Prague: High Performance Heterogeneity-Aware Asynchronous Decentralized Training

•••

Qinyi Luo, Jiaao He, Youwei Zhuo, Xuehai Qian
University of Southern California and Tsinghua University

The Problem

Data Parallelism is a very useful procedure for training DNNs to convergence faster but there are limits to the current state of the art methods.

“Stragglers” (the slowest workers) are at the heart of the issue.

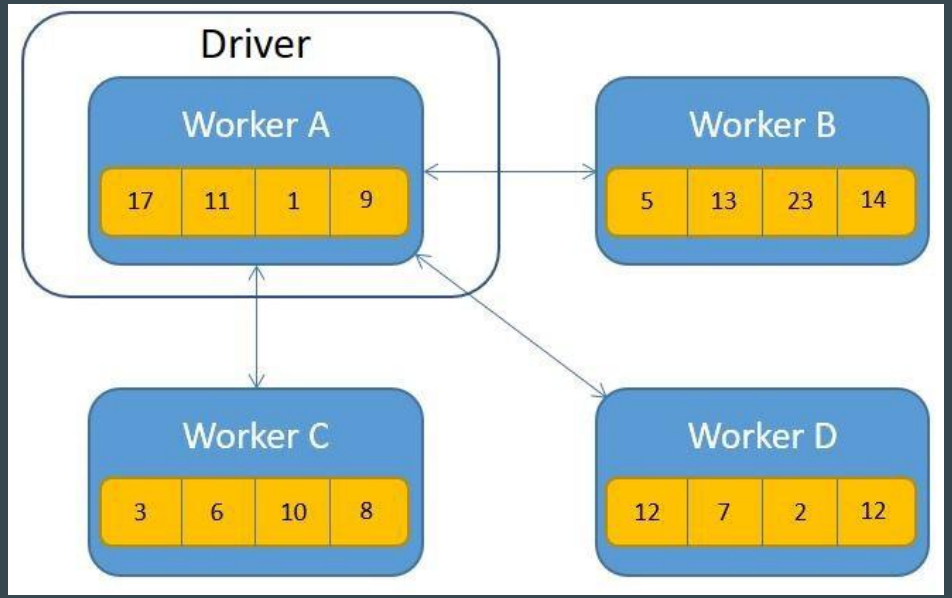
The baseline method is the outdated Parameter Server method where a single centralized server handles all worker communication and synchronization messages.

The all-reduce method works well when all workers / networks perform similarly but is fundamentally limited to the speed of the slowest worker (the straggler).

The AD-PSGD method handles heterogeneous environments and provides significant speed-up over baseline, but falls far short of the speed of All-Reduce in homogeneous environments.

