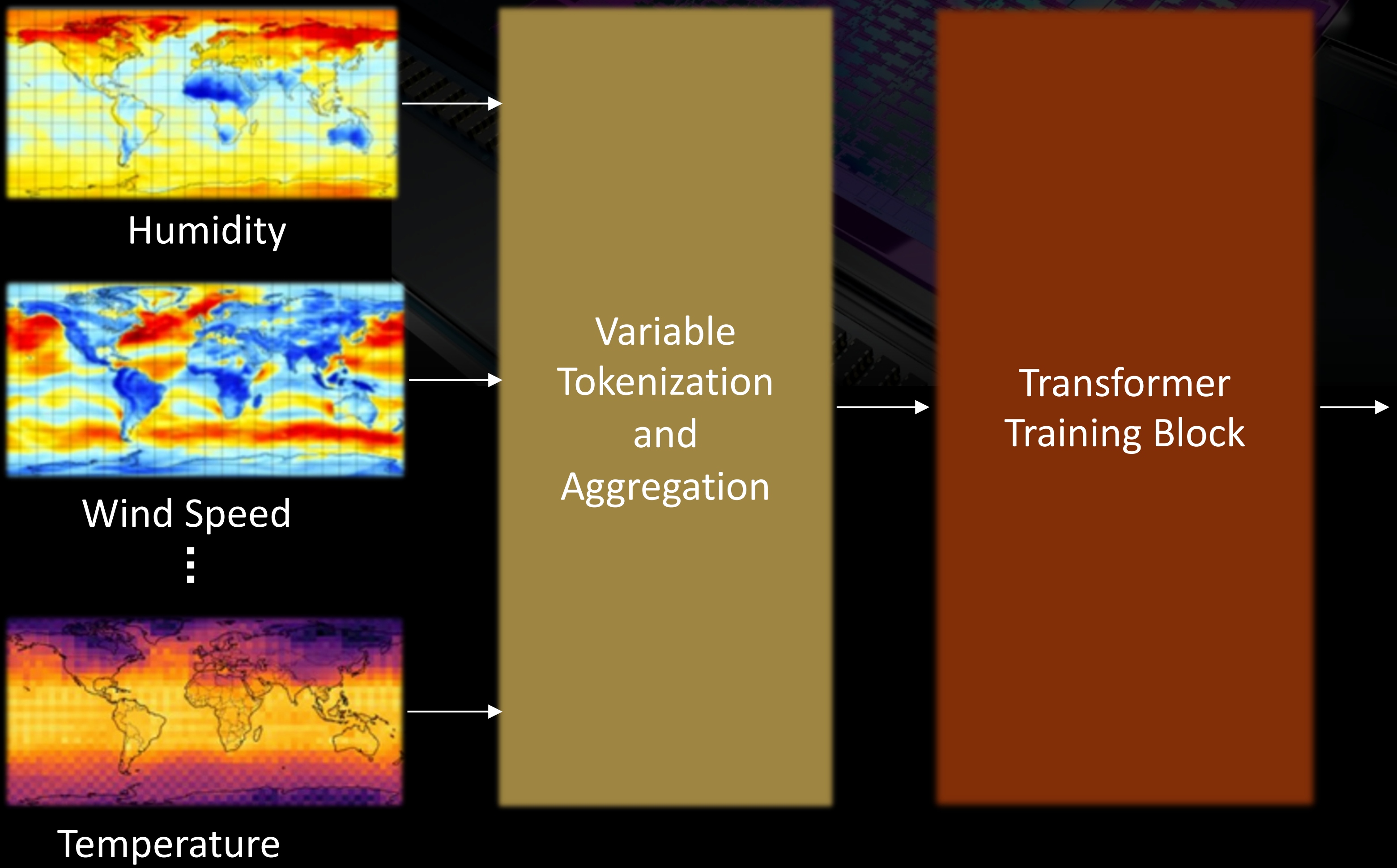
Ground Truth





AMD
together we advance_

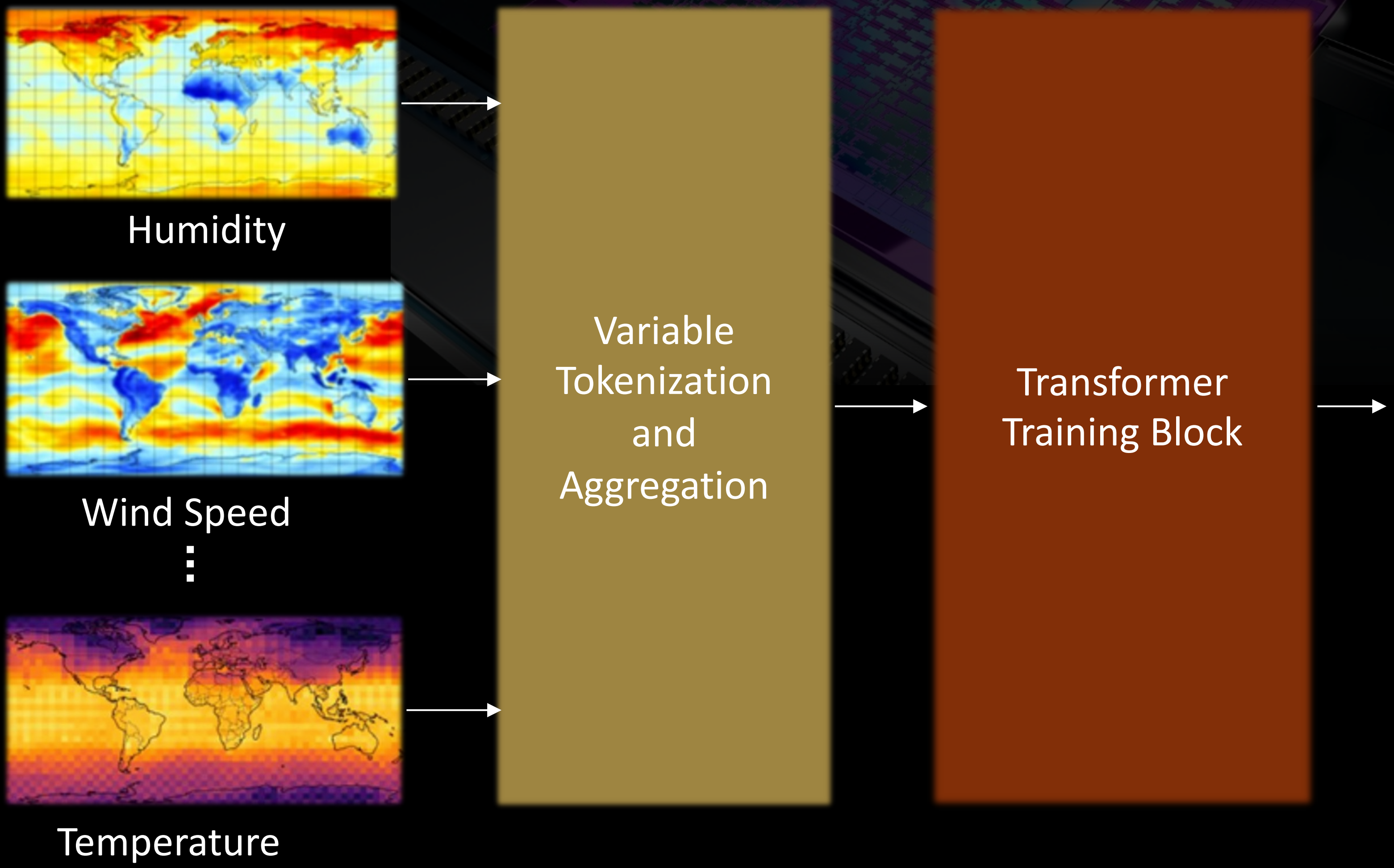
CLIMAX: VISION TRANSFORMER BACKBONE



Application Features

- High-resolution and high-dimensional image data
- Complex spatial dependencies within images

CLIMAX: VISION TRANSFORMER BACKBONE



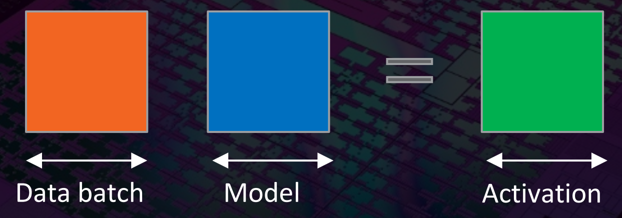
Application Features

- High-resolution and high-dimensional image data
- Complex spatial dependencies within images

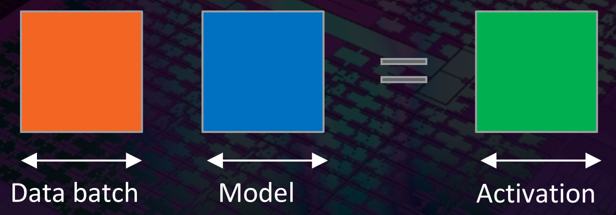
Implications

- More compute power
- Higher memory capacity
- Sophisticated processing for efficient scaling
- Complex parallelization strategies

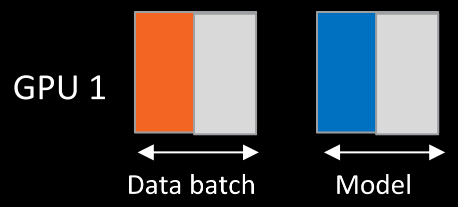
CONVENTIONAL PARALLELIZATION STRATEGIES



CONVENTIONAL PARALLELIZATION STRATEGIES

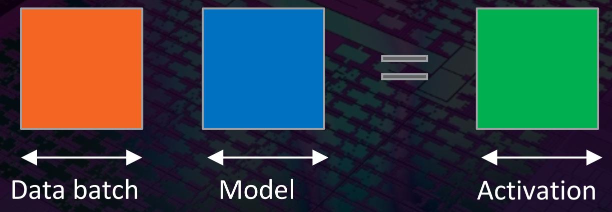


Fully Sharded Data Parallel (FSDP)

