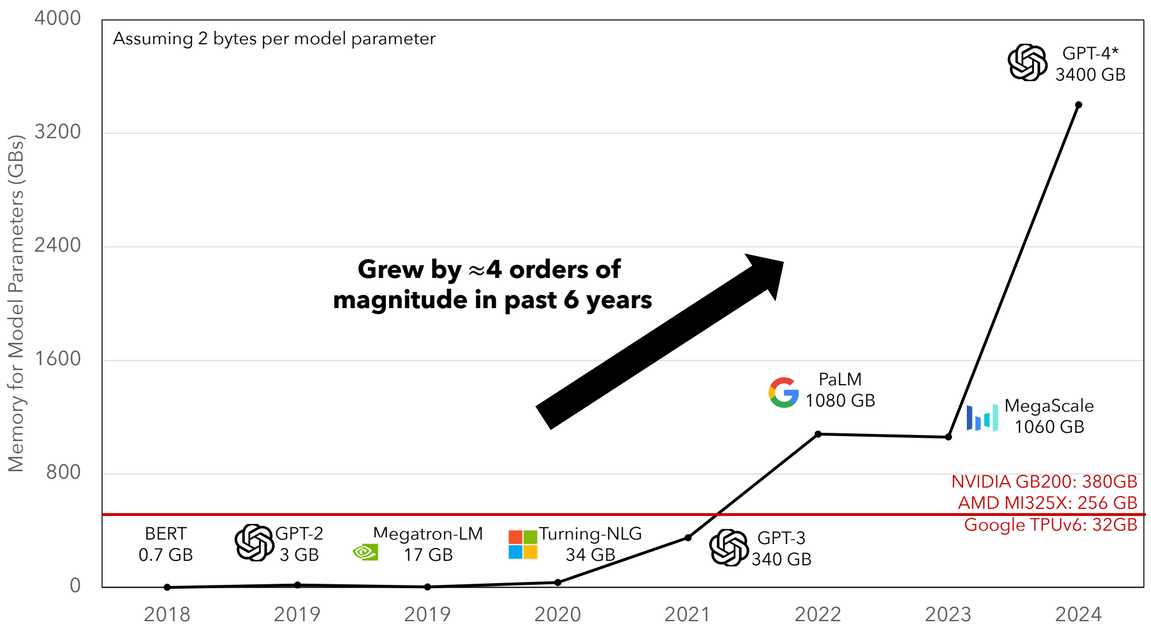
ReCycle: Resilient Training of Large DNNs using Pipeline Adaptation

Swapnil Gandhi, Mark Zhao, Athinagoras Skiadopoulos, Christos Kozyrakis

Models Are Becoming Larger

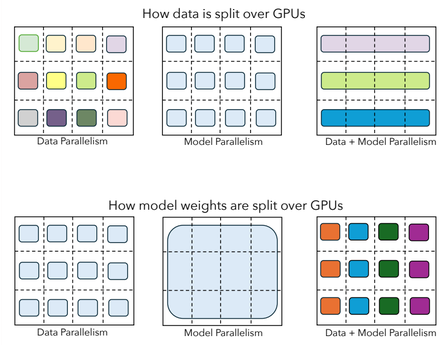
Recent work in language modeling demonstrates that training large transformer models advances the state of the art in NLP applications.

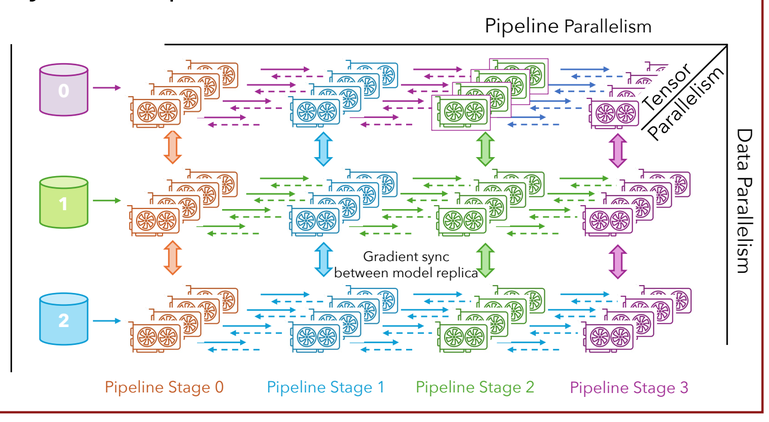
However, their compute and memory requirement far outstrips the capacity of a single GPU.



Distributed Training is Becoming a Norm

DNN training frameworks use a combination of Tensor, Pipeline, and Data parallelism to efficiently scale up to thousands of GPUs.





Trade-off in Distributed Training

Pick One	Using all GPUs for training	Reserving some GPUs as hot spares	ReCycle
Performance	✓ No Overhead in Fault-Free Case	✗ Constant Overhead; Spares remain idle in Fault-Free Case	✓
Resiliency	✗ Training stalls when a GPU fails	✓ Hot spare ensures continual training in presence of faults	✓

Failures Getting Noticeable

As training scales up and extends over longer durations, the likelihood of encountering failures also rises.