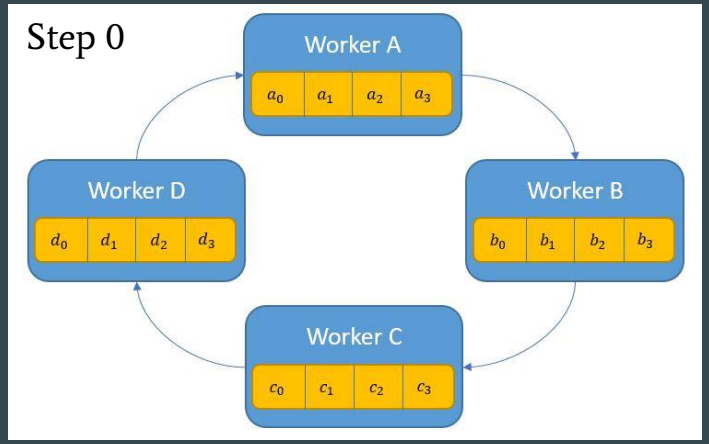
Parameter Server Algorithm

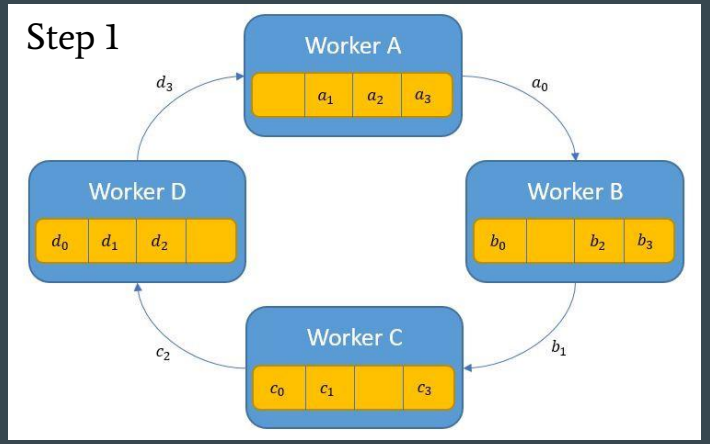
- Reference algorithm.
- Single Worker handles all communication and synchronizes with all other workers.
- Very high communication overhead.

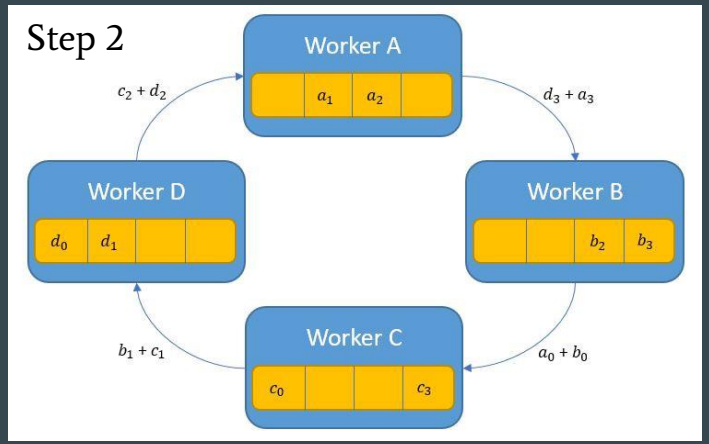


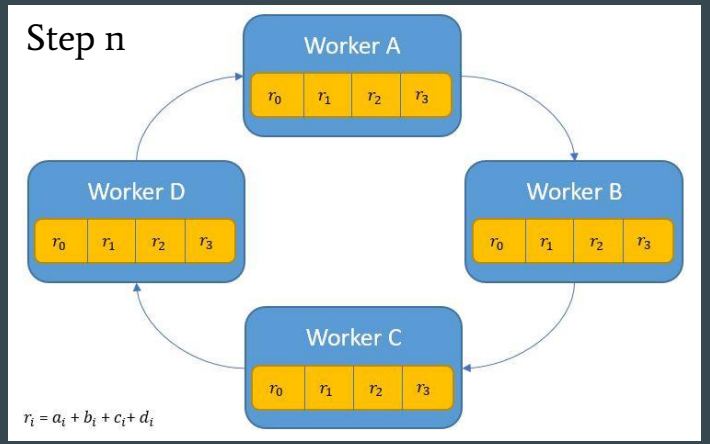
All-reduce Algorithm (Ring All-reduce)

- Best suited to homogeneous environments.
- Computation is evenly distributed across workers.
- Each step is finished in the time it takes the slowest worker to complete.





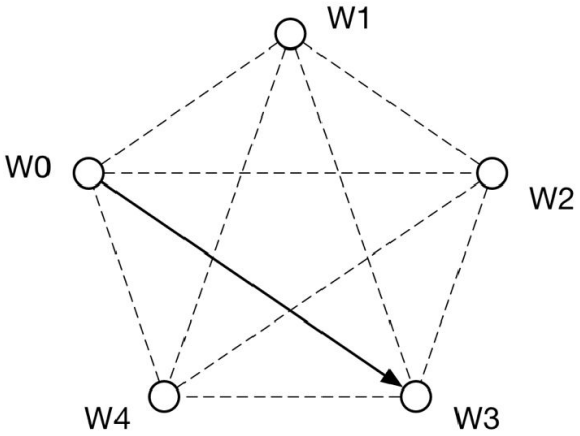




AD-PSGD Algorithm

Asynchronous Decentralized Parallel Stochastic Gradient Descent

- After completing SGD each node randomly selects another node and synchronizes with just that node.
- An adjustment to the original algorithm splits the graph equally into active and passive worker groups. Only the active group initiates synchronization. This reduced the chances of deadlock.