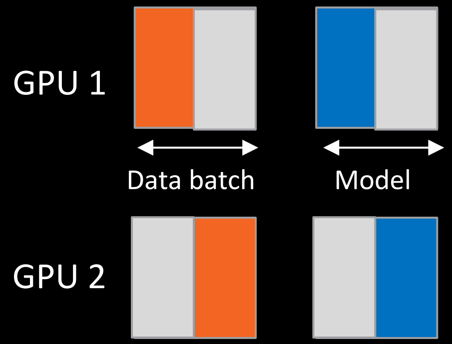


- Each GPU works on a different data batch

CONVENTIONAL PARALLELIZATION STRATEGIES

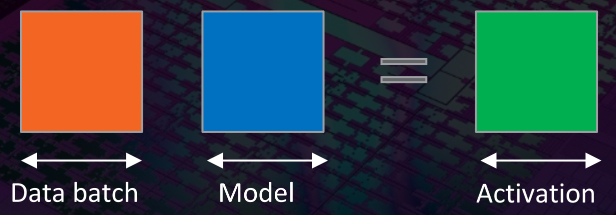


Fully Sharded Data Parallel (FSDP)

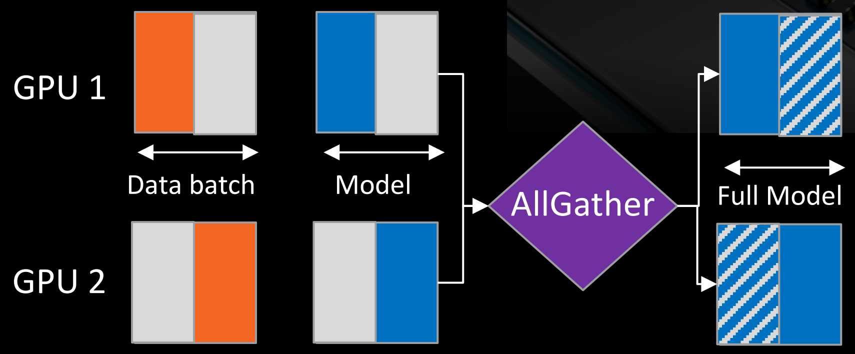


- Each GPU works on a different data batch

CONVENTIONAL PARALLELIZATION STRATEGIES

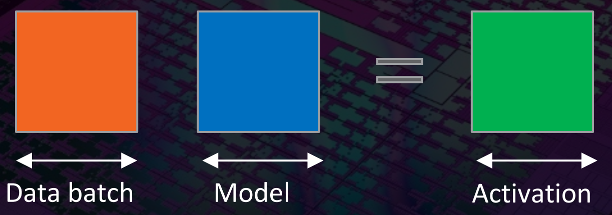


Fully Sharded Data Parallel (FSDP)

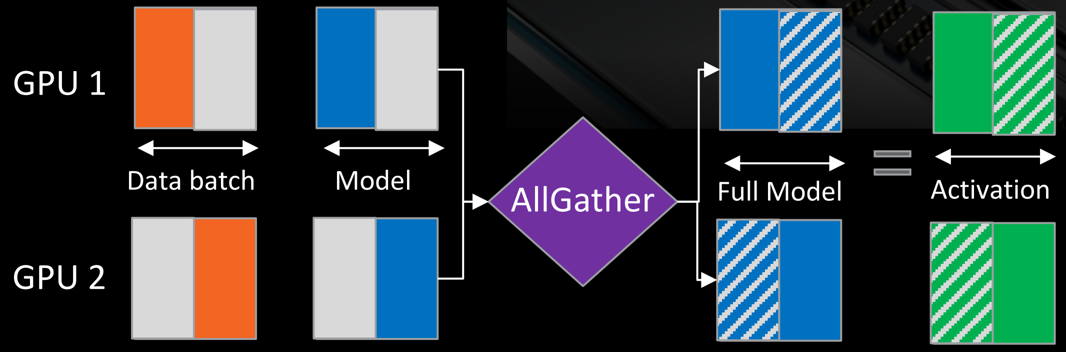


- Each GPU works on a different data batch
- AllGather to bring in all the parameters before compute

CONVENTIONAL PARALLELIZATION STRATEGIES

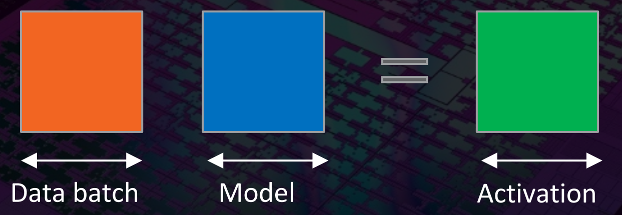


Fully Sharded Data Parallel (FSDP)

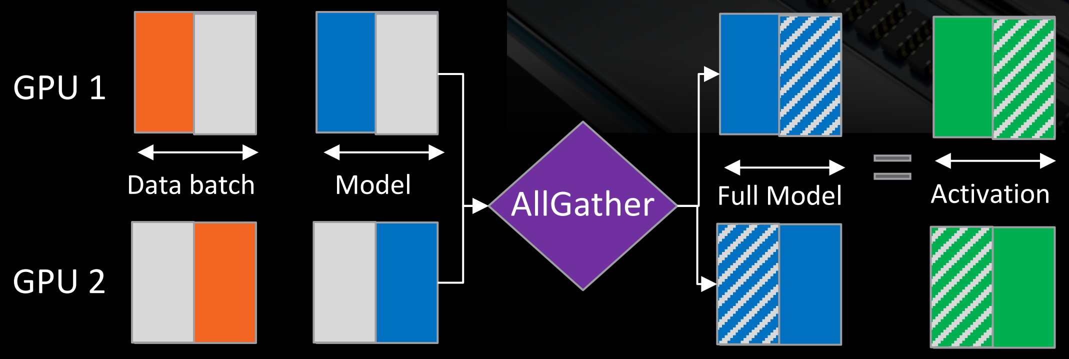


- Each GPU works on a different data batch
- AllGather to bring in all the parameters before compute

CONVENTIONAL PARALLELIZATION STRATEGIES

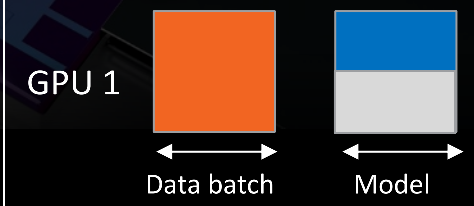


Fully Sharded Data Parallel (FSDP)



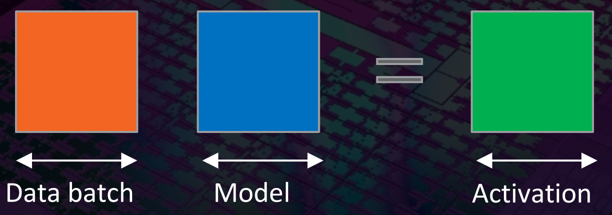
- Each GPU works on a different data batch
- AllGather to bring in all the parameters before compute

Tensor Parallel (TP)

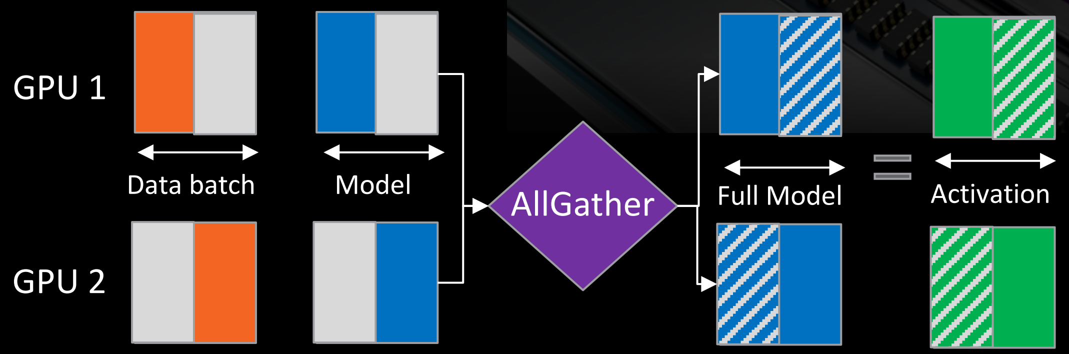


- Each GPU works on partials of entire model

CONVENTIONAL PARALLELIZATION STRATEGIES

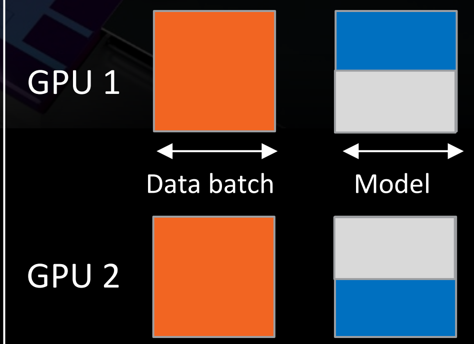


Fully Sharded Data Parallel (FSDP)



- Each GPU works on a different data batch
- AllGather to bring in all the parameters before compute

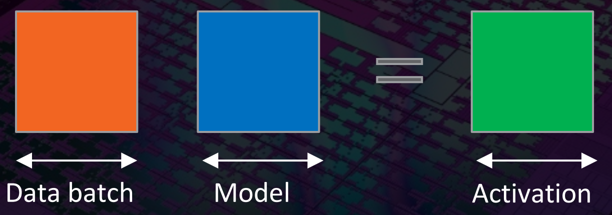
Tensor Parallel (TP)



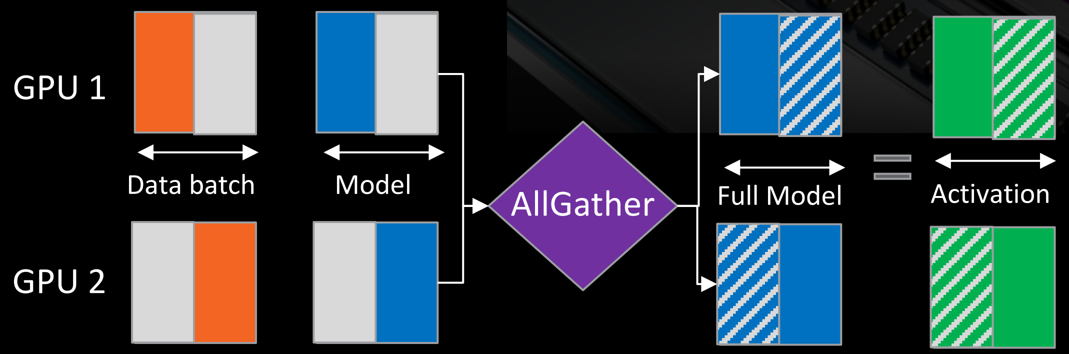
- Each GPU works on partials of entire model

