Optimizing Resource Allocation in Pipeline Parallelism for Distributed DNN Training

Yubin Duan and Jie Wu

Dept. of Computer and Information Sciences

Temple University, USA

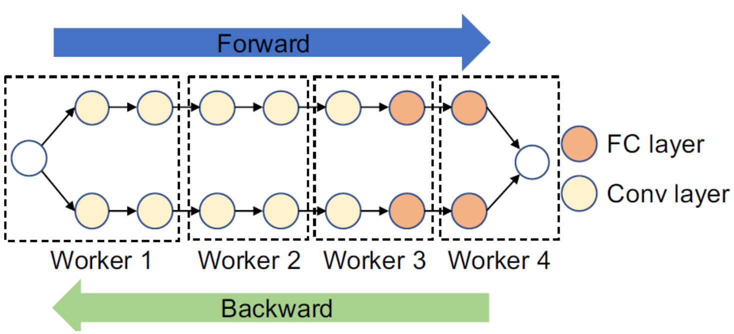
ICPADS 2022

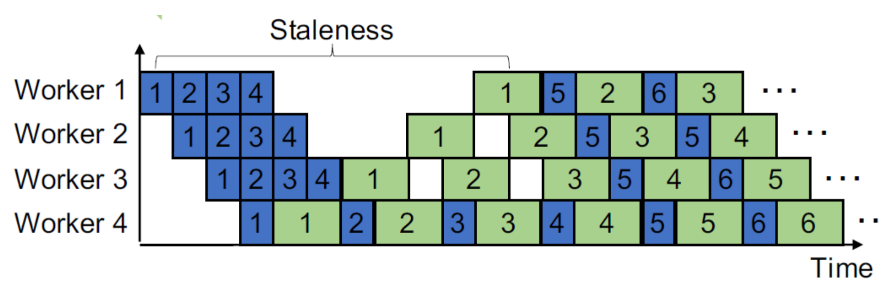
Outline

1. Introduction
2. Extended Pipeline Parallelism
   - Homogeneous Workers
   - Heterogeneous Workers
3. Experiment
4. Conclusions

1. Introduction

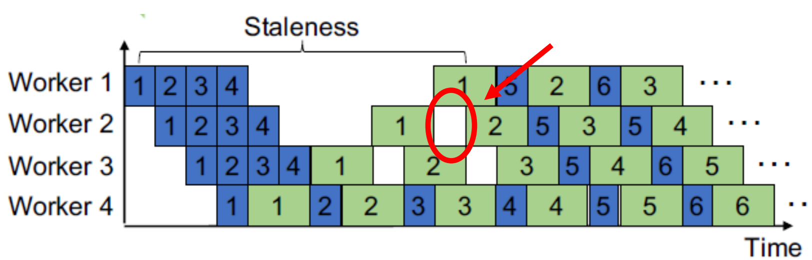
- Distributed DNN Training
  - Data Parallelism
    - Partition data and assign to multiple workers
    - Each worker node has parameters of the whole model
  - Model Parallelism
    - Partition models
  - **Pipeline Parallelism**
    - Data + Model parallelism





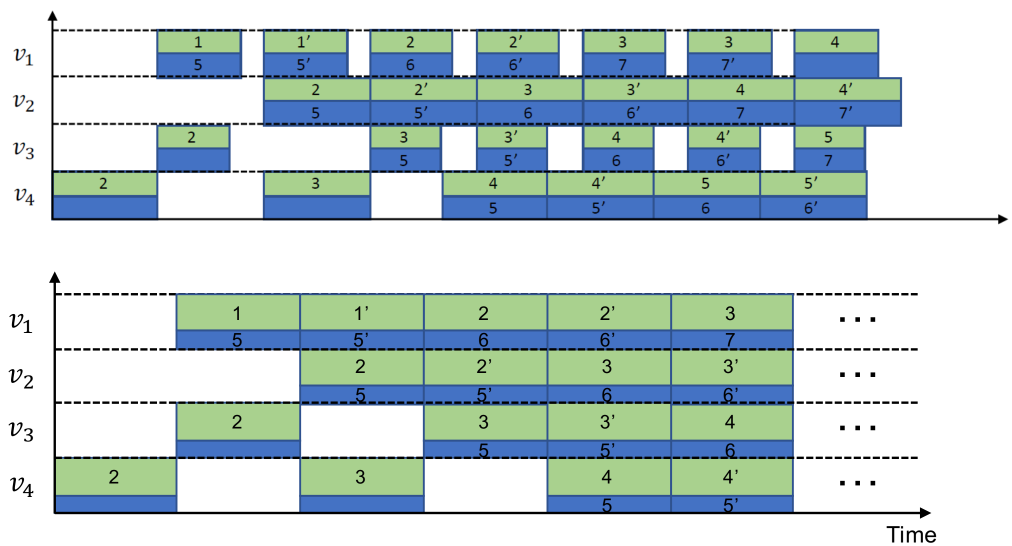
Motivation

- Each worker may have multiple types of computation resources
  - Resource types: CPU, GPU, FPGA, and ASIC
  - Objective: Minimize training duration
- Observation
  - Reduce resource idle time by adjusting the ratio of resources allocated to forward and backward pass



2. Extend Pipeline Parallelism

- Resource allocation pipeline parallelism
  - Multiple types of computation resources
  - Forward and backward operations
- Insights
  - Align forward and backward pass via resource allocation



Homogenous Workers

- Optimize resource allocation ratio to balance the duration of forward and backward operations

Theorem: The optimal resource allocation ratio $\beta_j = c/(c+1)$, if $f(p_i, r_j)/g(p_i, r_j) = c, \forall 1 \leq j \leq m$, where c is a constant.