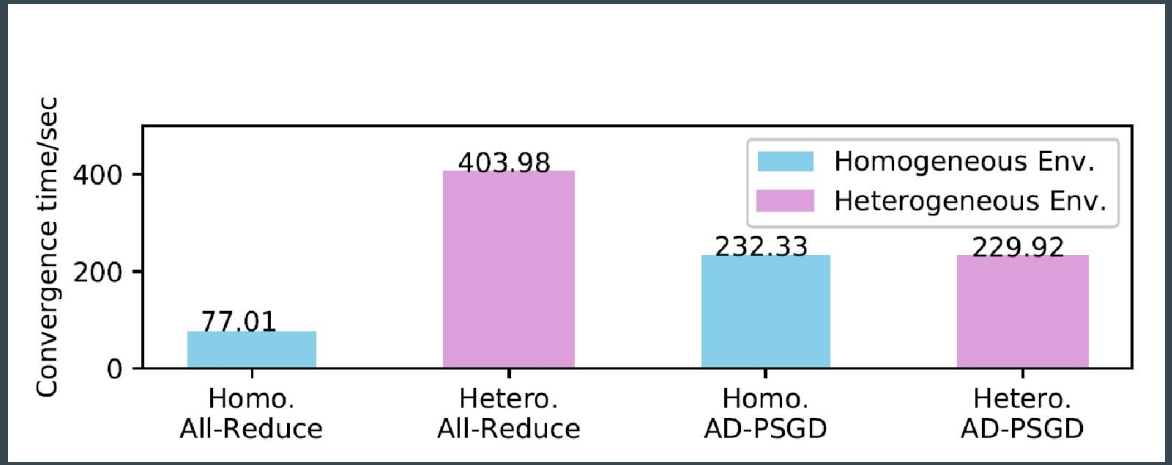


$$\begin{bmatrix} 0.5 & 0 & 0 & 0.5 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0.5 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 4. Synchronization in AD-PSGD

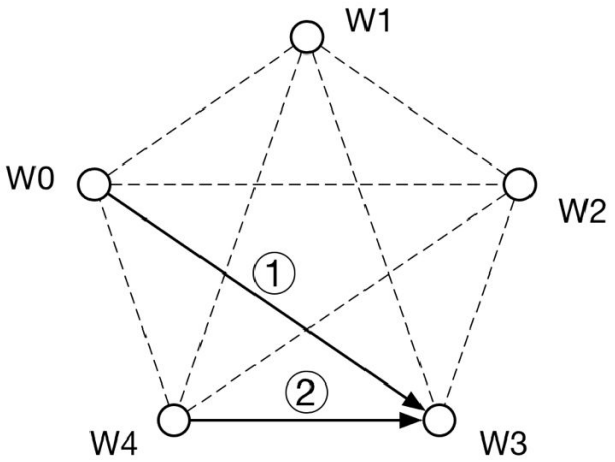
All-Reduce vs. AD-PSGD

- All-Reduce performs well in homogeneous environments but not heterogeneous.
- AD-PSGD has much higher communication costs and but is resilient to heterogeneous environments



Inefficiency in AD-PSGD

- In cases where multiple active nodes select the same passive node to synchronize with the operations are done sequentially rather than as a group.

