Hippie: A Data-Paralleled Pipeline Approach to Improve Memory-Efficiency and Scalability for Large DNN Training

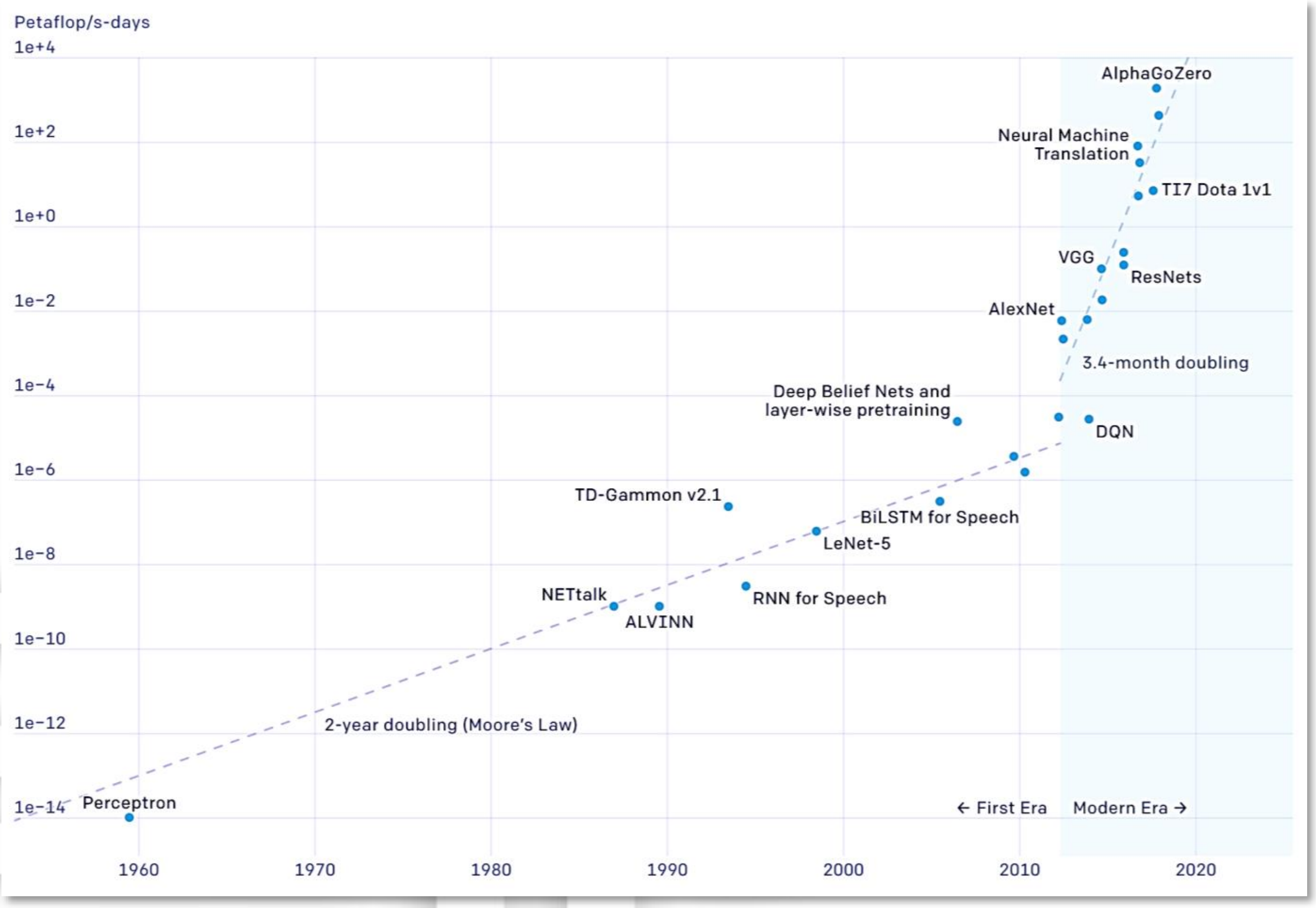
Xiangyu Ye, Zhiquan Lai, Shengwei Li, Lei Cai, Ding Sun, Linbo Qiao, Dongsheng Li

National Key Laboratory of Parallel and Distributed Processing

National University of Defence Technology

Contents

DNN (Deep Neural Network) models continue to grow



https://openai.com/blog/ai-and-compute

- Challenges of large DNN training
  - Memory limitation
    - Increase of model parameters
    - Increase of training data
  - Low scalability

Motivation

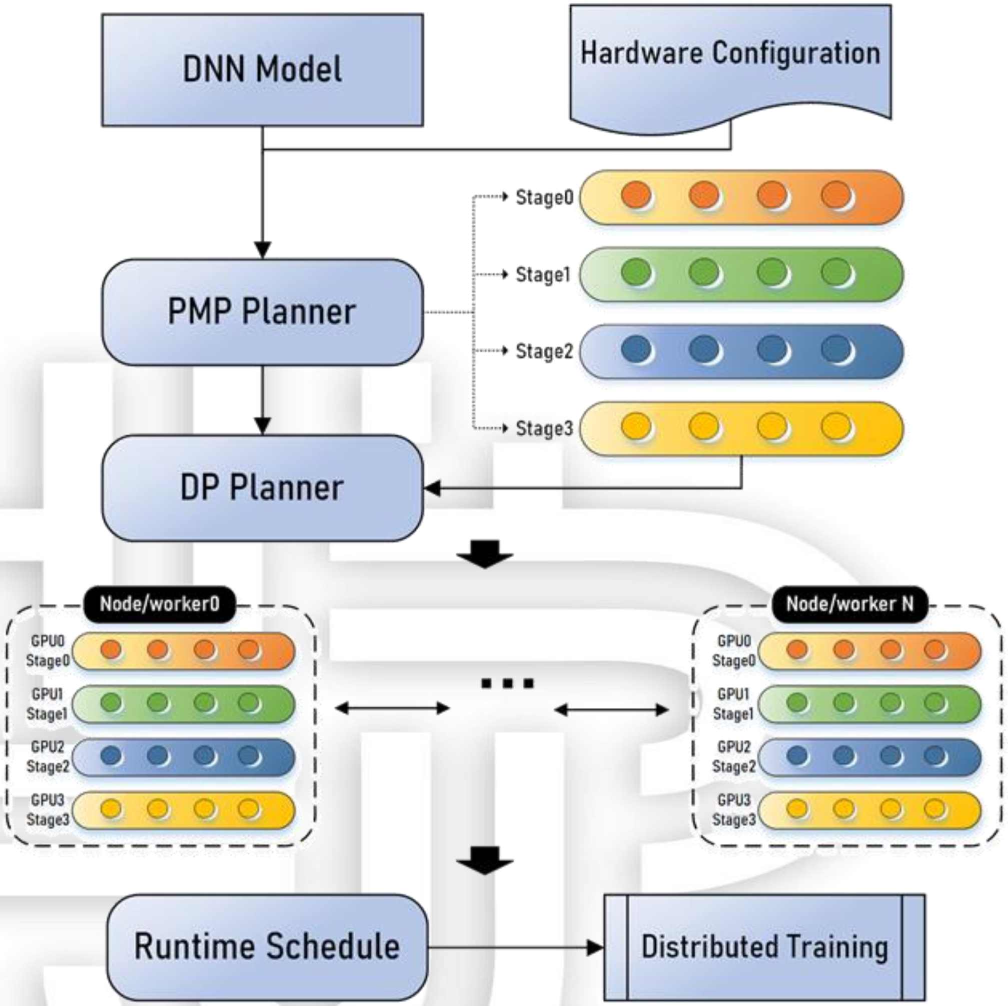
- **Two challenges of parallelizing DNN Training :**
  - High scalability
  - Low memory overhead