Reports about the impact of failures in training large models:

*"During a 54-day snapshot period of pre-training, we experienced a total of **466 job interruptions**....Approximately 78% of the unexpected interruptions are attributed to confirmed hardware issues, such as GPU or host component failures..."*

- Llama Team @ META[1]

*"This is a particularly annoying problem to handle as **if one GPU has an issue,** the synchronized nature of distributed training means that **all GPUs get stuck**."*

- LAION Team[2]

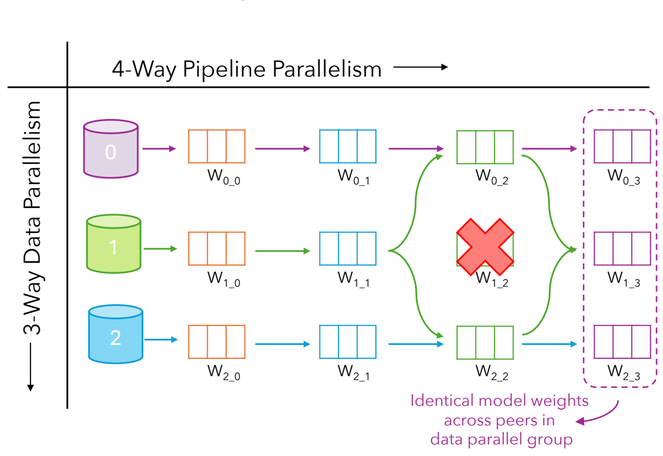
*"Estimated 100+ host restarts due to hardware failures over the course of 2 months...**178,000 GPU hours of wasted time due to various malfunctions**"*

- OPT 175B Team[3]

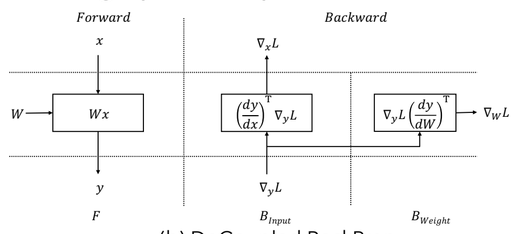
[1] The Llama 3 Herd of Models. https://arxiv.org/pdf/2407.21783
[2] Large Scale Openclip: L/14, H/14 And G/14 Trained On LAION-2B. https://laion.ai/blog/large-openclip/
[3] OPT: Open Pre-trained Transformer Language Models. https://arxiv.org/abs/2205.01068

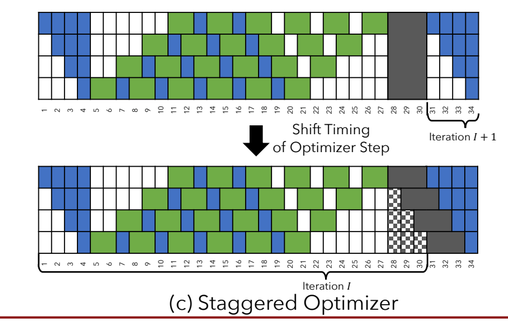
Techniques in ReCycle

Key Insight: We leverage inherent **functional redundancy** and **pipeline bubbles** in Hybrid Parallelism to minimize throughput drop from failures