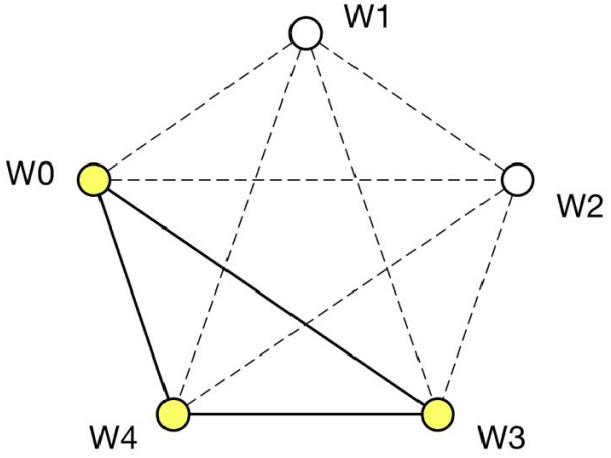


$$\begin{bmatrix} 0.5 & 0 & 0 & 0.25 & 0.25 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0.5 & 0 & 0 & 0.25 & 0.25 \\ 0 & 0 & 0 & 0.5 & 0.5 \end{bmatrix}$$

Figure 5. Conflict Between Two Pairs of Workers

Partial All-Reduce

- Apply All-Reduce at the group level
- Generate groups to synchronize with a random or “smart” method.
- Get the best of All-Reduce and remain resilient to “stragglers”.



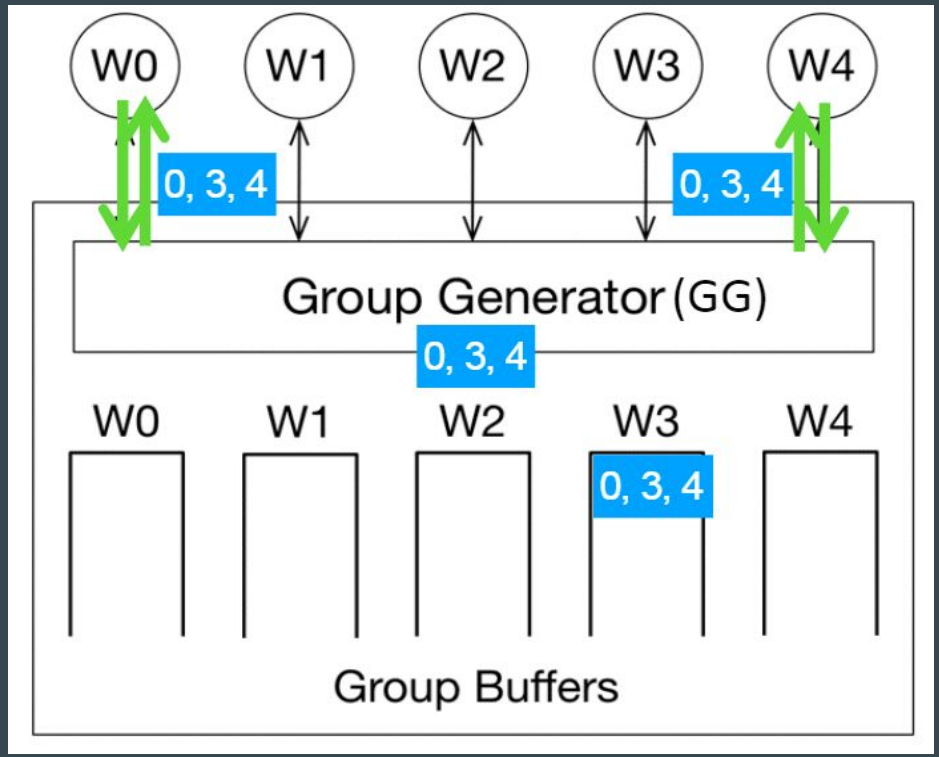
$$\begin{bmatrix} 1/3 & 0 & 0 & 1/3 & 1/3 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1/3 & 0 & 0 & 1/3 & 1/3 \\ 1/3 & 0 & 0 & 1/3 & 1/3 \end{bmatrix}$$

Figure 6. Synchronization with Partial All-Reduce

Prague System Design

Worker:

1. Compute & apply gradients locally to update model parameters.
2. Sync group information from the GG.
3. Perform P-Reduce collectively with other group members.