AMD
together we advance_

CONVENTIONAL PARALLELIZATION STRATEGIES

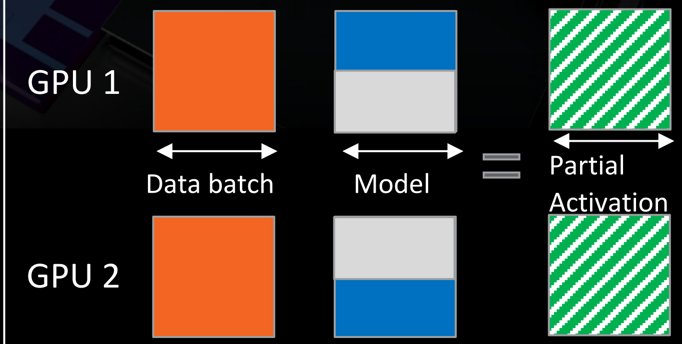


Fully Sharded Data Parallel (FSDP)



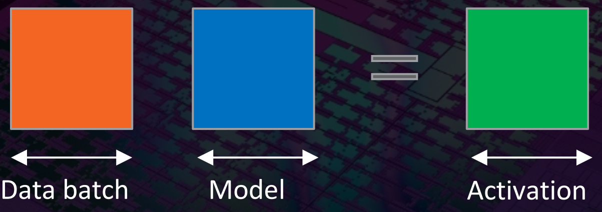
- Each GPU works on a different data batch
- AllGather to bring in all the parameters before compute

Tensor Parallel (TP)

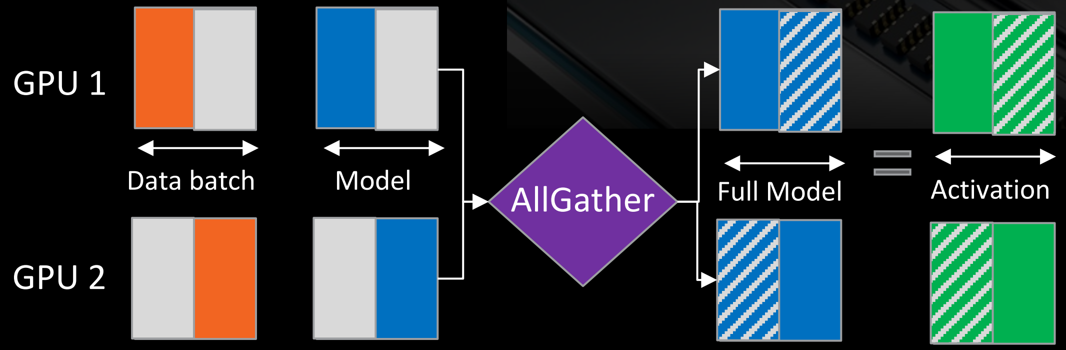


- Each GPU works on partials of entire model

CONVENTIONAL PARALLELIZATION STRATEGIES



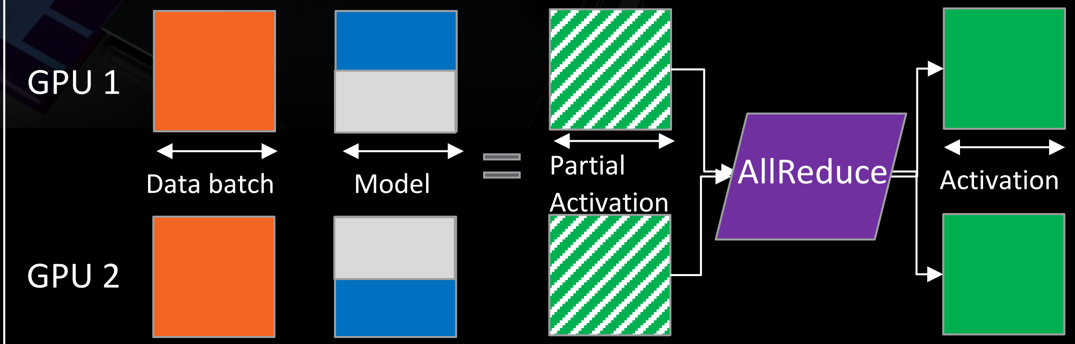
Fully Sharded Data Parallel (FSDP)



- Each GPU works on a different data batch
- AllGather to bring in all the parameters before compute

Performance is limited by the peak memory use when gathering full model parameters

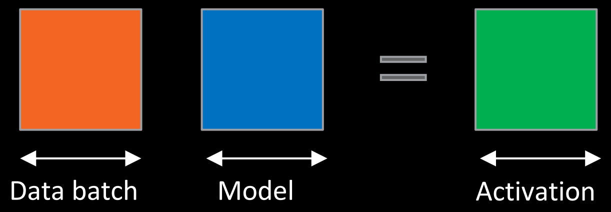
Tensor Parallel (TP)



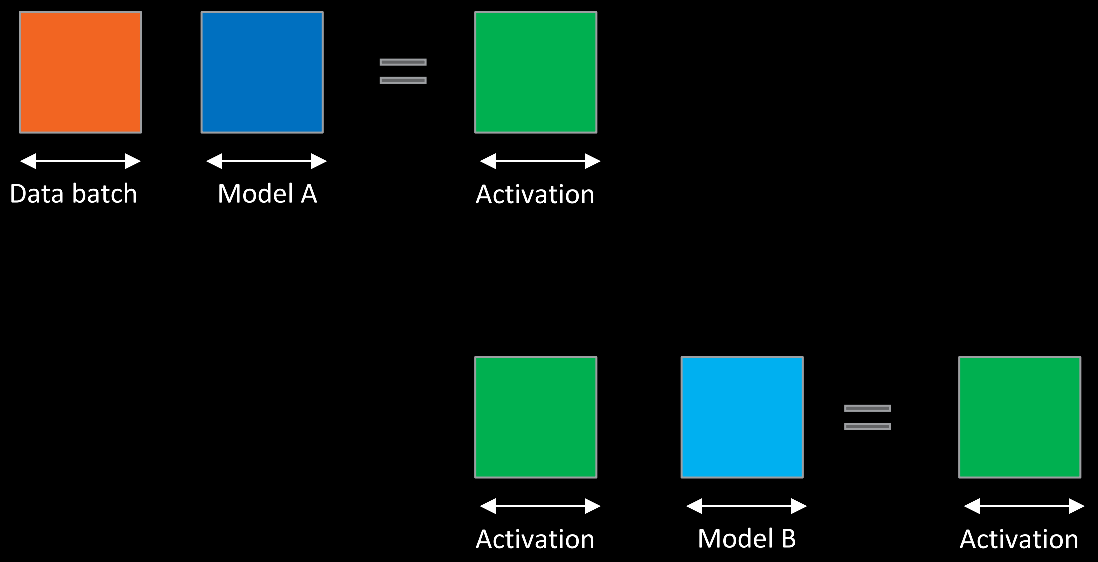
- Each GPU works on partials of entire model
- AllReduce to reduce all the partial activations together