- **Define a new index to measure the efficiency:**
  - Memory Efficiency(ME) = $\frac{Scalability * Throughput}{Memory}$

Contents

- Motivation & Background
- **The Hippie approach**
  - **Overview**
  - Communication Schedule
  - Last-stage Schedule
  - Pipeline Plan
- Evaluation
- Conclusion

Overview

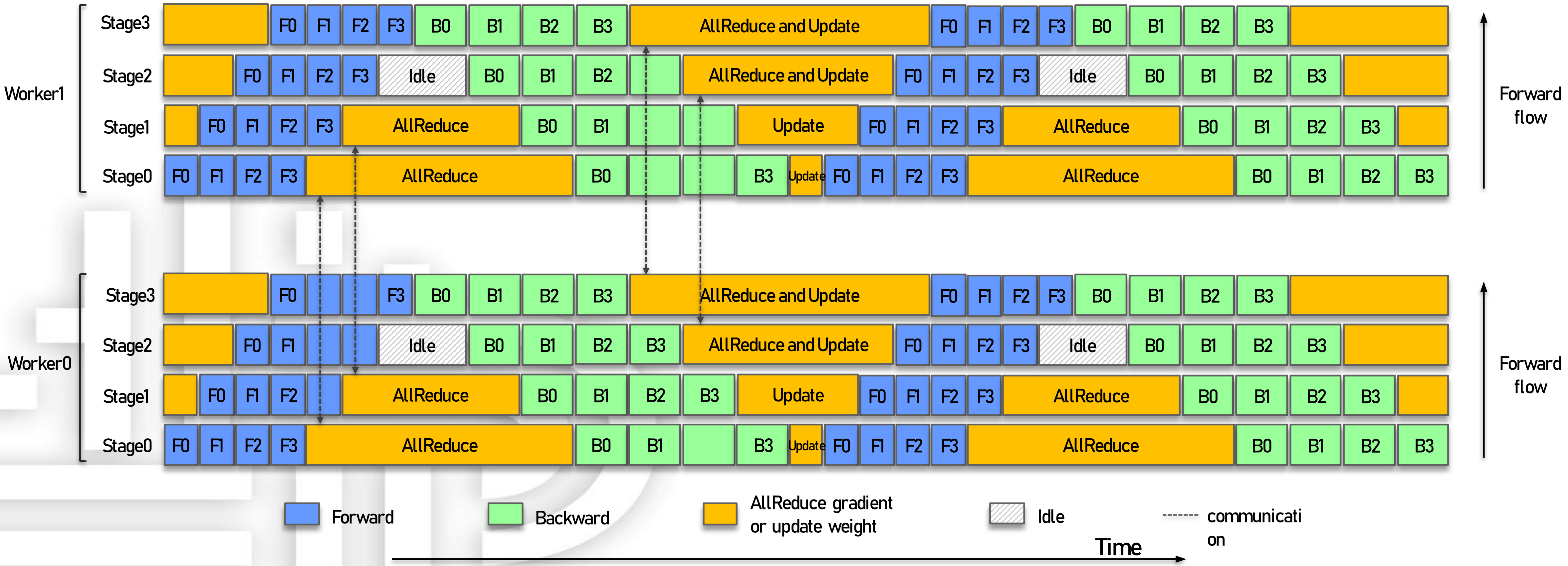


- **Improve scalability**
  - Communication Schedule
- **Reduce memory overhead**
  - Last-stage Schedule
  - Pipeline Planner

Contents

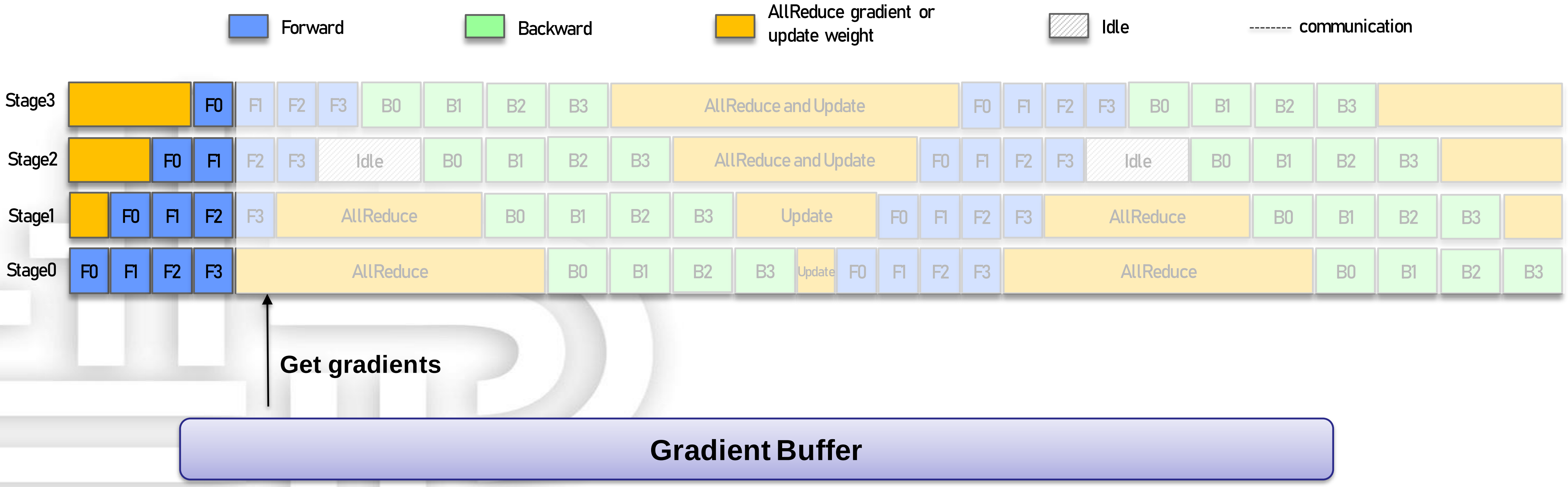
- Motivation & Background
- **The Hippie approach**
  - Overview
  - **Communication Schedule**
  - Last-stage Schedule
  - Pipeline Plan
- Evaluation
- Conclusion