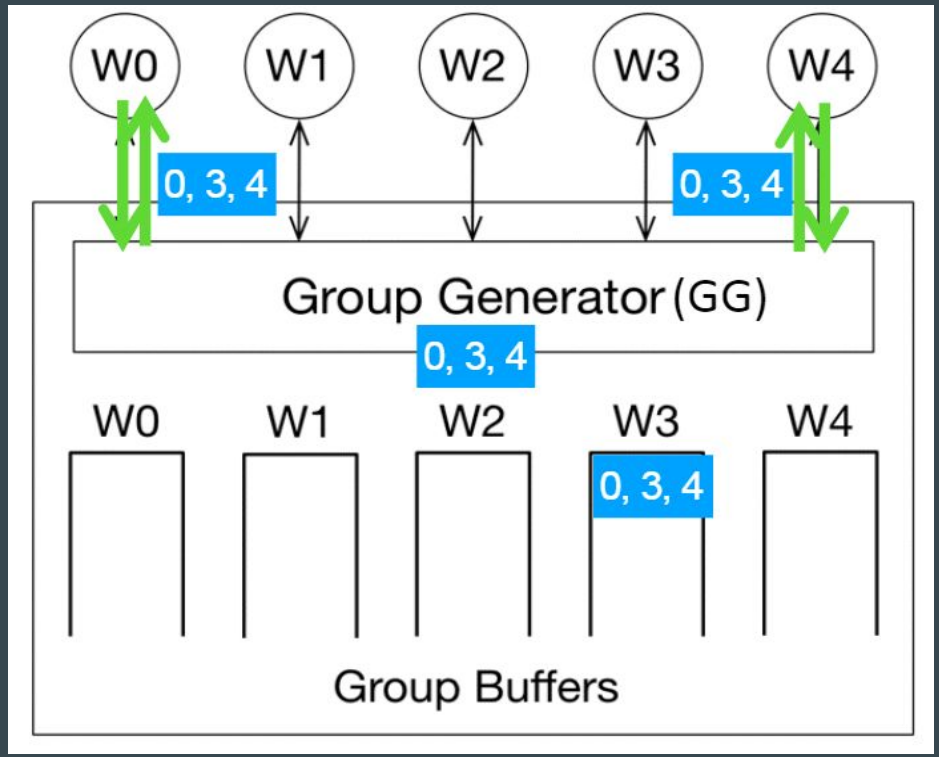


Prague System Design

Group Generator (GG):

