Bamboo

Making Preemptible Instances Resilient for Affordable Training of Large DNNs

John Thorpe*, Pengzhan Zhao*, Jonathan Eyolfson, Yifan Qiao, Zhihao Jia, Minjia Zhang, Ravi Netravali, Harry Xu

UCLA    CMU    Microsoft Research    Princeton University
Generative AI Is Changing the World

How can I help?

Can you write my code for me?

…

…

…
Model Sizes are Increasing

Number of Parameters per Model

Billions of Parameters

BERT  GPT-2  Megatron  Turing-NLG  GPT-3  GPT-4

?
Prohibitive Costs for Most Organizations

32GB GPU cannot scale past 1.4B parameters
- Many accelerators needed to scale to today’s 100B+ parameter models
- $4.6 million to train GPT-3

Model Compression
- Accuracy tradeoffs

Can we take advantage of particular resources in the cloud to train large models with much lower costs?
Spot Instances Can Lower Costs

Instances can be acquired cheaply
- Up to 70% lower costs

Preemptions can be unavoidable
- Price based preemptions: Just raise bid
- Capacity preemptions: No excess capacity left
  - Unavoidable!
Spot Instances Can Have High Failure Rates

A100 (target = 64)

V100 (target = 32)

Hard to predict!
## How To Deal With Preemptions?

<table>
<thead>
<tr>
<th>Approximation</th>
<th>Checkpointing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample dropping assumes that losing some samples is acceptable</td>
<td>Roll back to stable state upon failures</td>
</tr>
<tr>
<td>● Remains true with smaller failure rates</td>
<td>● Maintains accuracy</td>
</tr>
<tr>
<td>● Loss severely impacted with higher rates</td>
<td>● Acceptable performance for low fail rates</td>
</tr>
<tr>
<td></td>
<td>● Frequent restart when failures frequent</td>
</tr>
</tbody>
</table>
Pipeline Parallelism Enables Large Model Training

Pipeline Parallelism partitions the model among workers

Stage 0
Stage 1
Stage 2
Stage 3
Redundancy Provides Resilience

Redundant stages provide redundancy more quickly

High overheads if done naively!
Pipeline Parallelism Has Bubbles

Each mini-batch split up into micro-batches

Accumulate micro-batch gradients to get full batch gradients

Node 0
Node 1
Node 2
Node 3

Bubble

Forward
Backward
Pipeline Bubbles Still Exist

Can we use this idle time to minimize redundancy overhead?

Bamboo

Node i

Node i+1

Node i+2

Forward  Backward  Redundant
Consecutive Failures Are Fatal

Node 0
Stage 0
Stage 1

Node 1

Node 2

Node 3
Stage 3
Stage 0

Active Stage
Redundant Stage
How To Avoid Consecutive Failures?

Bulk failures tend to happen in the same zone

Zone A

Zone B

Zone C

Bulk failures are indicated by red crosses.
Careful Placement Avoids Consecutive Preemptions

Zone A

Zone B

Zone C

\[ \text{Diagram showing connections between zones with consecutive preemptions marked by Xs.} \]
Careful Placement Avoids Consecutive Preemptions
Evaluated Bamboo On Many Datasets

<table>
<thead>
<tr>
<th>Model</th>
<th>DataSet</th>
<th>Data Parallel Size</th>
<th>Pipeline Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResNet-152</td>
<td>ImageNet</td>
<td>4</td>
<td>8x1.5 (12)</td>
</tr>
<tr>
<td>VGG-19</td>
<td>ImageNet</td>
<td>4</td>
<td>4x1.5 (6)</td>
</tr>
<tr>
<td>AlexNet</td>
<td>Synthetic Data</td>
<td>4</td>
<td>4x1.5 (6)</td>
</tr>
<tr>
<td>GNMT-16</td>
<td>WMT16EN-De</td>
<td>4</td>
<td>4x1.5 (6)</td>
</tr>
<tr>
<td>BERT-Large</td>
<td>Wikicorpus En</td>
<td>4</td>
<td>8x1.5 (12)</td>
</tr>
<tr>
<td>GPT-2</td>
<td>Wikicorpus En</td>
<td>4</td>
<td>8x1.5 (12)</td>
</tr>
</tbody>
</table>
Experiments

- Overall performance with respect to training costs
- Full simulations at different preemption rates
- Comparison against existing systems

Value: Performance-per-dollar
We Provide Comparable Performance to On-Demand Performance (samples/s)
Bamboo Significantly Reduces Cost

On-Demand

Value: 1.21

Bamboo

Value: 2.34

Cost ($/hr)

Time (hours)
Simulation of BERT to Completion

Simulated BERT at different levels of preemption

<table>
<thead>
<tr>
<th>Probability</th>
<th>Throughput</th>
<th>Cost ($/hr)</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>87.99</td>
<td>41.11</td>
<td>2.10</td>
</tr>
<tr>
<td>0.05</td>
<td>76.35</td>
<td>39.73</td>
<td>1.90</td>
</tr>
<tr>
<td>0.10</td>
<td>72.12</td>
<td>37.94</td>
<td>1.88</td>
</tr>
<tr>
<td>0.25</td>
<td>60.12</td>
<td>32.58</td>
<td>1.82</td>
</tr>
<tr>
<td>0.50</td>
<td>40.37</td>
<td>24.53</td>
<td>1.59</td>
</tr>
</tbody>
</table>

On-Demand Value: 1.1

Even at high preemptions maintain high value
Bamboo Provides More Value Than Similar Systems

Bamboo provides more value at different levels of preemption than Varuna.

Frequent restarts and checkpoints slow Varuna.
Bamboo Provides Resilience on Preemptible GPUs

Redundancy allows quick recovery from preemptions

Training efficiently on a changing set of resources

Provides 1.9x more value than On-Demand and 1.5x more than Varuna