Communication Schedule



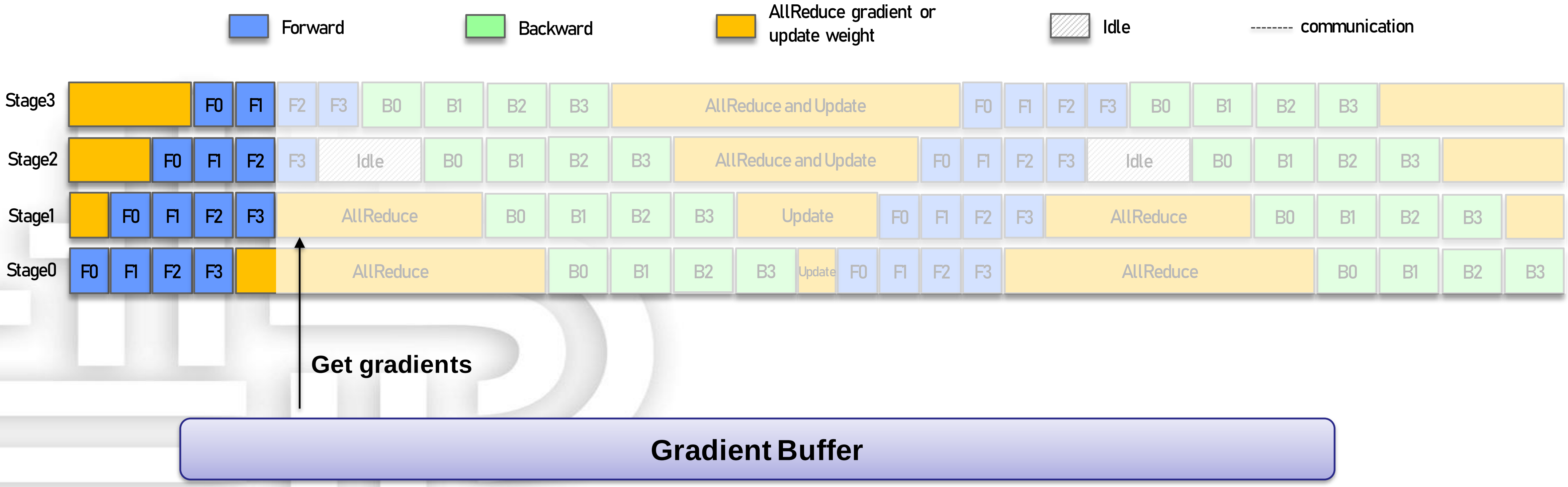
Communication Schedule

- **Stage0 starts to perform AllReduce:**



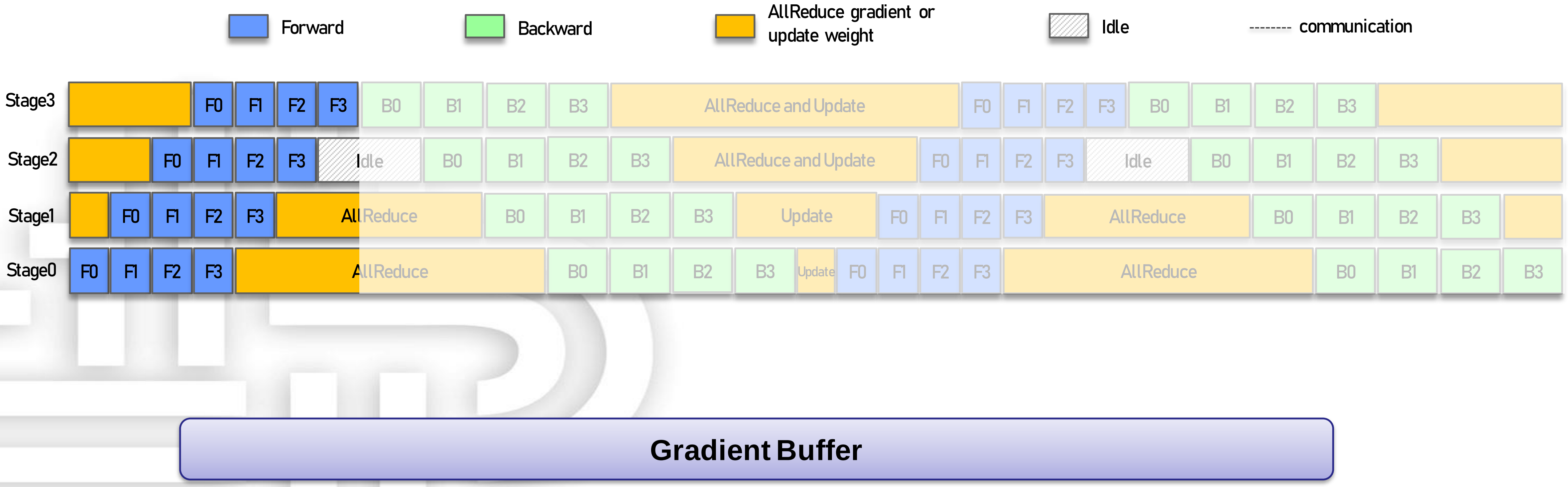
Communication Schedule

- **Stage1 starts to perform AllReduce:**



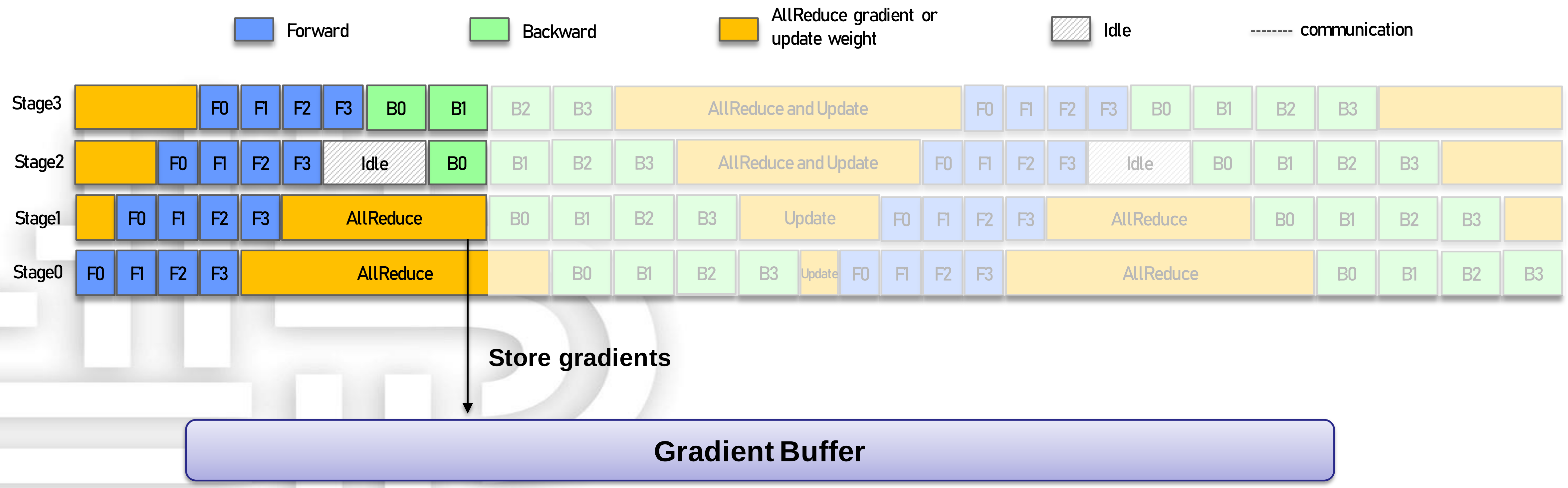
Communication Schedule

- **Pipeline starts to perform backward:**



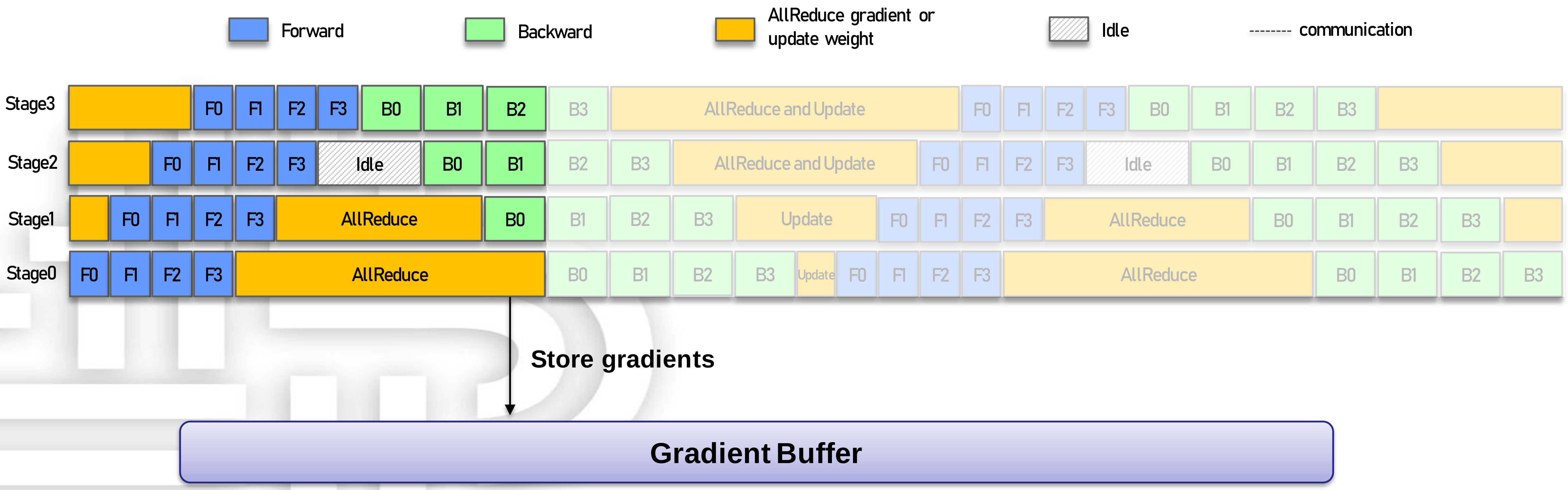
Communication Schedule

- **Stage1 ends the AllReduce:**



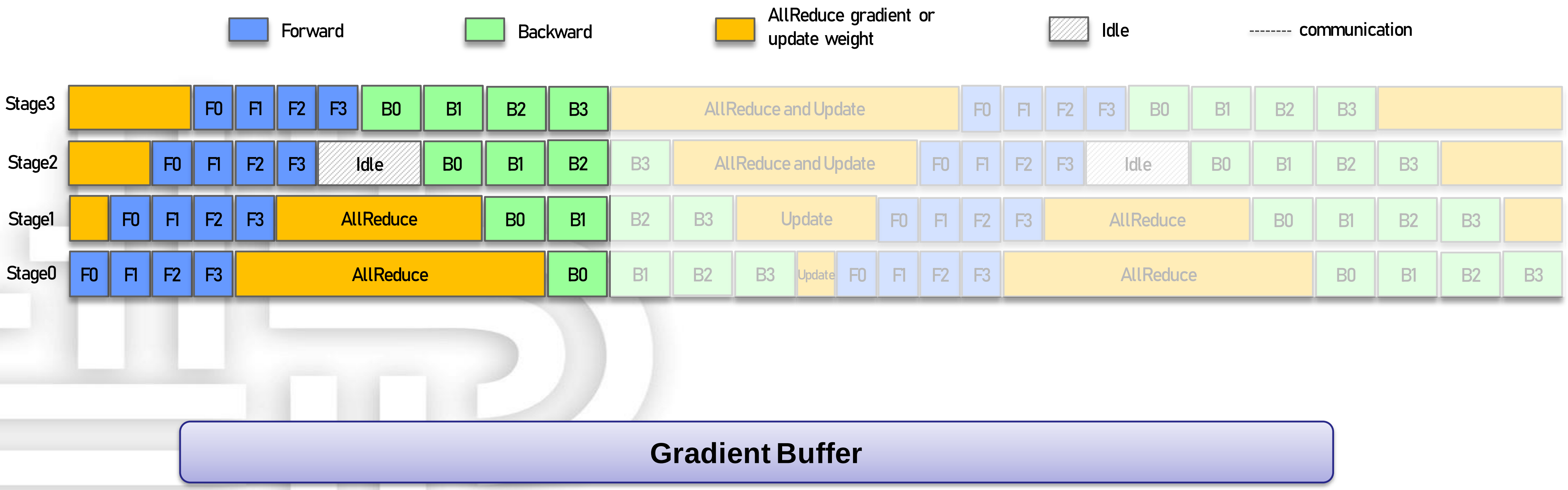
Communication Schedule

- **Stage0 ends the AllReduce:**



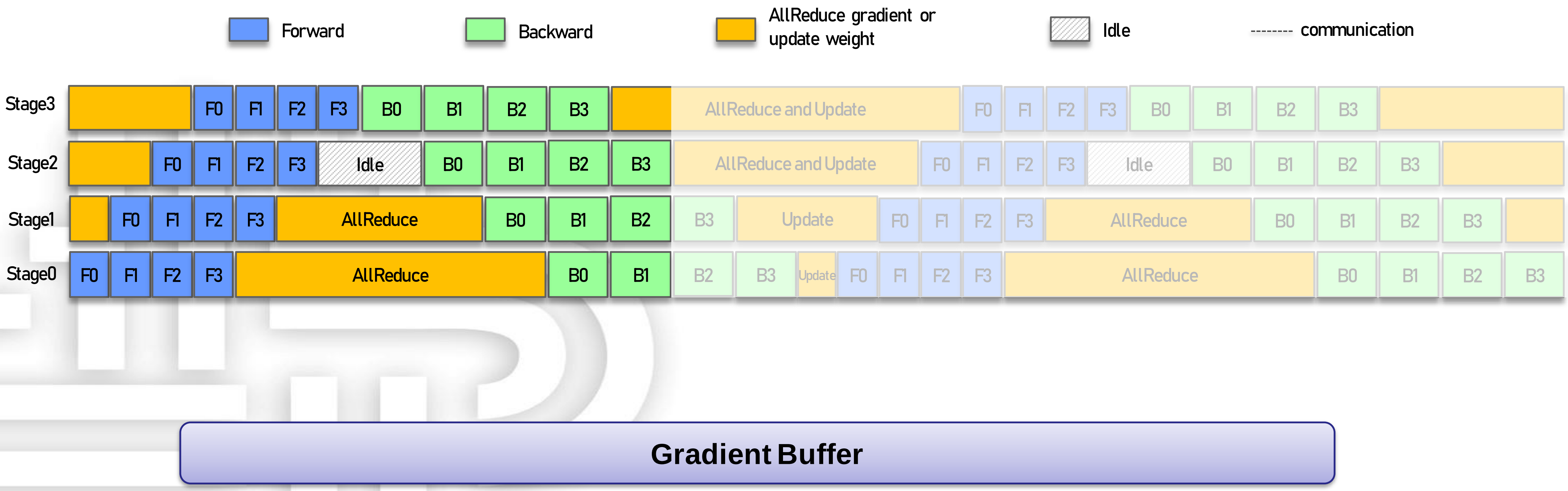