Performance scalability is bottlenecked by the limited number of attention heads

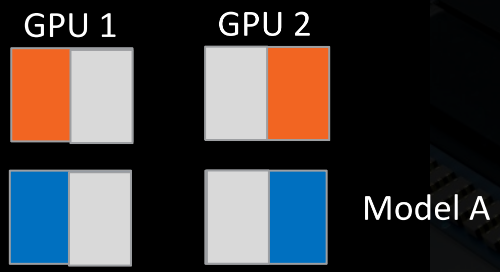
MATRIX CHAIN MULTIPLICATION



MATRIX CHAIN MULTIPLICATION

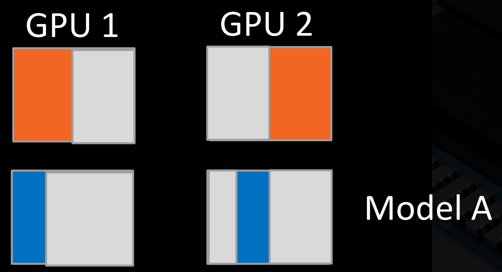


HYBRIDSTOP PARALLELIZATION STRATEGY



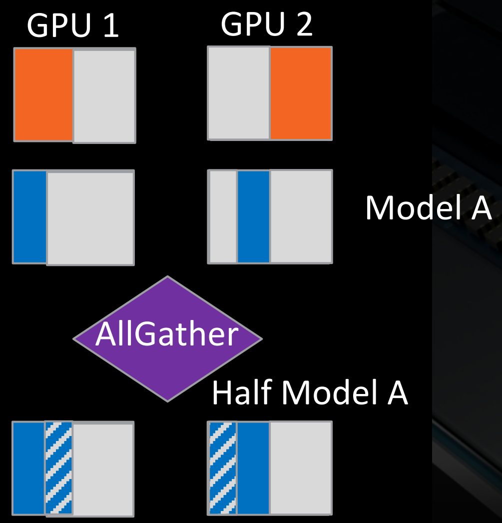
Fusing both FSDP and TP

HYBRIDSTOP PARALLELIZATION STRATEGY



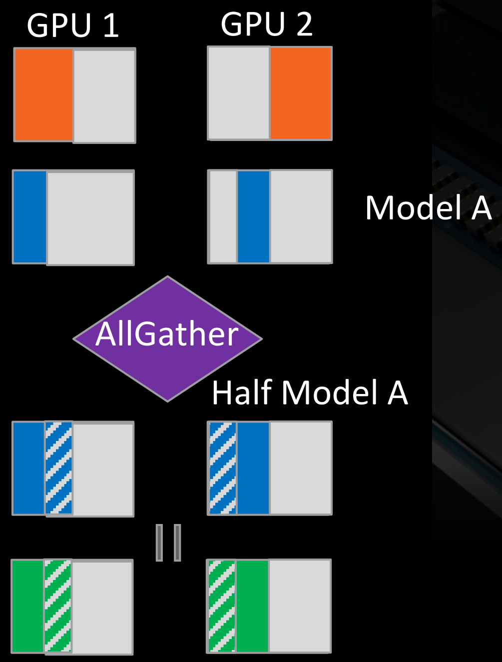
Fusing both FSDP and TP

HYBRIDSTOP PARALLELIZATION STRATEGY



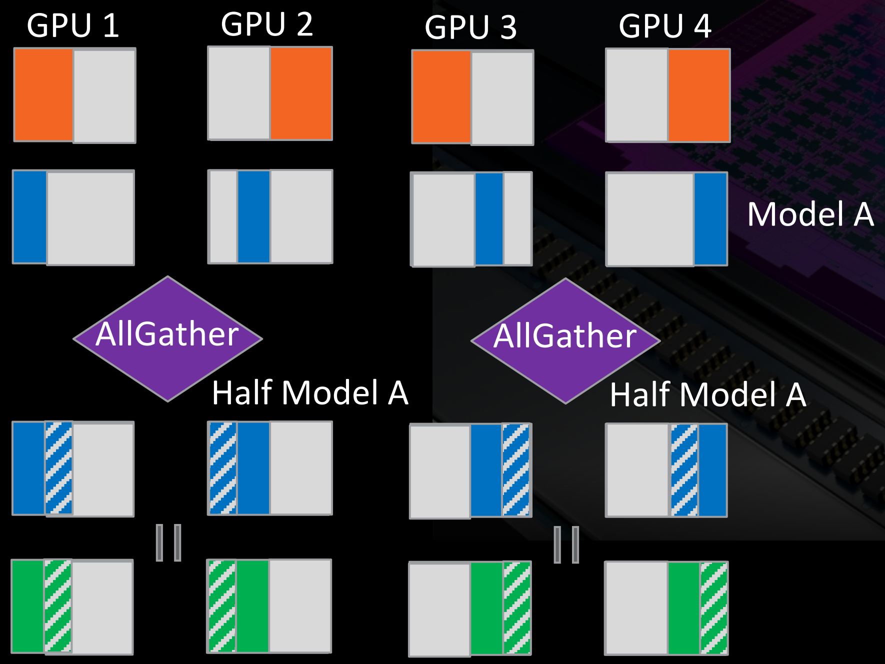
Fusing both FSDP and TP

HYBRIDSTOP PARALLELIZATION STRATEGY



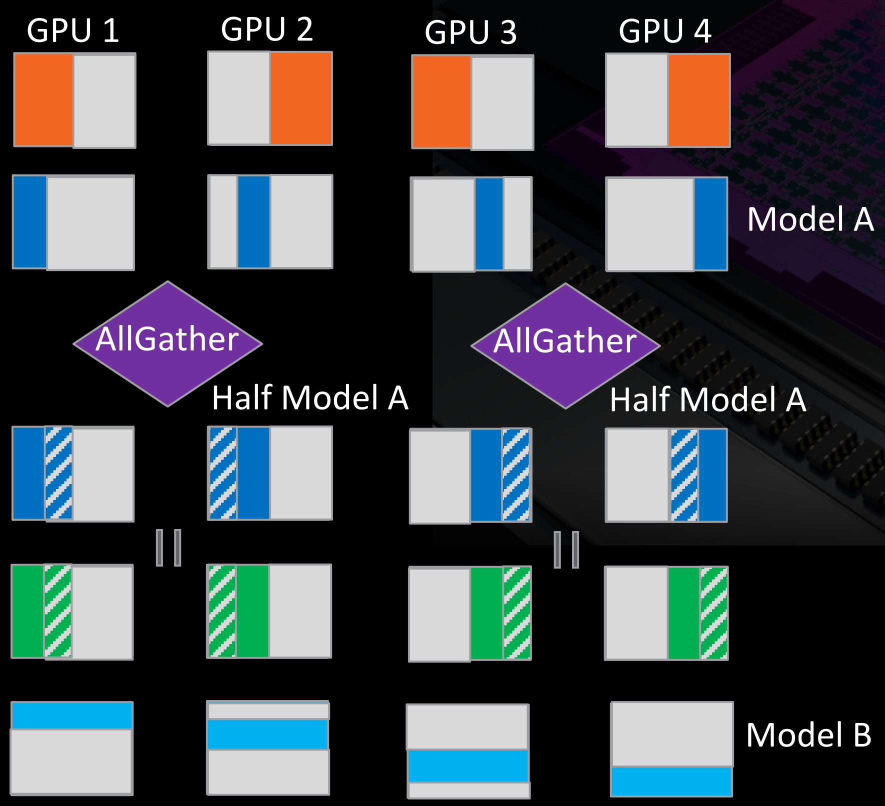
Fusing both FSDP and TP

HYBRIDSTOP PARALLELIZATION STRATEGY



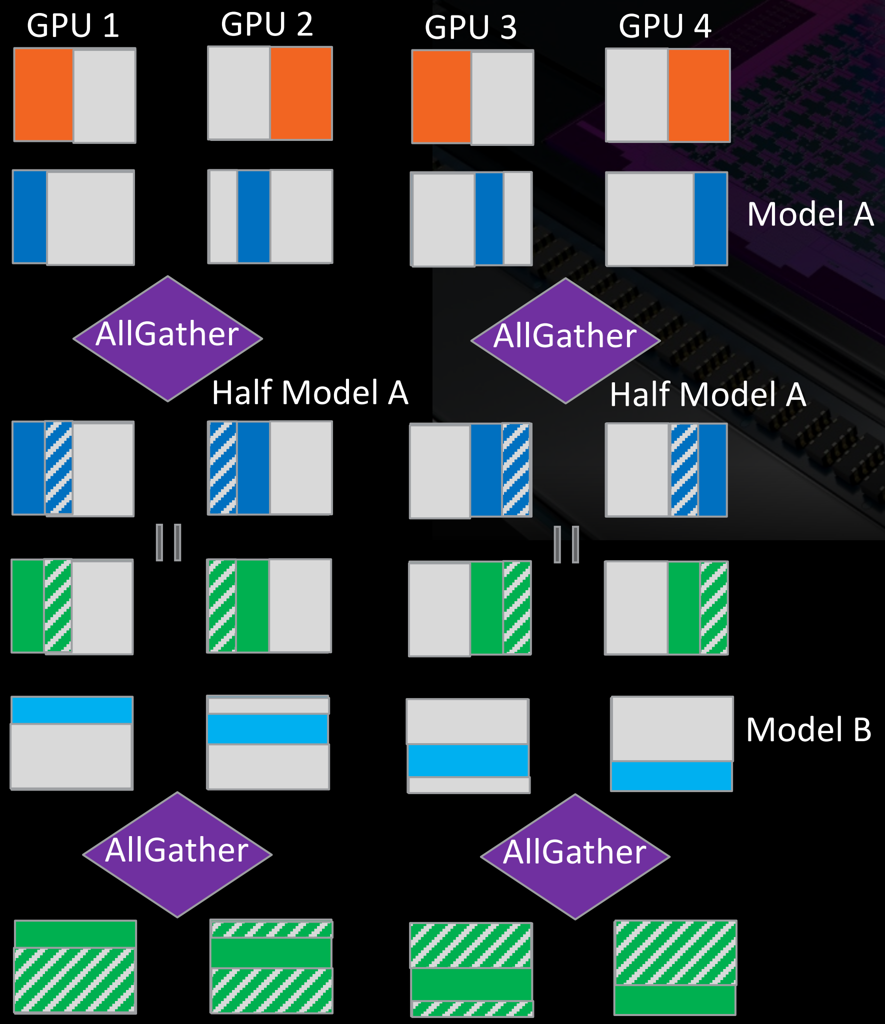
Fusing both FSDP and TP

HYBRIDSTOP PARALLELIZATION STRATEGY



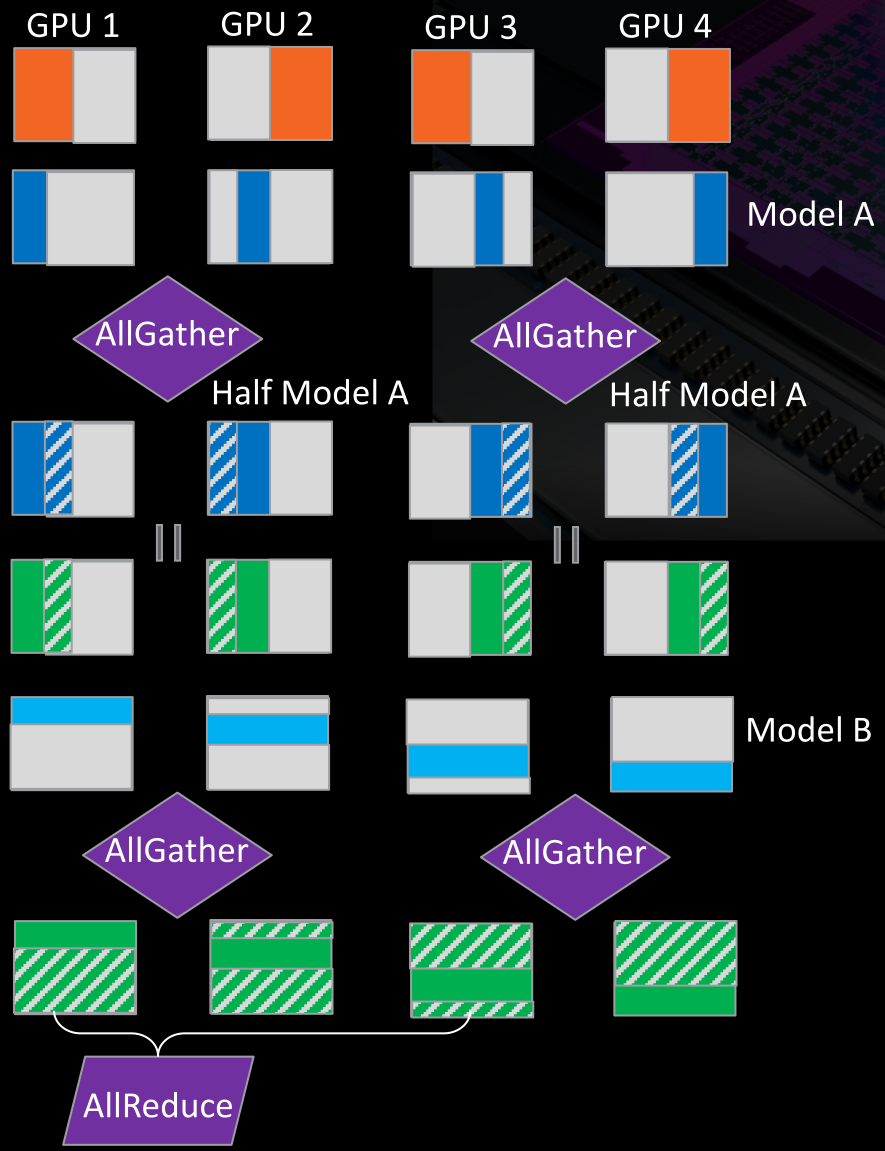