1. Return the associated group associated with the requester (if previously created).

2. For the first time request, generate a new group.

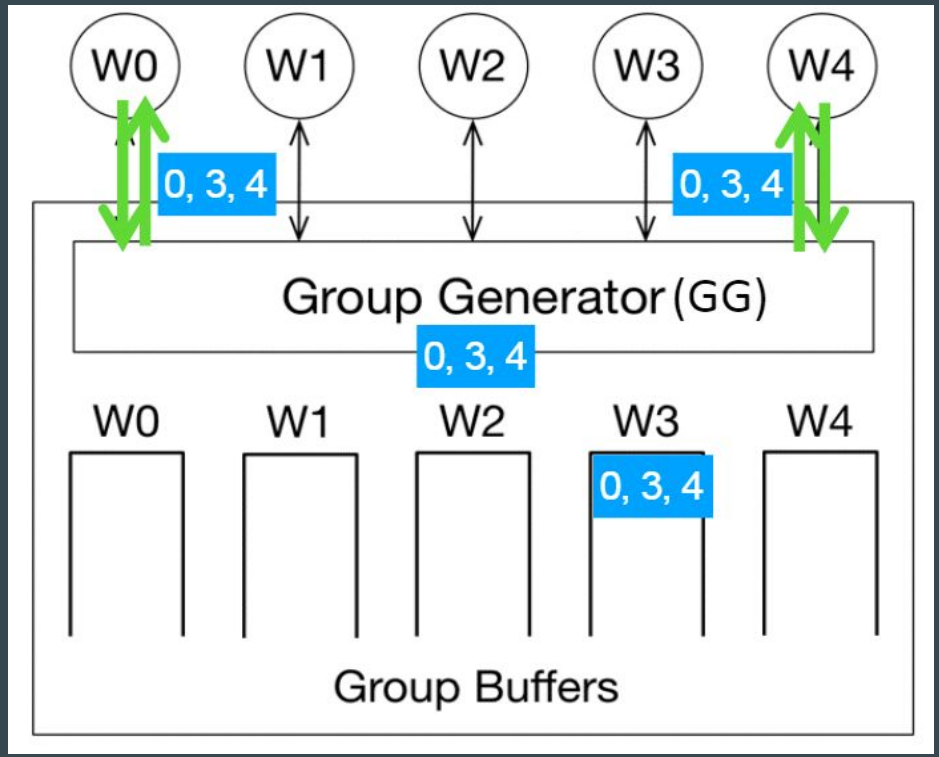


Prague System Design

Group Generator (GG):

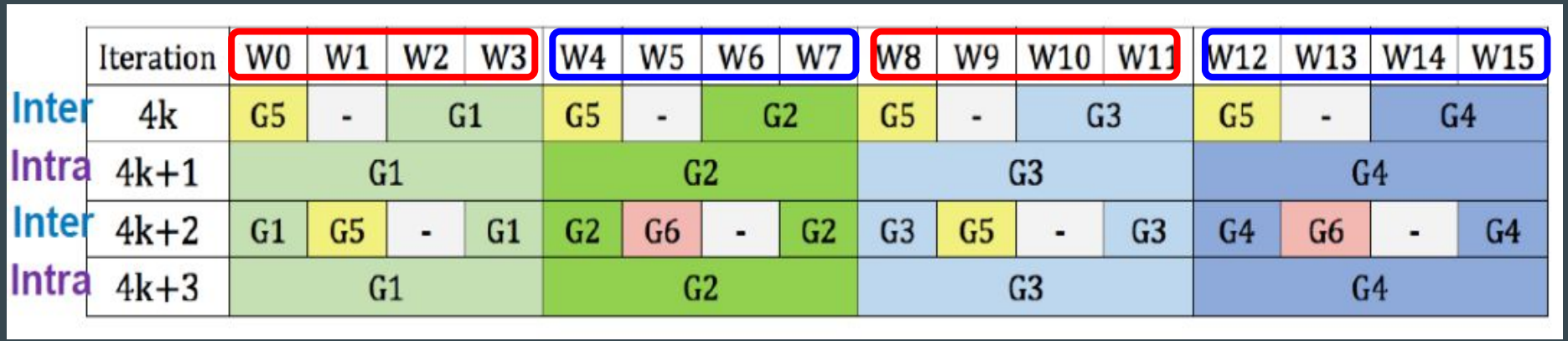
Challenge:

- group generator strategy, e.g., avoiding conflict among groups
- Straggler Problem



Conflict Avoidance

- Conflict-Free Grouping
  - Manually implemented by static GG (Fixed Schedule)
  - Cost of conflict-avoidance is cheaper.
  - Weaker resistance to stragglers.