- Optimize the model partition to balance the workload assigned to each workers
  - Proposed a DNN partition method based on binary-search
  - Insights
    - It is difficult to directly find the optimal partition, but we can quickly verify if a feasible partition exists given a partition limitation.

Heterogeneous Workers

- Cluster heterogeneous workers into groups, such that every group has similar computational power
  - Use min-max objective function for balancing
  - A grouping method based on local search is proposed

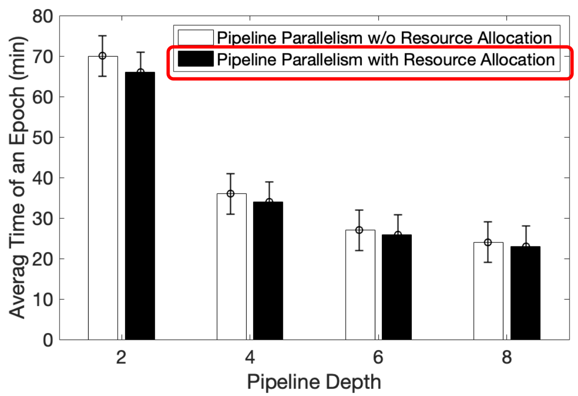
Algorithm 2 Grouping Heterogeneous Devices

Input: Heterogeneous device set V, depth of the pipeline q
Output: Workers that group heterogeneous devices $V_i, i = 1, 2, \ldots, q$

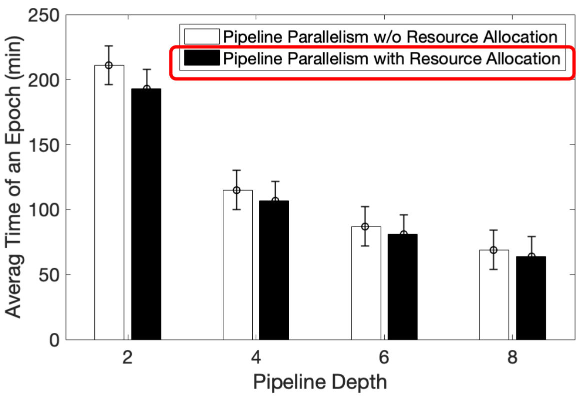
```
V_i ← ∅ for all i = 1, 2, ..., q
for i = 1, 2, ..., q do
    initialize the cost function of each worker, cost(V_i) ←
    max{f(p_i, Σ_{v∈V_i} Σ_{j=1}^{m} r_j), g(p_i, Σ_{v∈V_i} Σ_{j=1}^{m} r_j)}
while V is not empty do
    choose the worker V_i with the largest cost
    v* ← arg max_{v∈V} cost(V_i) − cost(V_i ∪ v*)
    assign v* to V_i.
    remove v* from V
return V_i, i = 1, 2, ..., q as workers
```

3. Experiment Results

Pipeline Depth

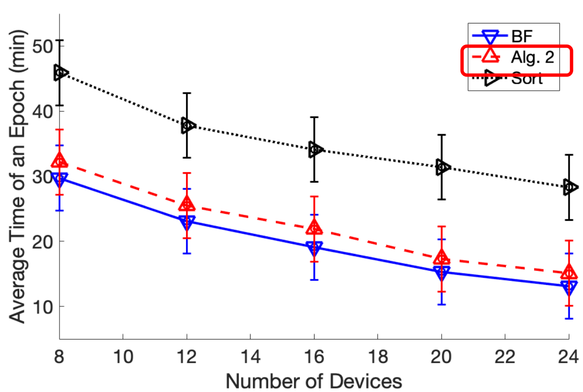


AlexNet

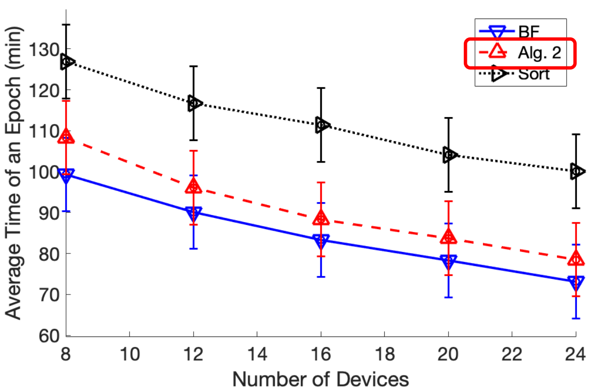


GoogLeNet

Number of devices



AlexNet



GoogLeNet

4. Conclusion

- Extend the pipeline parallelism for training DNNs on devices with multiple types of computational resources
- Homogeneous workers: theoretically analyze the resource allocation ratio, propose a model partition method
- Heterogeneous workers: propose a clustering algorithm to group workers
- Trace-based simulation shows our scheme can efficiently improve resource utilization and reduce the training time

Thank you!
Q & A

yubin.duan@temple.edu