Fusing both FSDP and TP

HYBRIDSTOP PARALLELIZATION STRATEGY



Fusing both FSDP and TP

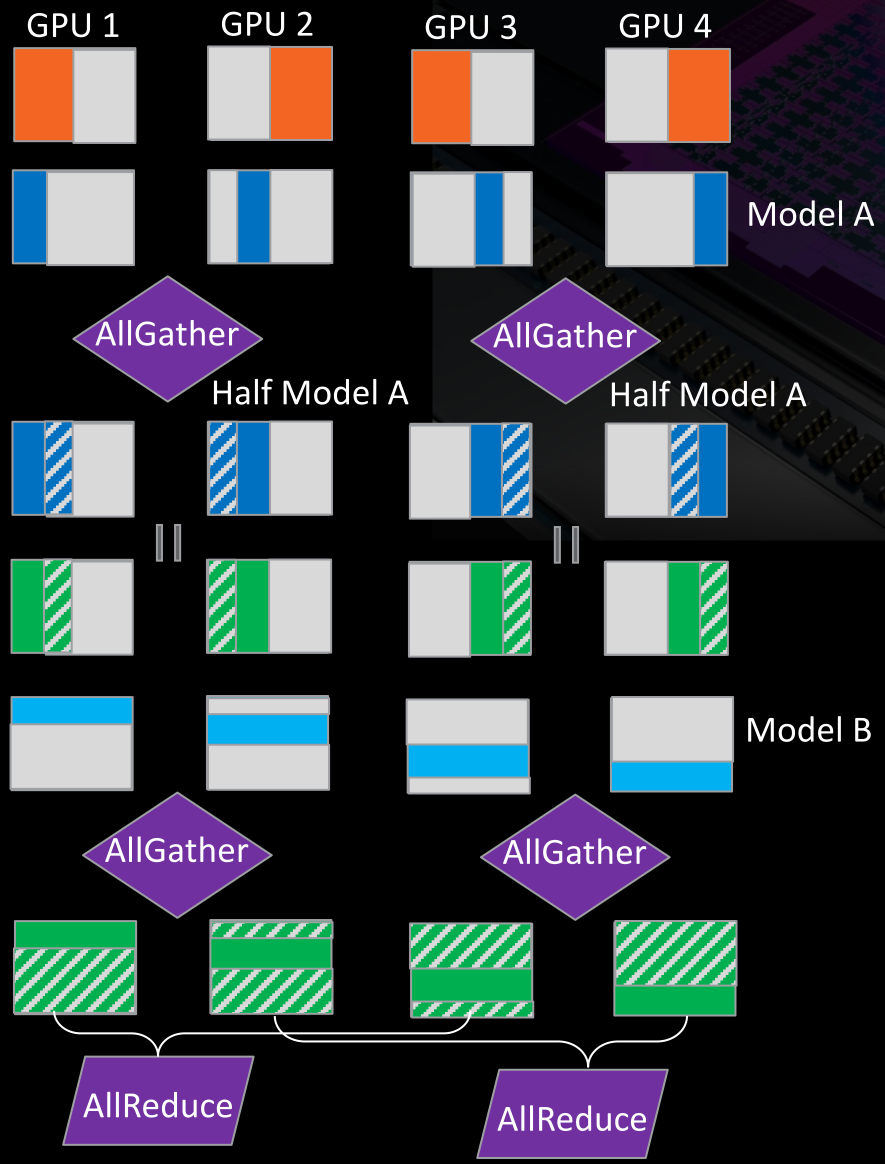
HYBRIDSTOP PARALLELIZATION STRATEGY



Fusing both FSDP and TP

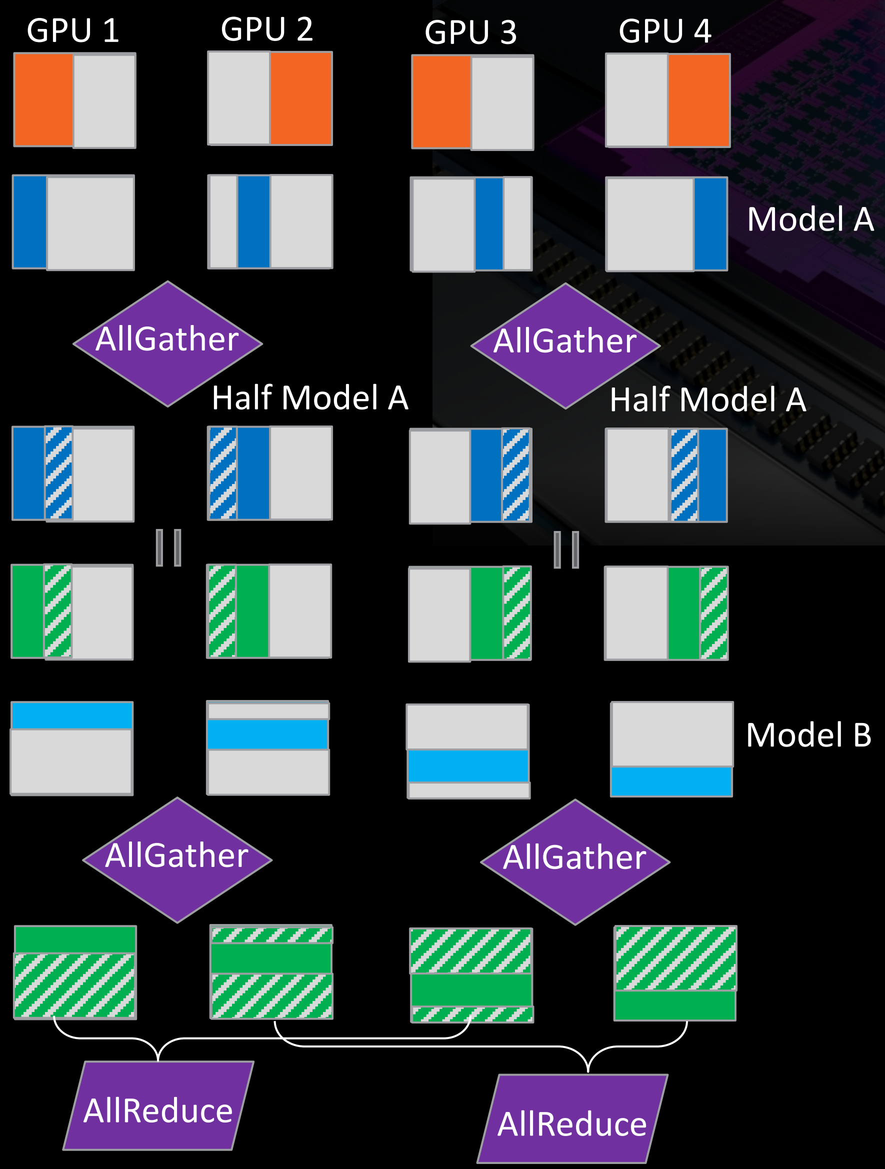
AMD
together we advance_

HYBRIDSTOP PARALLELIZATION STRATEGY



Fusing both FSDP and TP

HYBRIDSTOP PARALLELIZATION STRATEGY



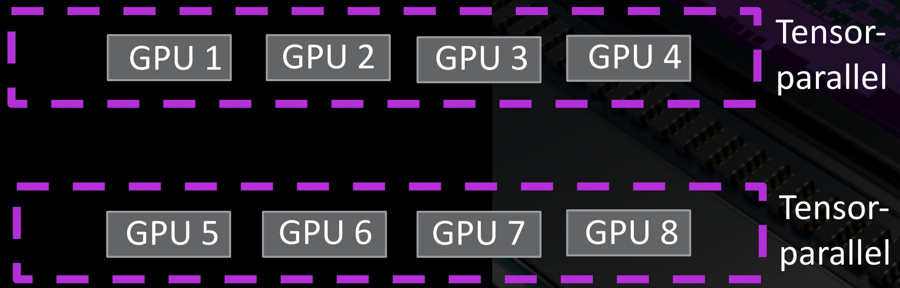
Fusing both FSDP and TP

- Does not need to gather a temporary copy of all parameters like FSDP => Lower peak memory footprint

- Scalability is not limited by the number of attention heads => Higher scalability

AMD
together we advance_

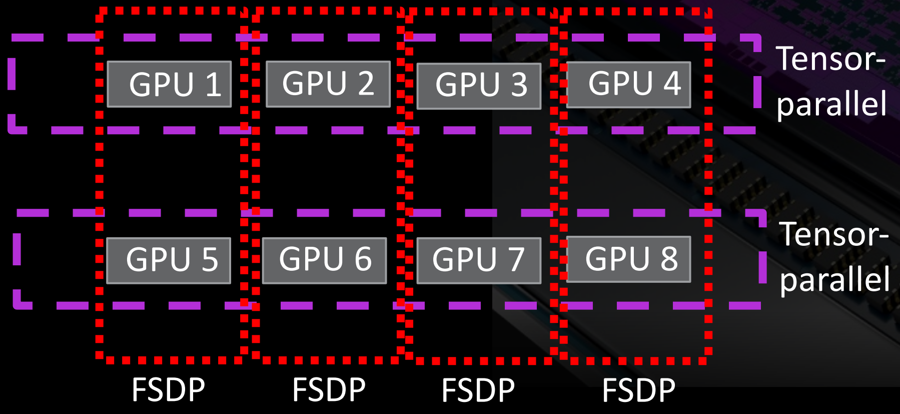
HYBRIDSTOP HIERARCHICAL PARALLELIZATION STRATEGY



- Each horizontal purple rectangle represents a tensor-parallel group.

