GEMS: GPU-Enabled Memory-Aware Model-Parallelism System for Distributed DNN Training

Authors: Arpan Jain, Ammar Ahmad Awan, Asmaa M. Aljuhani, Jahanzeb Maqbool Hashmi, Quentin G. Anthony, Hari Subramoni, Dhableswar K. Panda, Raghu Machiraju, Anil Parwani

Presented by: Ayyappa Kolli
Outline

• Motivation
• Existing solutions and limitations
• Proposed Model parallelism systems
  • GEMS-MAST
  • GEMS-MASTER
  • GEMS-HYBRID
• Performance evaluation
• Summary/Critique
Motivation

• In basic model parallelism, only one GPU does the computation while the rest of the GPUs remain idle.

• Pipeline parallelism is only applicable if several training examples can fit inside the PE’s memory.

• Need a memory-efficient model parallelism system that offers better training speed compared to state-of-the-art systems.

• Supports emerging real-world use cases like digital pathology where 100,000 X 100,000 resolution images are used for training.
Outline

• Motivation
• Existing solutions and limitations
• Proposed Model parallelism systems
  • GEMS-MAST
  • GEMS-MASTER
  • GEMS-HYBRID
• Performance evaluation
• Summary/Critique
Basic Model parallelism

Approach:
• The out-of-core DNN is partitioned, and each partition is placed on a single GPU.

Limitations of Basic model parallelism:
• Under-utilization of resources
Model Parallelism with Pipelining

Approach:

• Pipelining divides the input batch into smaller batches called micro-batches.
• The number of parts should be equal to the number of DNN splits to utilize the pipeline fully.

Limitations:

• When the largest batch size is 1 for an out-of-core DNN on multiple GPUs, it is not possible to use pipelining as a batch cannot be further divided.
Model parallelism with delayed synchronization

Approach:
• In each backward pass, gradients are added to the previous gradients. After N backward passes, the accumulated gradients are used to update the DNN parameters

Limitations:
• It does not increase the performance as we increase the BS and throughput (images/second) remains constant in this approach
## Existing solutions limitations vs proposed designs

<table>
<thead>
<tr>
<th>Existing and Proposed Studies on Model Parallelism (MP)</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Synchronized Training</td>
</tr>
<tr>
<td>---------------------------------------------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>Basic Model Parallelism</td>
<td>✓</td>
</tr>
<tr>
<td>Model Parallelism - Delayed Synchronization</td>
<td>✓</td>
</tr>
<tr>
<td>Pipeline (GPipe [7] and HyPar-Flow [10])</td>
<td>✓</td>
</tr>
<tr>
<td>PipeDream [8]</td>
<td>✗</td>
</tr>
<tr>
<td>FlexFlow [11]</td>
<td>✓</td>
</tr>
<tr>
<td>LBANN [12]</td>
<td>✓</td>
</tr>
<tr>
<td>Mesh-TensorFlow [9]</td>
<td>✓</td>
</tr>
<tr>
<td>Proposed Designs (GEMS)</td>
<td>✓</td>
</tr>
</tbody>
</table>

**TABLE I**

Features offered by GEMS compared to existing frameworks
Outline

• Motivation
• Existing solutions and limitations
• Proposed Model parallelism systems
  • GEMS-MAST
  • GEMS-MASTER
  • GEMS-HYBRID
• Performance evaluation
• Summary/Critique
Memory Aware Synchronized Training (GEMS-MAST)

Approach:

- **Scope for improvement** - After completing the forward and backward passes for a given model partition, the GPU has both free memory and free compute.
- **GEMS-MAST** uses this free memory and compute by training a replica of the same DNN in an inverted manner.
- Each GPU will have two model partitions that are trained independently on two different batches, and their parameters are synchronized in the end.
Comparison of computation of Basic Model parallelism with the GEMS - MAST

(a) Delayed synchronization can be used in GEMS-Basic to train a model with any batch size but there is no computation overlap possible.
Outline

• Motivation
• Existing solutions and limitations
• Proposed Model parallelism systems
  • GEMS-MAST
  • GEMS-MASTER
  • GEMS-HYBRID
• Performance evaluation
• Summary/Critique
Memory Aware Synchronized Training with Enhanced Replications (GEMS-MASTER)

Approach

• Synchronizing the parameters between replicas after more than two forward and backward passes, Gradients for the same model replica can be reduced locally with a summation operation.

• Synchronization happens after all N forward and backward passes; therefore, we need only two all-reduce operations per batch.