Communication Schedule

- **Stage3 starts to perform AllReduce and update:**



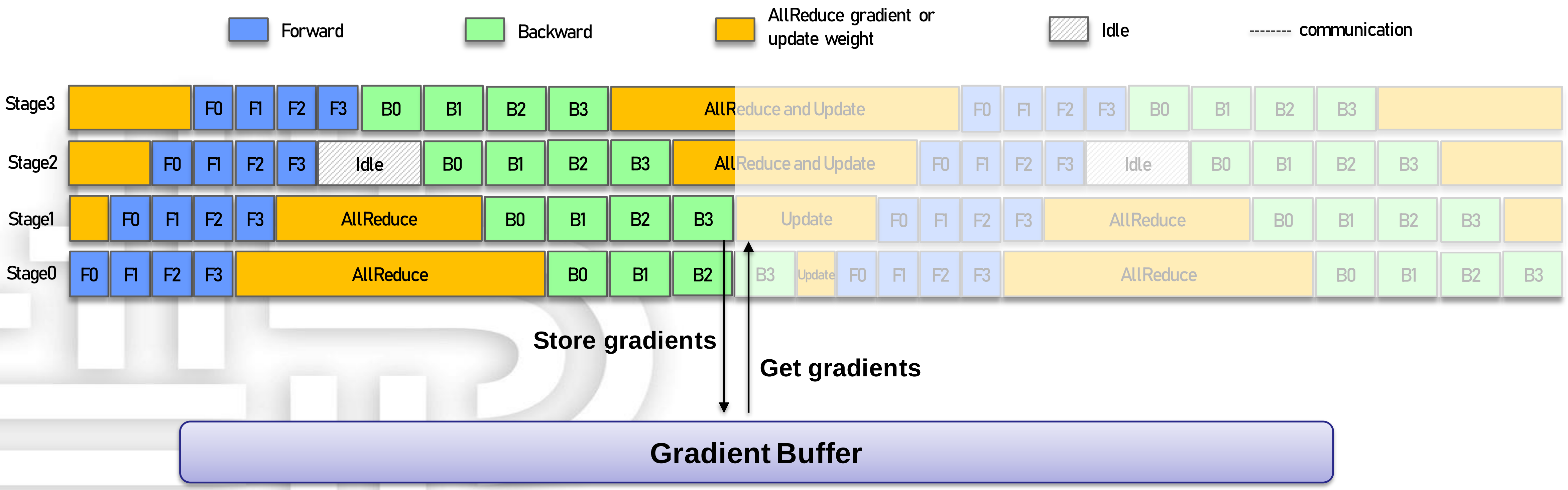
Communication Schedule

- **Stage2 starts to perform AllReduce and update:**



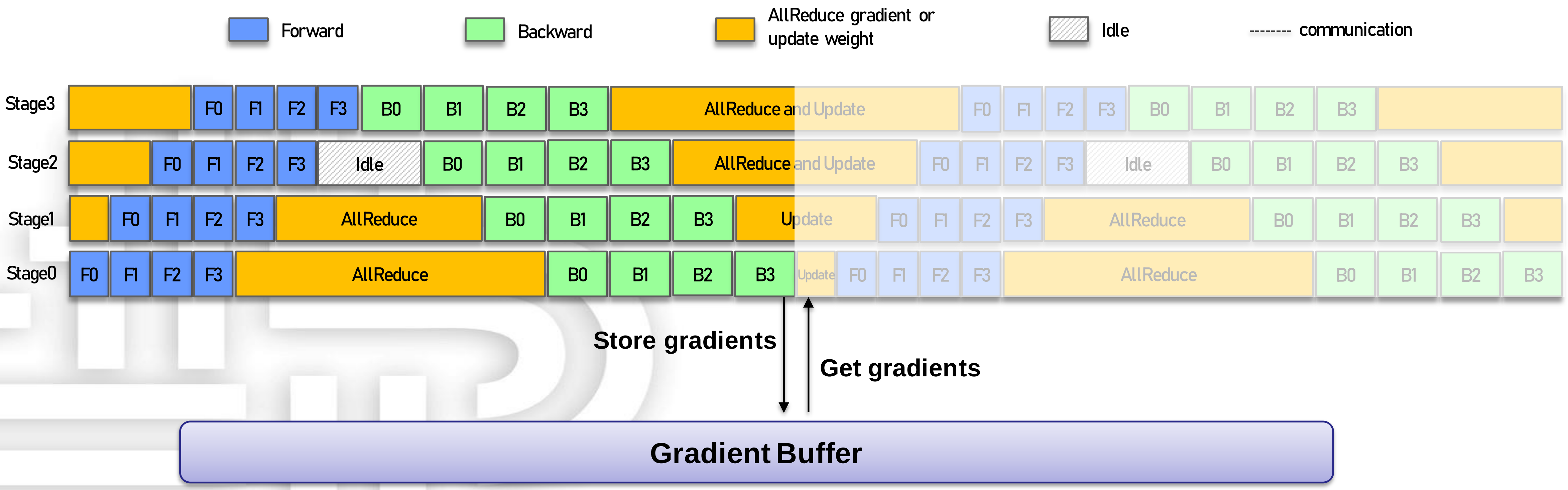
Communication Schedule

- **Stage1 starts to perform update:**



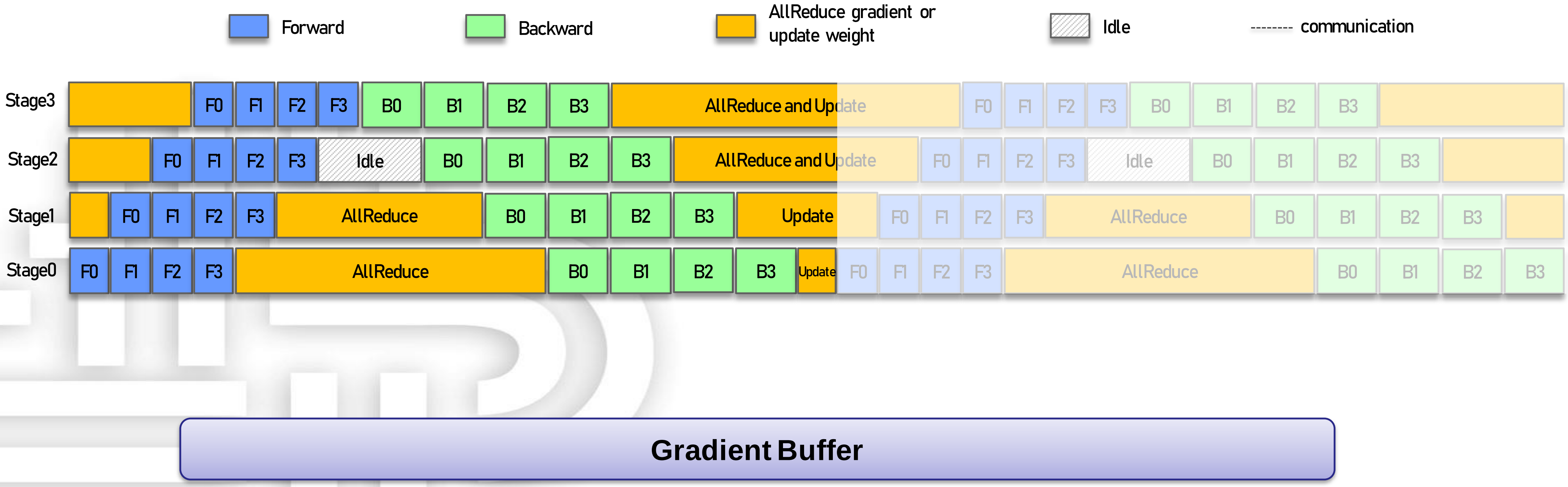
Communication Schedule

- **Stage1 starts to perform update:**



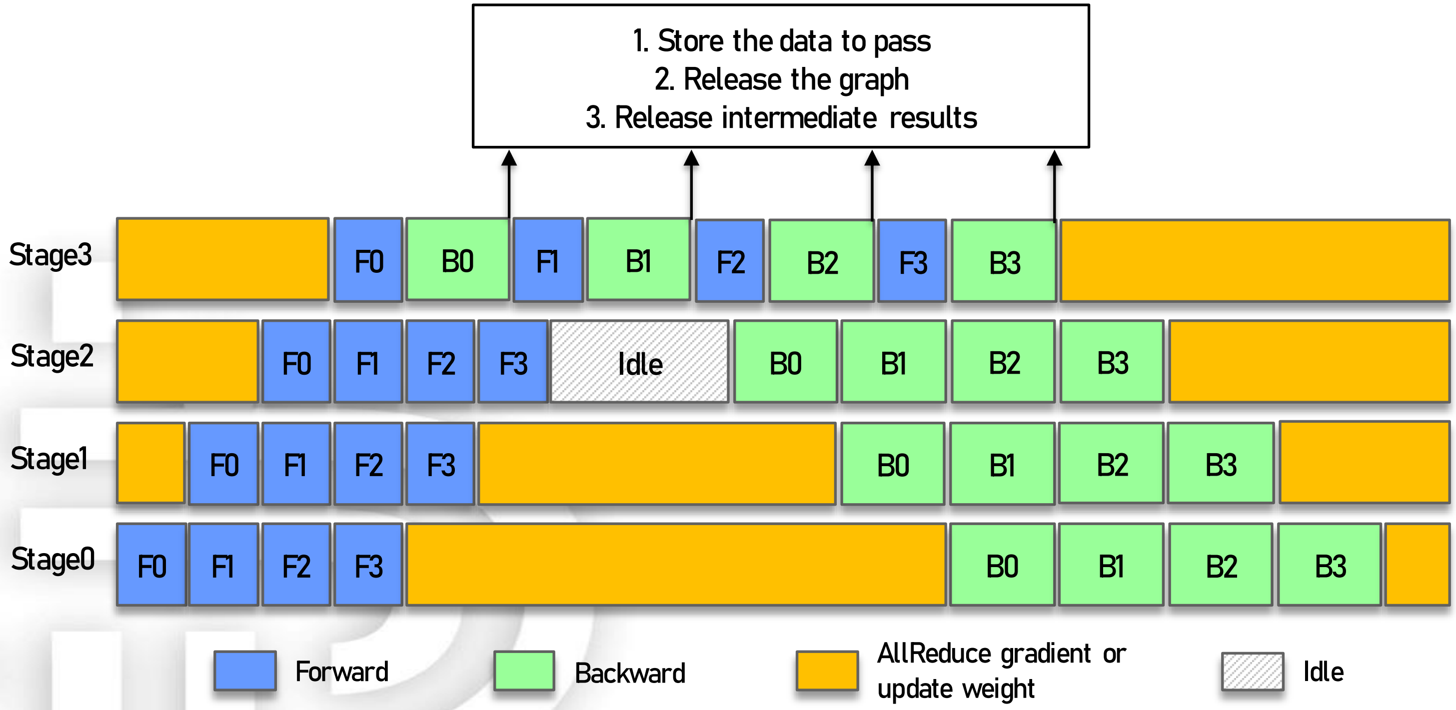
Communication Schedule

- **Pipeline starts to perform next step:**



Contents

Last-stage Schedule



Pipeline Planner

- **Aim:**