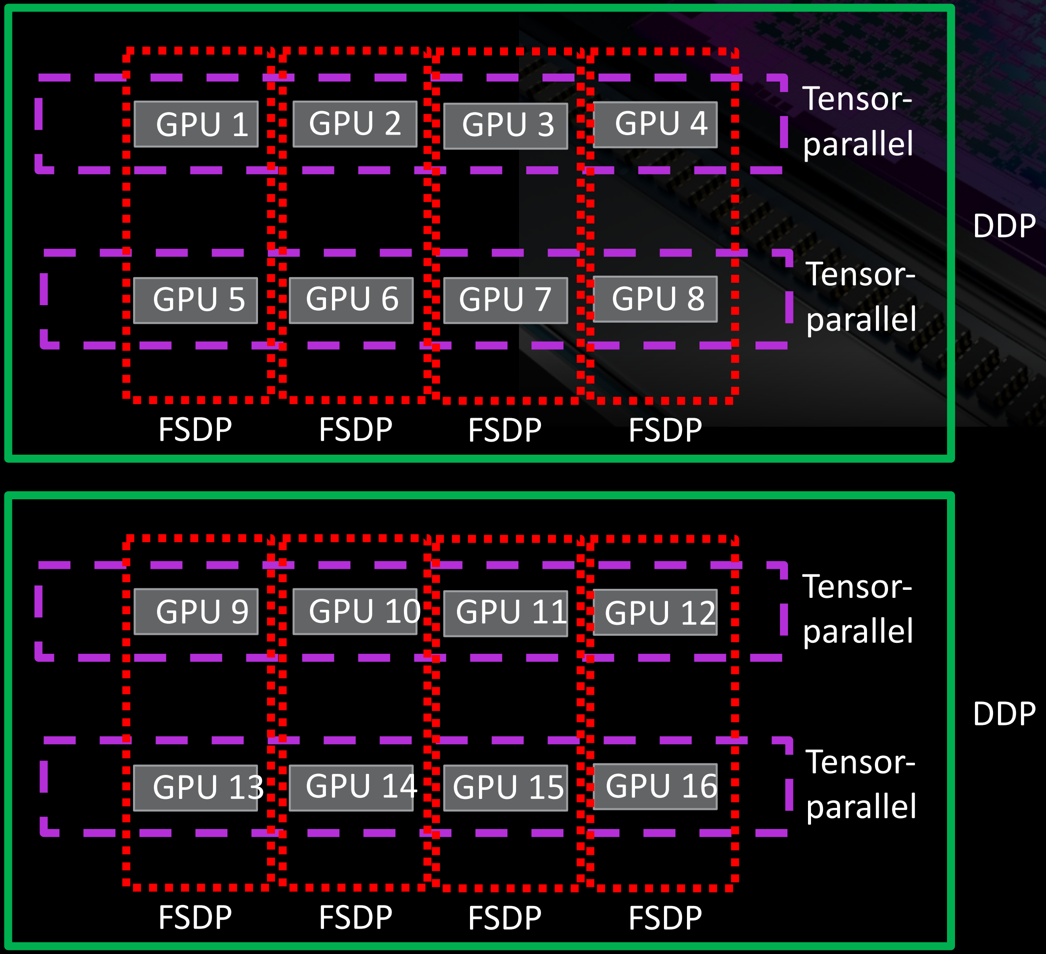
AMD
together we advance_

HYBRIDSTOP HIERARCHICAL PARALLELIZATION STRATEGY



- Each horizontal purple rectangle represents a tensor-parallel group.
- Vertical red rectangles represent Fully Sharded Data Parallel (FSDP) groups.

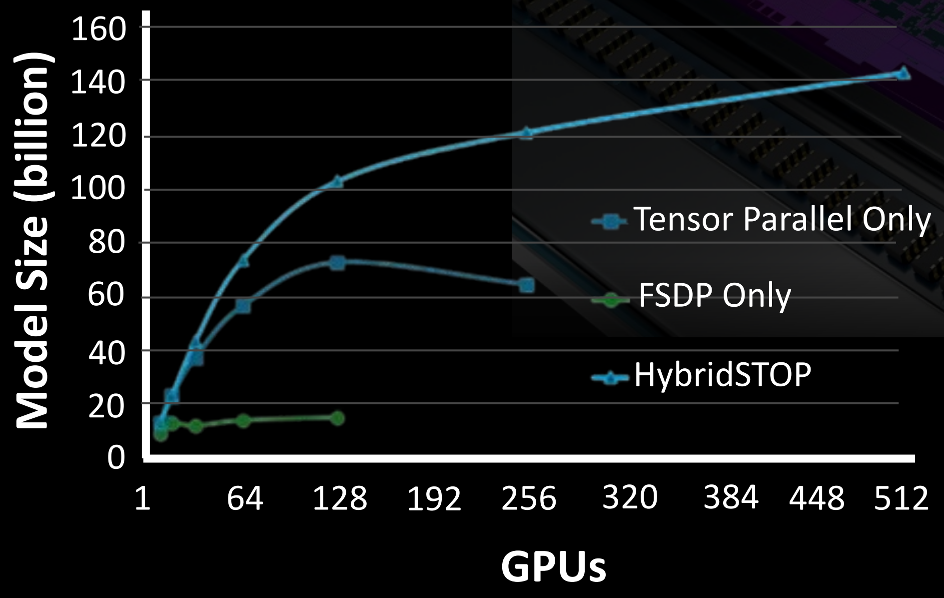
HYBRIDSTOP HIERARCHICAL PARALLELIZATION STRATEGY



- Each horizontal purple rectangle represents a tensor-parallel group.
- Vertical red rectangles represent Fully Sharded Data Parallel (FSDP) groups.
- Green rectangles represent Distributed Data Parallel (DDP) groups.

AMD
together we advance_

HYBRIDSTOP PARALLELIZATION STRATEGY

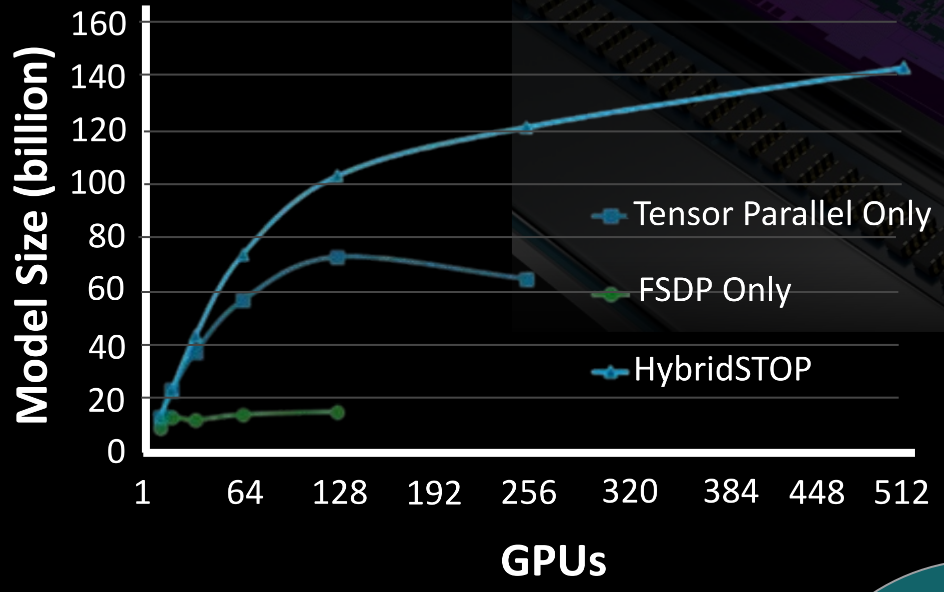


Fusing both FSDP and TP

- Does not need to gather a temporary copy of all parameters like FSDP => Lower peak memory footprint

- Scalability is not limited by the number of attention heads => Higher scalability

HYBRIDSTOP PARALLELIZATION STRATEGY



Fusing both FSDP and TP

- Does not need to gather a temporary copy of all parameters like FSDP => Lower peak memory footprint
- Scalability is not limited by the number of attention heads => Higher scalability

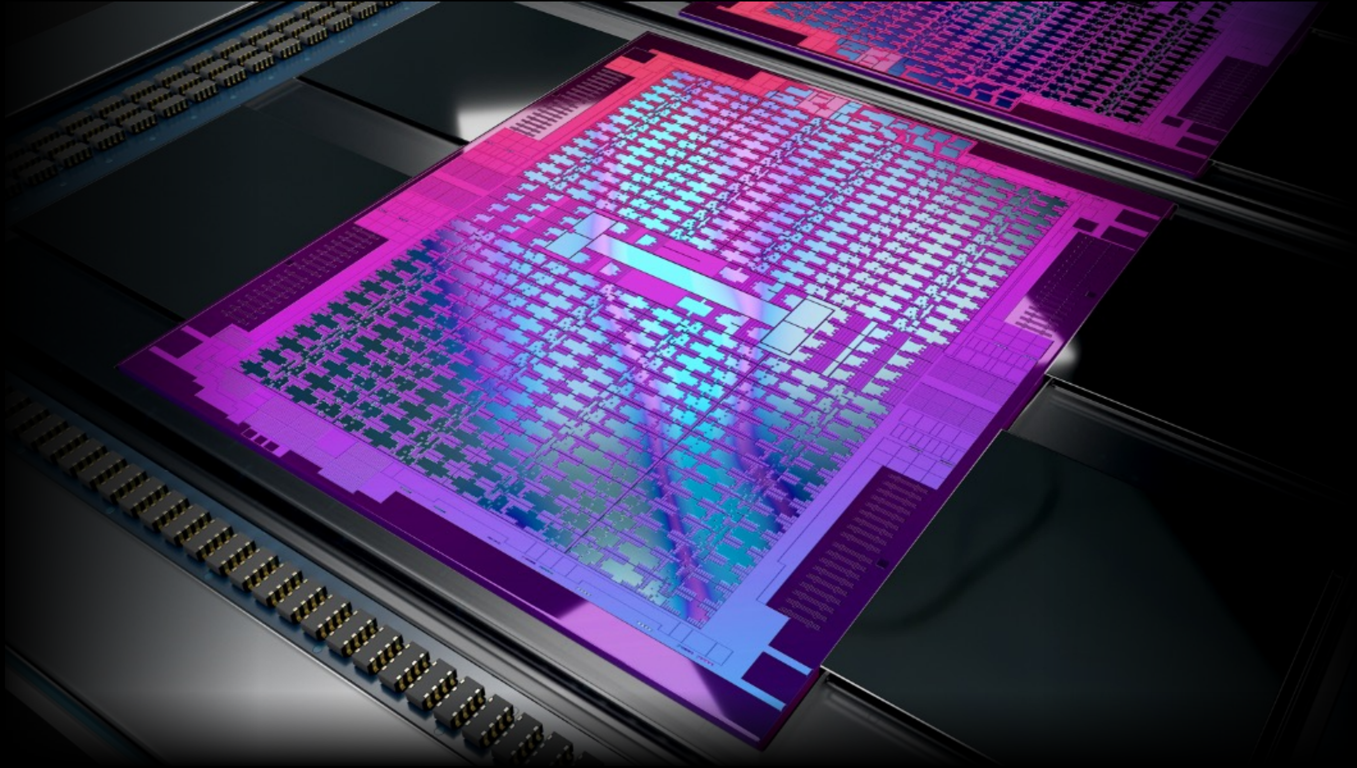
(81-96)% Strong Scaling*

*On 24,576 MI250X GCDs

AMD
together we advance_

SCALING VISION TRANSFORMERS SUMMARIZED

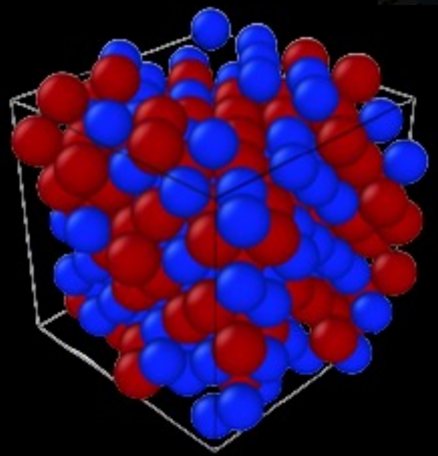
1. Complexity of Environmental Systems
   - Predicting Earth system processes requires robust, adaptable, and scalable computational models due to their inherent complexity and numerous influencing variables.
2. Limitations of FSDP and TP
   - FSDP is constrained by peak memory use during model gathering, and tensor parallelism is limited by attention heads.
3. Efficient AI Scaling
   - Hybrid Sharded Tensor-Data Orthogonal Parallelism (HybridSTOP) retains 81-96% strong scaling efficiency at 24,576 GPUs, overcoming these limitations.
4. Broad Applicability
   - The proposed techniques benefit fields with large datasets like astrophysics and biology, enhancing AI and HPC integration.