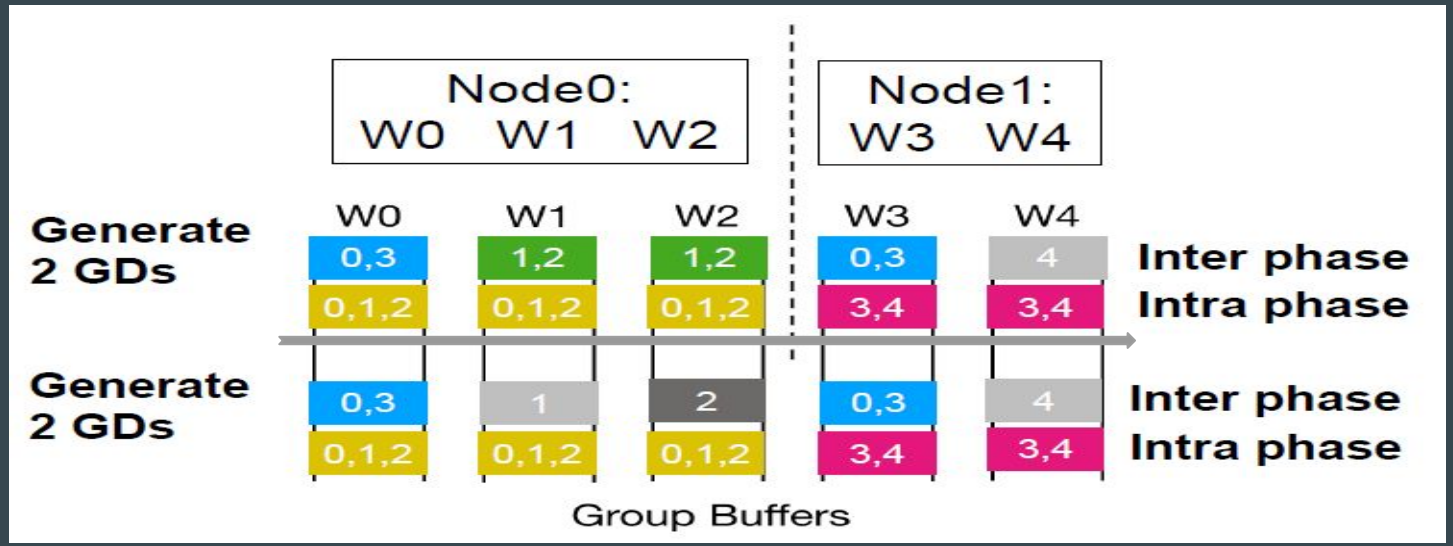
	Iteration	W0	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15
Inter	4k	G5	-	G1		G5	-	G2		G5	-	G3		G5	-	G4	
Intra	4k+1	G1				G2				G3				G4			
Inter	4k+2	G1	G5	-	G1	G2	G6	-	G2	G3	G5	-	G3	G4	G6	-	G4
Intra	4k+3	G1				G2				G3				G4			

Conflict Avoidance

- Randomized Schedule
  - Puts forward the idea of Group Division (GD)
  - Generates 1 random division at a time: computes group for all nodes simultaneously (bias-avoidance)

Conflict Avoidance

- Inter-Intra Synchronization
  - Alternates between inter-intra when generating groups.
  - Improved Performance



Tolerating Stragglers

- Applies threshold rule method
  - Avoids putting slow workers and fast workers in the same group.
  - Provides slow workers with leeway to ‘catch up’ with other workers.

GG keeps track of how many times a worker has requested a group (denoted by c_w for worker w)

A worker w can only be included in a group if

$$c_w > c_i - \Delta c$$

where worker i is the requester and Δc is the threshold

System Implementations

- Implemented as customized TensorFlow operator
    - P-Reduce uses NCCL as the backend & creates NCCL communicator through MPI
    - GG uses gRPC python package
- Supports 2 implementations of the Prague:
    - *Static GG*: group schedules are manually designed.
    - *Smart GG*: implements randomized schedules (incl randomized group division, inter-intra synchronization, & threshold rule.

System Implementations

- Implements two baseline algorithms
  - *All-Reduce*: Horovod with NCCL backend on TensorFlow
  - *AD-PSGD*: TensorFlow remote variable access

Experiments

- Models and datasets used
  - *Vision:* VGG-16 on CIFAR-10; Resnet-18, Resnet-50 and Resnet-200 on ImageNet
  - *NLP: Transformer on News-Commentary dataset*
- *Modeling Heterogeneity*
  - *Adding Artificial Slowdown:*
    - *each i.i.d. worker slows down with a probability of 1/(#workers)*

Results

1. Throughput
    - Modeled with and without slowdown

Without artificial slowdown

	All-Reduce	AD-PSGD	Static GG	Smart GG
Resnet-18	1	0.62	**1.10**	**1.10**
*Resnet-50	1	0.58	**1.04**	1.02
Resnet-200	**1**	0.52	0.96	0.98
Transformer	1	-	-	**4**