  - Generate an efficient pipeline

- **Optimization goal:**
  - Memory efficiency(ME)

- **Process:**
  - Partition the model
  - Select specific layers to apply re-computation

Contents

- Background & Motivation
- The Hippie approach
- **Evaluation**
  - **Experimental setup**
  - Memory efficiency
  - Scalability
  - Convergence
- Conclusion

Experimental setup

- **Models and datasets**

Model	# of Params	Dataset	Target Accuracy
GNMT-8	191M	WMT16 EN-De	24 BLEU
GNMT-16	290M	WMT16 EN-De	24 BLEU
VGG-16	138M	ImageNet	70% top-1
AmoebaNet-18	318M	ImageNet	70% top-1

Experimental setup

- **Training approaches**

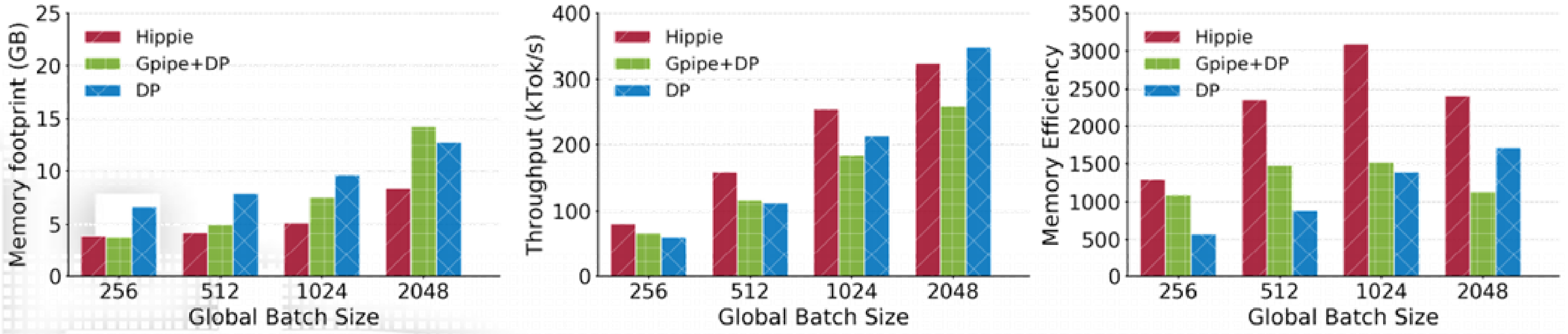
Hippie	The Hippie with four stages and with two stages within single nodes performs better as efficiency will be reduced greatly by the cross-node pipeline.
Gpipe+DP	We implement a training process that integrates Gpipe and DP, without applying recomputation and hiding communication.
DP	Data parallelism with intra-iteration computation communication overlap, which is one of the most efficient distributed training approaches under the PyTorch framework.
MP+DP	The method implemented to integrate MP and DP for training larger models.