HYDRA-GNN: A SCALABLE GNN ARCHITECTURE FOR MATERIALS SCIENCE APPLICATIONS

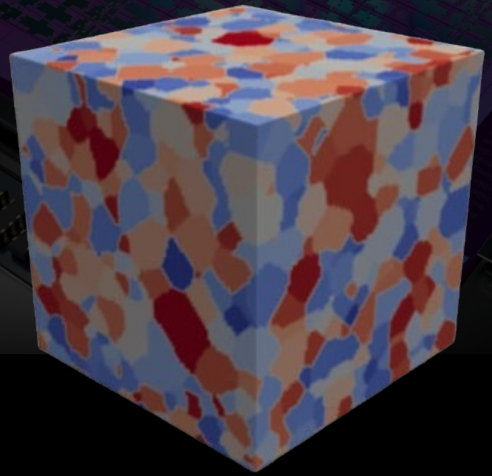
GRAPH REPRESENTATIONS OF MATERIALS AT DIFFERENT SCALES

Atomic Scale



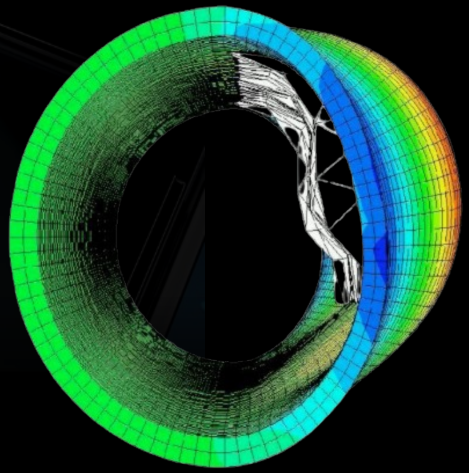
Nodes = atoms
Edges = interatomic bonds

Mesoscale



Nodes = Voronoi centers
Edges = connection between Voronoi centers

Continuum Scale



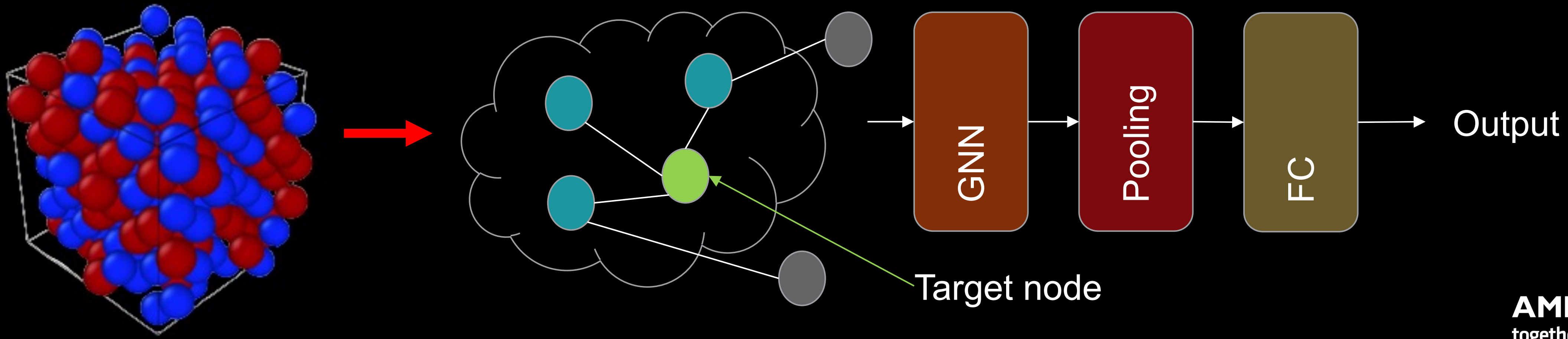
Nodes = vertices of the finite element mesh
Edges = edges of the finite element mesh

Image Sources: https://link.springer.com/article/10.1007/s11837-019-03808-x; http://biomechanics.stanford.edu/me309/me309_c01.pdf;

GNN ARCHITECTURE

The architecture of GNN is made of:

1. A graph embedding layer
2. Hidden graph layers capturing the short-range interactions between nodes
3. Pooling layers interleaved with graph layers
4. Fully connected (FC) dense layers



HYDRA-GNN: CRITICAL PROPERTIES

HydraGNN

- Capability to simultaneously read data from multiple sources
- Scalable training on leadership class supercomputing facilities
- Flexibility to support diverse DOE-relevant scientific applications
- Portability across heterogeneous computing environments
- Upgraded software dependencies

HYDRA-GNN: CRITICAL PROPERTIES

HydraGNN

- Capability to simultaneously read data from multiple sources
- Scalable training on leadership class supercomputing facilities
- Flexibility to support diverse DOE-relevant scientific applications
- Portability across heterogeneous computing environments
- Upgraded software dependencies

HYDRA-GNN: CRITICAL PROPERTIES

HydraGNN

- Capability to simultaneously read data from multiple sources
- Scalable training on leadership class supercomputing facilities
- Flexibility to support diverse DOE-relevant scientific applications
- Portability across heterogeneous computing environments
- Upgraded software dependencies

HYDRA-GNN: CRITICAL PROPERTIES

HydraGNN

- Capability to simultaneously read data from multiple sources
- Scalable training on leadership class supercomputing facilities
- Flexibility to support diverse DOE-relevant scientific applications
- Portability across heterogeneous computing environments
- Upgraded software dependencies

HYDRA-GNN: CRITICAL PROPERTIES

- HydraGNN
  - Capability to simultaneously read data from multiple sources
  - Scalable training on leadership class supercomputing facilities
  - Flexibility to support diverse DOE-relevant scientific applications
  - Portability across heterogeneous computing environments
  - Upgraded software dependencies