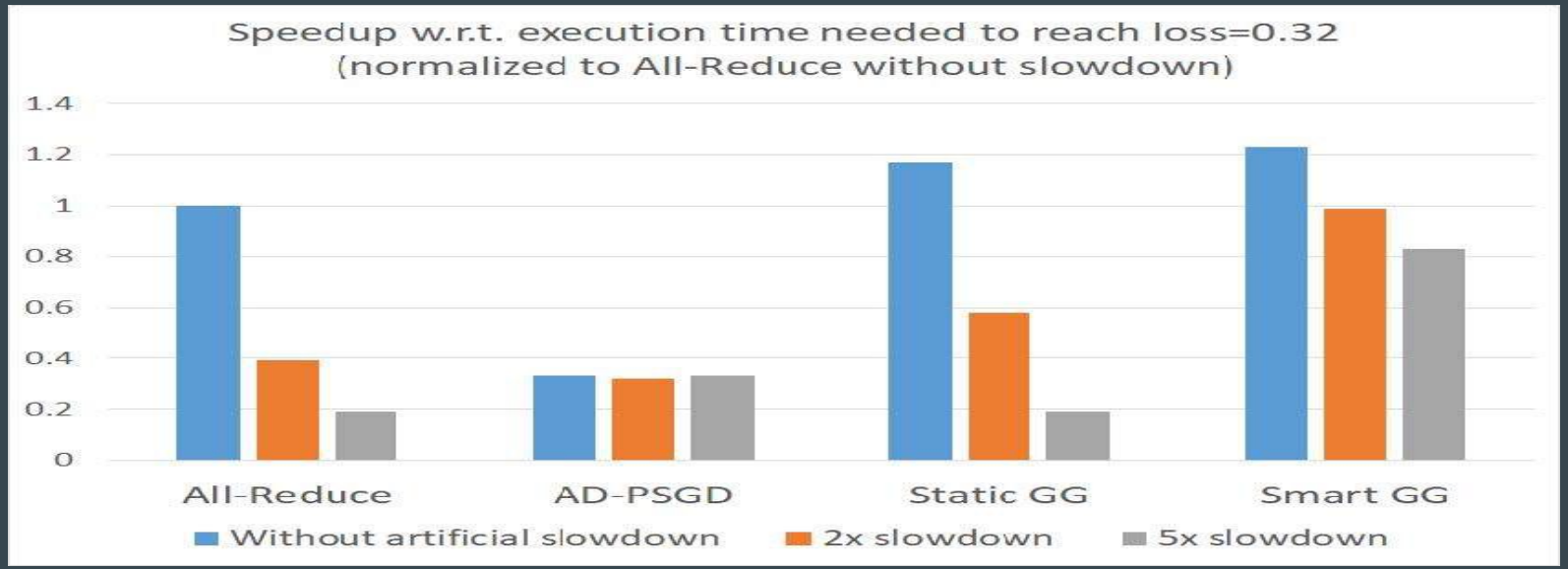
*Run on 32 GPUs

With 5X artificial slowdown

	All-Reduce	AD-PSGD	Smart GG
Resnet-18	0.24	0.28	**0.45**
Resnet-200	0.24	0.23	**0.45**

Results

2. Convergence
   - Time required to attain a desired loss (for VGG-16)



Results

3. Accuracy of the final model

	Prague	All-Reduce
Resnet-50 accuracy	74.16%	74.05%
Transformer BLEU score (5h, ref:27)	25.5	21

Conclusion

1. Proposed Prague as a high-performance heterogeneity-aware asynchronous decentralized training approach.

2. Introduced a novel communication primitive and randomized Partial All-Reduce (P-Reduce), to lower communication costs.

3. Designed smart group generation strategies to eliminate conflicts & tolerate stragglers.

4. Without loss in utility to All-Reduce in homogeneous environments, Prague displayed superior tolerance towards stragglers.

Limitations

1. Experiments have been performed with only a single slowdown rate, hence the speedup claims may not be consistent.

2. The final accuracy of both ResNet-50 and Transformer models is without the artificial slowdown. The claims that the proposed approach can achieve the same accuracy in a time-bounded fashion to P-Reduce may not be valid throughout.