Contents

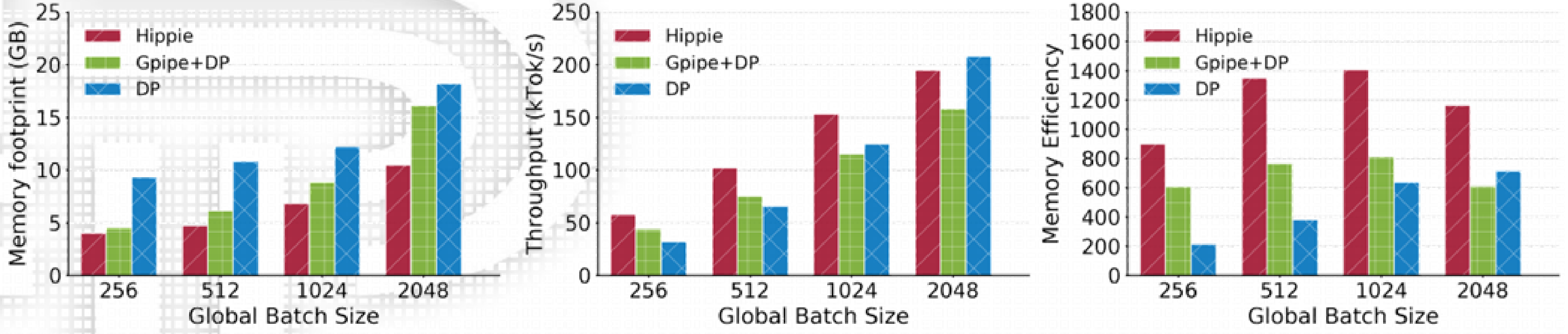
- Background & Motivation
- The Hippie approach
- **Evaluation**
  - Experimental setup
  - **Memory efficiency**
  - Scalability
  - Convergence
- Conclusion

Memory efficiency

- **Performance comparison using 16 GPUs**



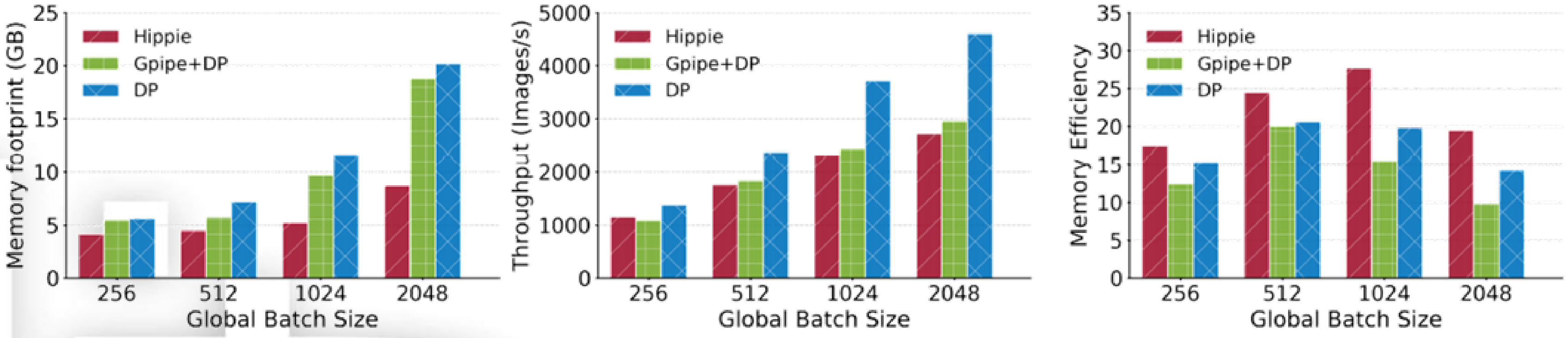
(a) GNMT-8



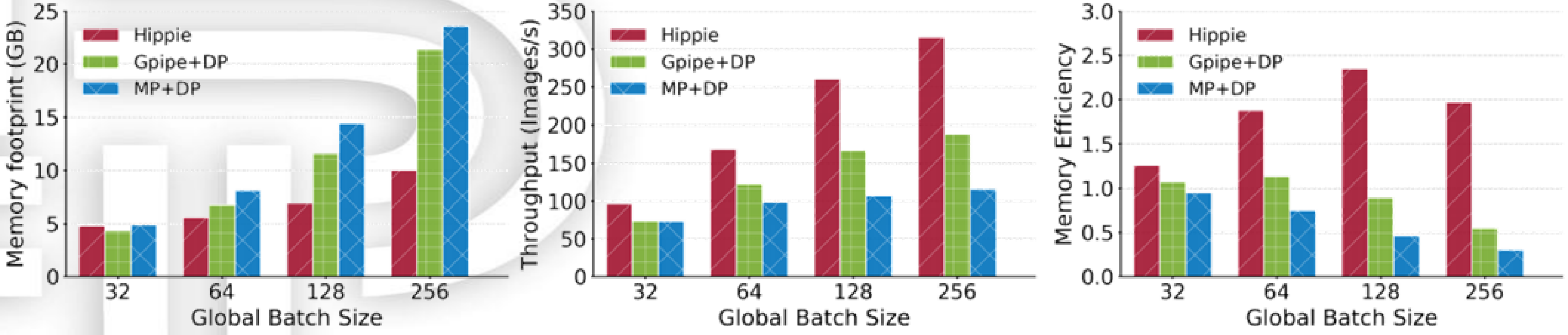
(b) GNMT-16

Memory efficiency

- **Performance comparison using 16 GPUs**



(c) VGG-16



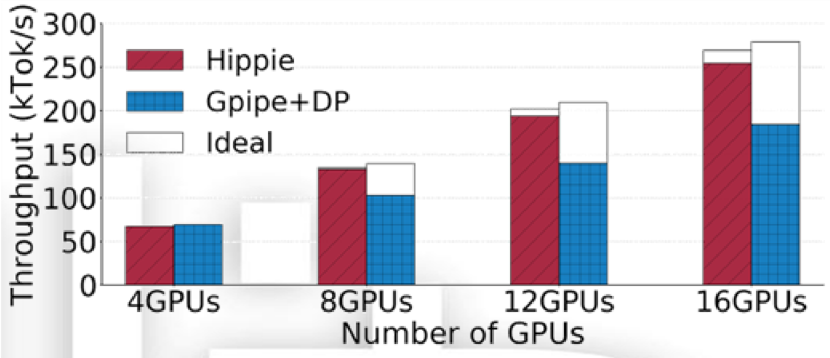
(d) AmoebaNet-18

Contents

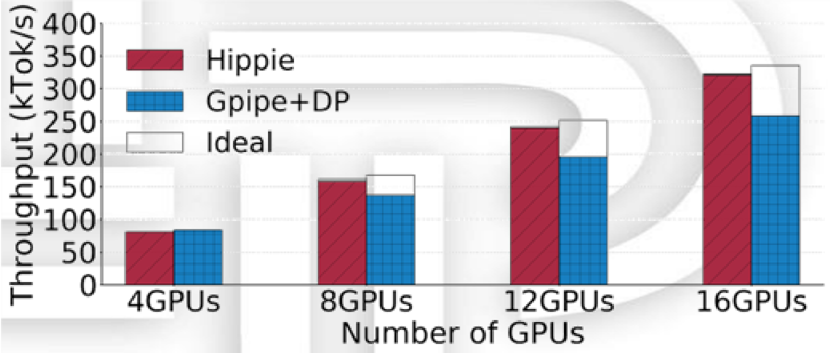
- Background & Motivation
- The Hippie approach
- **Evaluation**
  - Experimental setup
  - Memory efficiency
  - **Scalability**
  - **Convergence**
- Conclusion