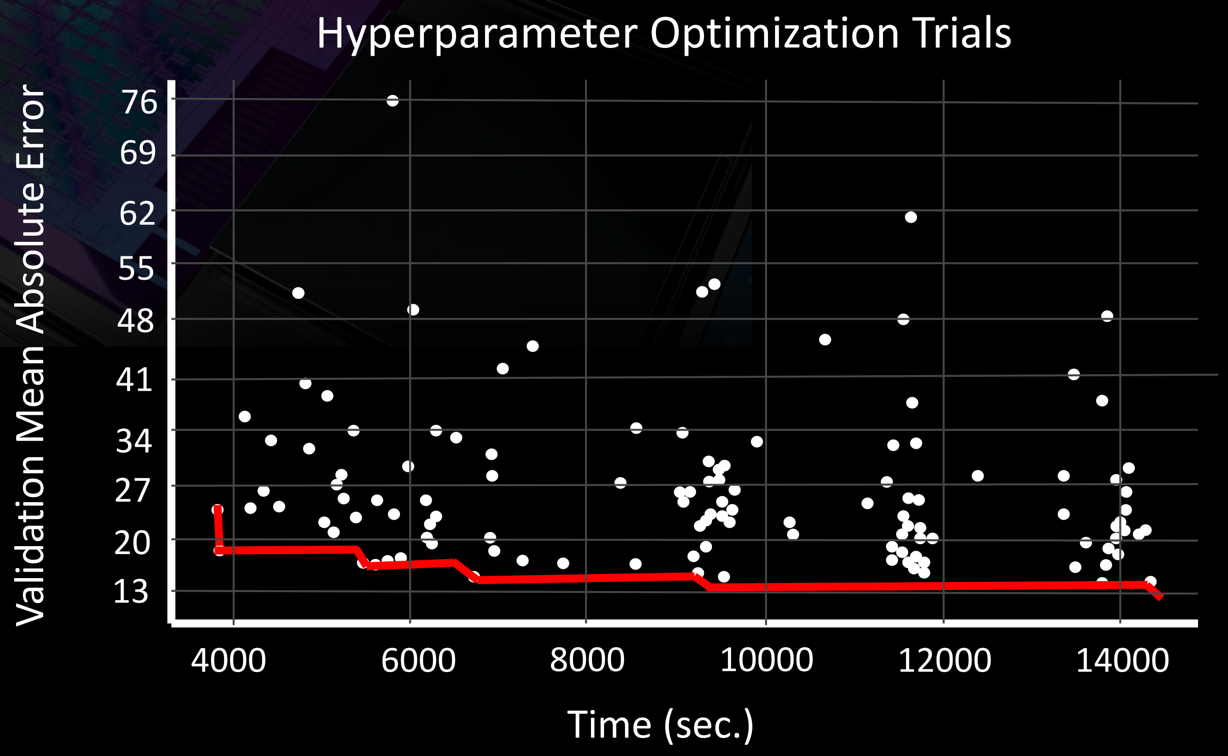
HYDRA-GNN: CRITICAL PROPERTIES

HydraGNN

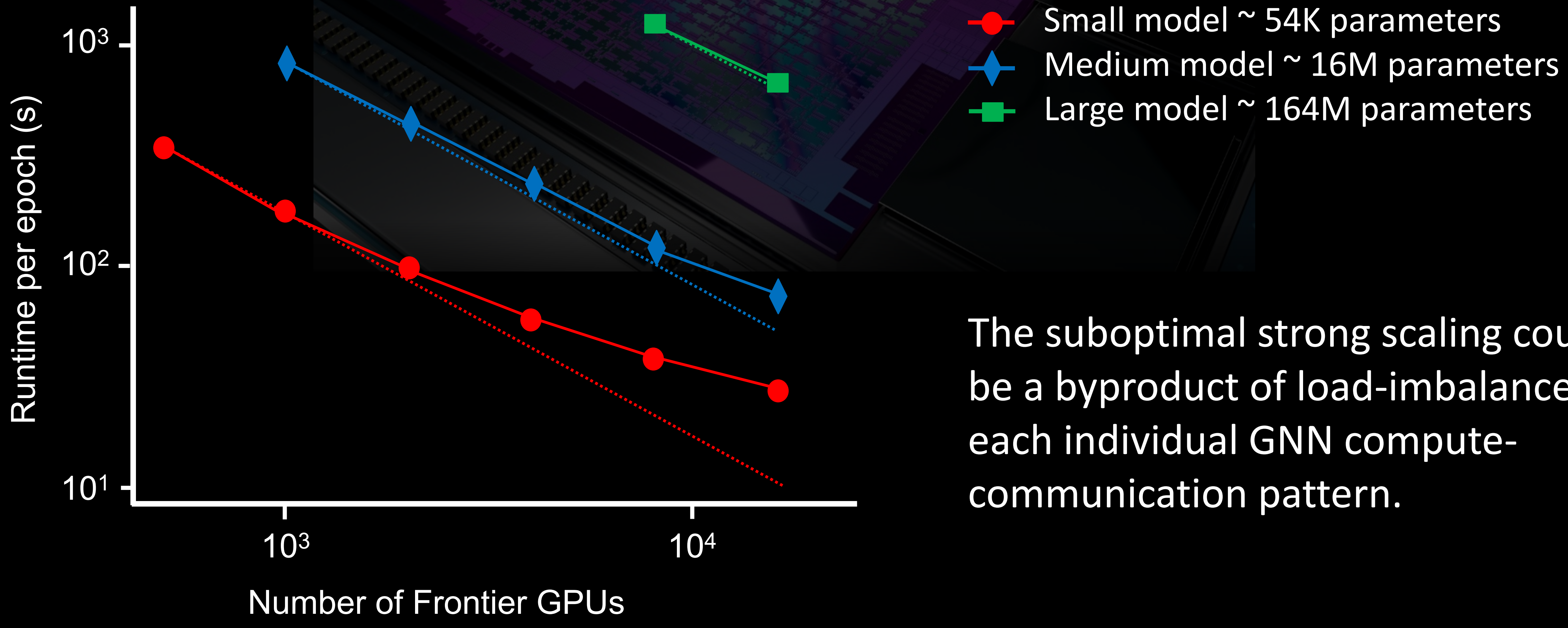
- Capability to simultaneously read data from multiple sources
- Scalable training on leadership class supercomputing facilities
- Flexibility to support diverse DOE-relevant scientific applications
- Portability across heterogeneous computing environments
- Upgraded software dependencies

EXTREME-SCALE HYPERPARAMETER OPTIMIZATION TRIALS

- Conduct hyperparameter trails (HPO) to determine the top-K best GNN configurations
- Run the top-K GNNs in an ensemble fashion
- 8,192 nodes of OLCF Frontier (87% of the machine) have been used to explore 200 unique hyperparameter configurations.



STRONG SCALING OF HYDRA-GNN ON FRONTIER



The suboptimal strong scaling could be a byproduct of load-imbalance in each individual GNN compute-communication pattern.

AMD
together we advance_

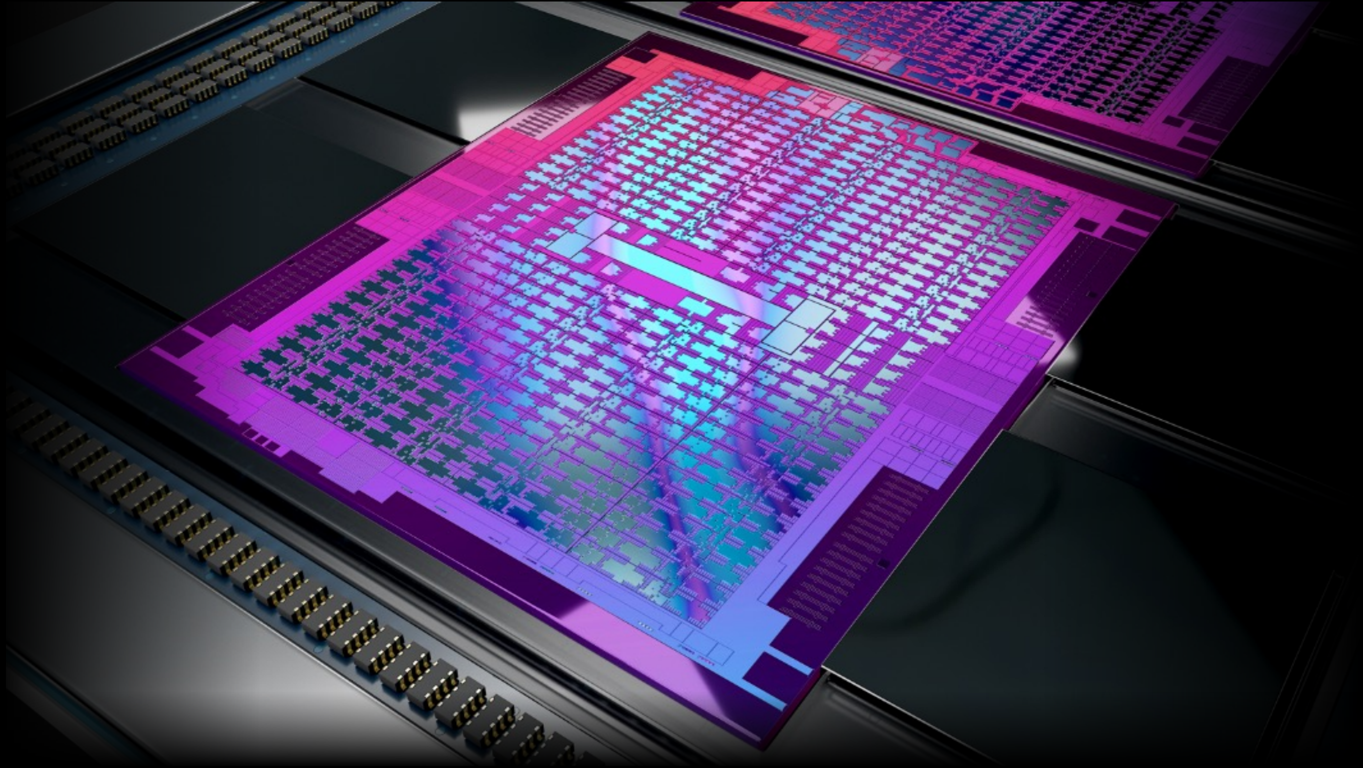
SCALING GRAPH NEURAL NETWORKS SUMMARIZED

1. Application Features
   - Materials at different scaling (e.g., atomic scale, mesoscale) are represented as graphs and processed using graph neural networks.

2. Model Selection and Hyperparameter Optimization
   - Conduct HPO trails and determine the top-K GNN hyperparameter configurations that result in the least mean absolute error. Run all the top-K GNN models in an ensemble fashion for the execution of the application.

3. Critical Properties of Hydra-GNN
   - Hydra-GNN must satisfy five critical properties corresponding to flexibility, scalability, and heterogeneity in applications, software, and hardware support.

4. Performance Achievements
   - Achieved high GPU throughput and strong scaling efficiencies up to few thousand GPUs.



CALL TO ACTION

CALL TO ACTION: GPU UTILIZATION

While our experiments show impressive strong and weak scaling, there is still opportunity for improvement

- GPU utilization is under 40% for all model sizes

Potential Opportunities

- Exploring opportunities to overlap communication with computations
- Efficient parallelization strategies that optimize data movement and memory accesses.
- Research novel attention algorithms

CALL TO ACTION: LOAD BALANCING

- Exploring load balancing opportunities while executing a single GNN model across multiple CUs / GPUs
- Exploring load balancing opportunities when running an ensemble of GNN architecture each with a different set of hyperparameters – architecture, number of layer, FLOPs, etc.
- Run-time / dynamic memory allocation to improve GPU memory utilization

Load balancing of a single GNN model

Load balancing across multiple GNN models

GPU memory utilization improvements

TAKEAWAYS

1. Significant Impact of AI-for-Science:
   - AI-for-Science applications can significantly help better lives, economies, and communities.

2. Frontier's Role:
   - Frontier, our cutting-edge hardware, along with AMD ROCm software support, stands as the essential backbone, driving the execution of these models with unmatched efficiency.

3. Three Foundational Models:
   - We've categorized these applications into 3 foundational models, each tailored to specific computational and communication needs and challenges.

4. Challenges and Roadblocks
   - GPU Compute and Memory Utilization: Optimizing the use of GPU resources remains a critical challenge.
   - Load Balancing: Efficiently distributing workloads to maximize performance is another significant hurdle.

5. Future Directions
   - We are working on more sophisticated tools to do nuanced analysis, characterization, and optimization of ML models at scale

ACKNOWLEDGEMENTS

- **Ashwin Aji**
  Organization: Advanced Micro Devices
- **Karl W. Schulz**
  Organization: Advanced Micro Devices
- **Michael Schulte**
  Organization: Advanced Micro Devices
- **Austin Ellis**
  Organization: Advanced Micro Devices
- **Jorda Polo**
  Organization: Advanced Micro Devices
- **Angela Dalton**
  Organization: Advanced Micro Devices
- **Massimiliano Lupo Pasini**
  Organization: ORNL
- **Aristeidis Tsaris**
  Organization: ORNL
- **Leon Song**
  Organization: Microsoft
- **Sajal Dash**
  Organization: ORNL
- **Pei Zhang**
  Organization: ORNL
- **Jong Youl Choi**
  Organization: ORNL
- **Prasanna Balaprakash**
  Organization: ORNL
- **John Gounley**
  Organization: ORNL
- **Xiao Wang**
  Organization: ORNL
- **Kshitij Mehta**
  Organization: ORNL
- **Dan Lu**
  Organization: ORNL

Relevant Sources

1. Sajal Dash et. al, “Optimizing Distributed Training on Frontier for Large Language Models”, arXiv, 2023
2. T. Nguyen et. al, “Climax: A foundation model for weather and climate,” 2023
3. Xiao Wang et. al, “ORBIT: Oak Ridge Base Foundation Model for Earth System Predictability”, arXiv, 2024
4. “HydraGNN - Distributed PyTorch implementation of multi-headed graph convolutional neural networks”, Computing and Computational Sciences Directorate, Oak Ride National Laboratory
5. "HydraGNN: Distributed PyTorch implementation of multi-headed graph convolutional neural networks", Copyright ID#: 81929619 https://doi.org/10.11578/dc.20211019.2

Oak Ridge National Laboratory

Microsoft

AMD

AMD

Copyright and Disclaimer