(a) Adaptive Pipelining

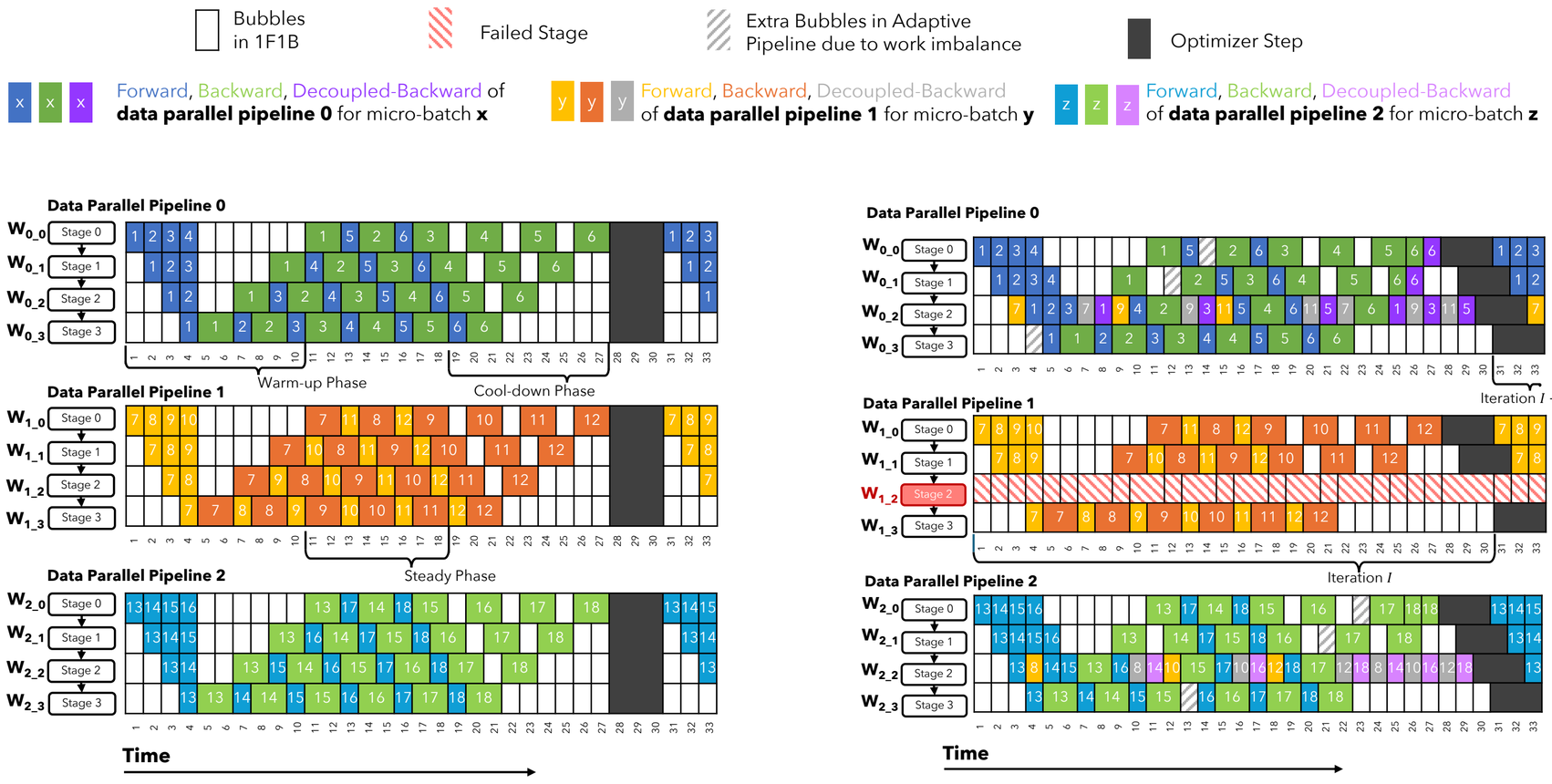




(c) Staggered Optimizer

Upon failure, **re-route computation** from failed worker to its functional data-parallel peers.

Working Around Failures

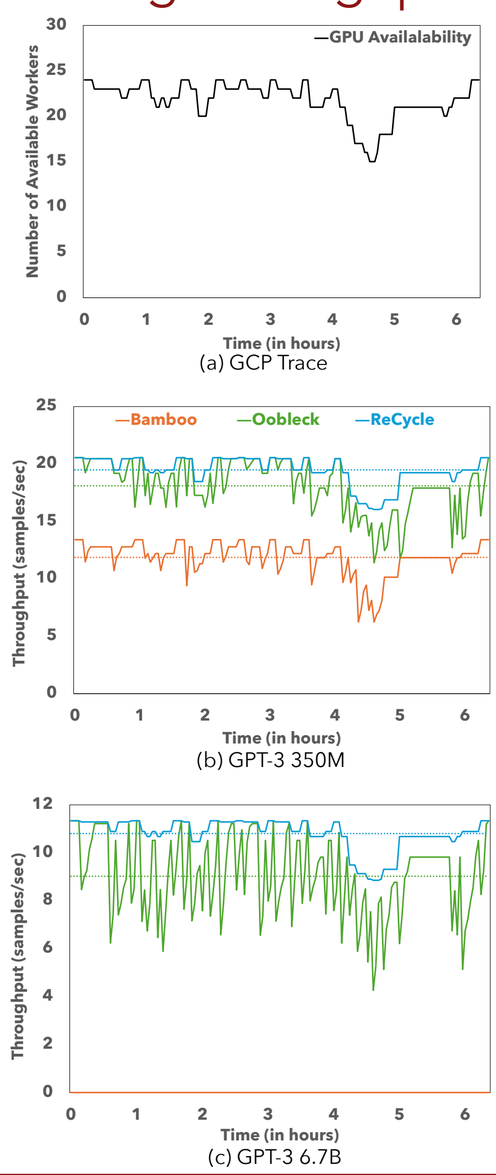


(A) Fault Free 1F1B Schedule

(B) ReCycle's Adaptive Schedule when W_{1_2} fails

Hybrid-parallel training with 3 data-parallel pipelines, 4 pipeline stages, and 18 micro-batches per iteration.

Training Throughput



Key Takeaway

- ✓ Stall-Free Fast Recovery from Failures
- ✓ No Impact on Model Accuracy from Failures
- ✓ Ensures High Training Throughput in presence and absence of Failures

Swapnil Gandhi, Mark Zhao, Athinagoras Skiadopoulos, and Christos Kozyrakis. 2024. ReCycle: Resilient Training of Large DNNs using Pipeline Adaptation. In ACM SIGOPS 30th Symposium on Operating Systems Principles (SOSP '24), November 4–6, 2024, Austin, TX, USA. ACM, New York, NY, USA, 18 pages. https://doi.org/10. 1145/3694715.3695960

Stanford University