Scalability

- **Multi-GPU scaling performance for GNMT-8**

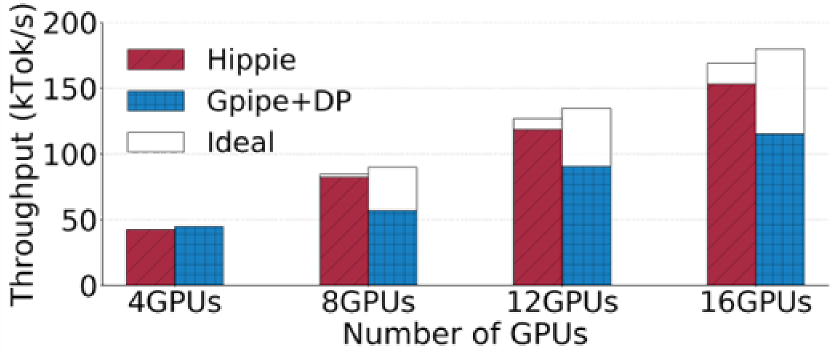


(a) 64 batch-size per GPU

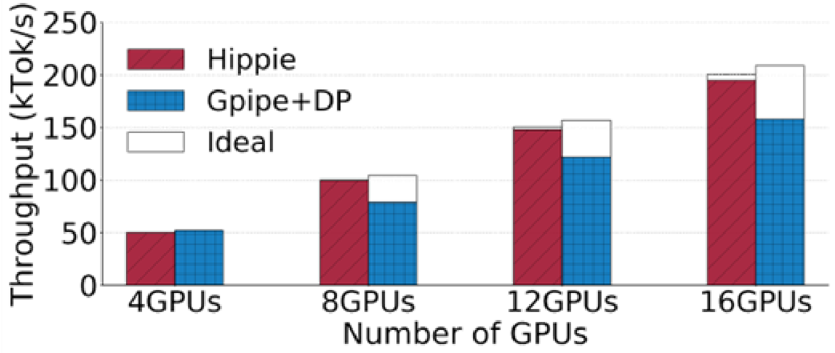


(b) 128 batch-size per GPU

- **Multi-GPU scaling performance for GNMT-16**



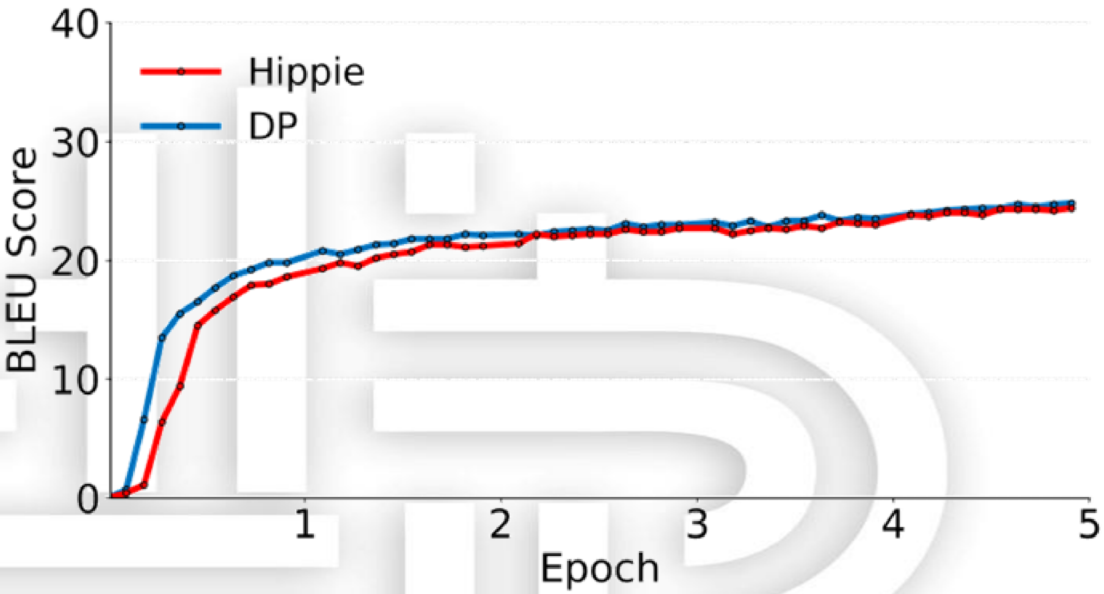
(a) 64 batch-size per GPU



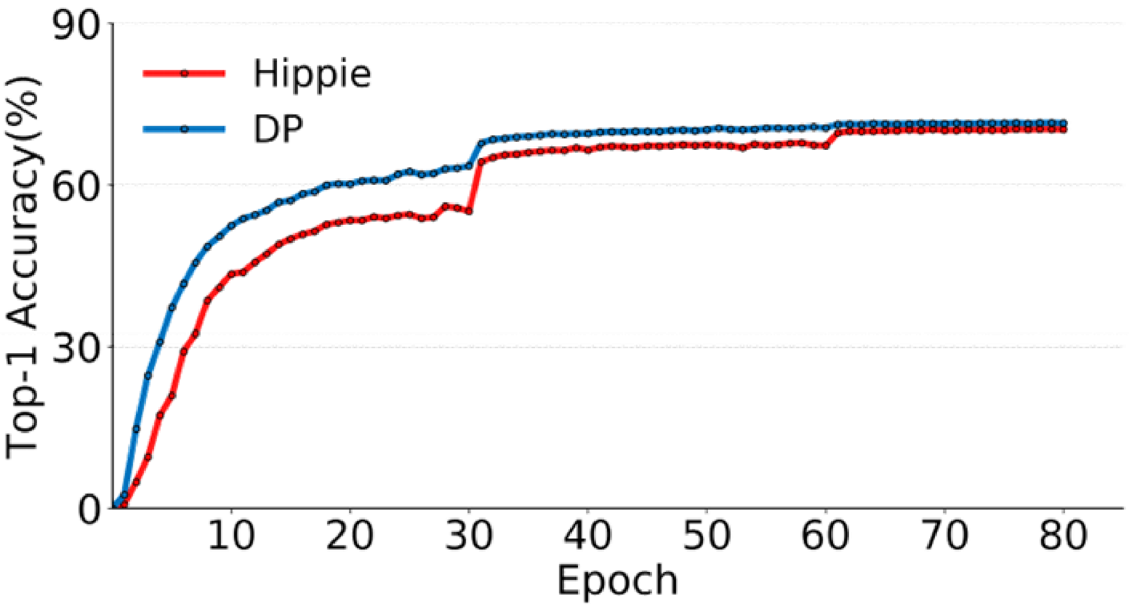
(b) 128 batch-size per GPU

Convergence

- **Accuracy vs. epoch using 16 GPUs**



(a) GNMT-16



(b) VGG-16

Contents

- Background & Motivation
- The Hippie approach
- Evaluation
  - Experimental setup
  - Memory efficiency
  - Scalability
  - Convergence
- **Conclusion**

Conclusion

- We present a distributed training framework which integrates pipelined model parallelism with data parallelism
- We introduces the ***Communication Schedule***, enabling Hippie to maintain **90%** scaling efficiency on a 16-GPU platform
- We introduce the ***Last-stage Schedule*** and ***Pipeline Planner*** to save **30%-60%** memory consumption
- Hippie outperforms DP by up to **4.18**× memory efficiency

Hippie: A Data-Paralleled Pipeline Approach to Improve Memory-Efficiency and Scalability for Large DNN Training

Xiangyu Ye, Zhiquan Lai, Shengwei Li, Lei Cai, Ding Sun, Linbo Qiao, Dongsheng Li

Thank you！

National Key Laboratory of Parallel and Distributed Processing

National University of Defence Technology