• This approach gives improvement as you increase the batch size, which is not possible in GEMS-Basic with delayed synchronization.
Comparison of computation of Basic Model parallelism with the GEMS - MASTER

Memory Aware Synchronized Training with Enhanced Replication (GEMS-MASTER)

GEMS-Basic
Outline

• Motivation
• Existing solutions and limitations
• Proposed Model parallelism systems
  • GEMS-MAST
  • GEMS-MASTER
  • GEMS-HYBRID
• Performance evaluation
• Summary/Critique
Integrating Memory Aware Model Parallelism and Data Parallelism (GEMS-Hybrid)

GEMS-HY BASIC

• Model parallelism methods can be combined with DP such that out-of-core DNNs can be trained while maintaining the performance speedup of Data Parallelism.

Approach

• At the end of the backward pass, each GPU initiates an allreduce operation for the model partition it is responsible for.
GEMS-HY MAST and GEMS-HY MASTER

- The allreduce operation cannot happen like GEMS BASIC
- All reduce operation occurs at the end of backward pass of all the replicas.
- For GEMS-HY MASTER all-reduce operation occurs after $N$ forward backward passes of all replicas.
Outline

• Motivation
• Existing solutions and limitations
• Proposed Model parallelism systems
  • GEMS-MAST
  • GEMS-MASTER
  • GEMS-HYBRID
• Performance evaluation
• Summary/Critique
Performance evaluation

Evaluation Methodology

• The number of images processed per sec
• Time needed to reach a particular accuracy for the given DNN.

Experiment Categories

• Performance analysis of proposed designs on different DNNs
• Comparison of GEMS with existing strategies
• Showing the benefit of GEMS on a real dataset.
Performance comparison for GEMS MAST and GEMS BASIC

- As the number of model partitions increases the speed up also increases.
- Speed up will never cross more than 1.5x as we overlapped FP with BP, and allreduce is also required.
GEMS-MASTER performance comparison and validation

- They show up to 1.83 speedup for ResNet-164 on 1024 X 1024 image size using GEMS-MASTER.

Fig. 12. GEMS-Basic and GEMS-MASTER performance comparison (ResNet-164 for 1024×1024 image size on 4 GPUs)

Fig. 13. Experimental Validation of the Analytical Model for GEMS-MASTER (ResNet-1K with 512×512 image size on 8 GPUs)
• GEMS-HY MASTER (N=64) gives a better speedup than other approaches because the computation to communication ratio is higher.

• The number of DP replicas on 1024 GPUs is 128 and the model is partitioned across 8 GPUs.

• The ideal speedup is 128 X, GEMS-HY MAST and GEMS-HY MASTER give 89 X and 124.58X speedup on 128 model replicas (1024 GPUs)

Fig. 14. GEMS-Hybrid: Scaling ResNet-1K on 1,024 GPUs.
GEMS vs. Frameworks with Spatial Parallelism

• (a) shows the comparison for ResNet- 56 using 512X512 images trainable with maximum BS=8.

• GEMS-MAST is 1.31X and 1.16X better than pipelining and MTF, respectively,

• GEMS-MASTER is up to 1.54X and 1.36X better than pipelining and MTF, respectively.

• (b) shows the comparison for ResNet- 110 using 1024X1024 images trainable with maximum BS=1

• GEMS-MASTER is up to 1.74X and 1.24X better than GEMS-Basic and MTF, respectively.

(a) ResNet-56 on 4 GPUs using 512×512 images

(b) ResNet-110 on 4 GPUs using 1024×1024 images

Fig. 15. Comparison of GEMS with Mesh-TensorFlow.
GEMS vs. Frameworks with Pipeline Parallelism

- (a) shows the comparison of GEMS-MAST and torchgpipe for AmoebaNet network with 1024×1024 image size.
- GEMS-MASTER is 1.32X better than torchgpipe.
- (b) shows the comparison for ResNet-110 using 1024×1024 images trainable with maximum BS=1.
- GEMS-MAST is 1.21X and GEMS-MASTER is 1.65X better than torchgpipe

![Comparison of GEMS with torchgpipe](image)

Fig. 16. Comparison of GEMS with torchgpipe.
Performance of proposed parallelism approaches on pathology images

- ResNet110-v2 model is an out-of-core DNN, and it is not possible to train this model with a batch size of two on four GPUs with GEMS-Basic.
- Table II shows the time required to train the ResNet110-v2 model to 90% accuracy.
- Decreased the training time from 7.2 hours to 28 minutes using GEMS-Hybrid.

<table>
<thead>
<tr>
<th>Approach</th>
<th># Nodes (#GPUs)</th>
<th>Epochs</th>
<th>Learning Rate</th>
<th>Total Time in hours (Total Time in mins)</th>
<th>Training speed (img/sec)</th>
<th>Speedup Over GEMS-MASTER</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEMS-Basic</td>
<td>1 (4)</td>
<td>8</td>
<td>0.001</td>
<td>7.25 (435)</td>
<td>1.04</td>
<td>-</td>
</tr>
<tr>
<td>GEMS-MAST</td>
<td>1 (4)</td>
<td>8</td>
<td>0.001</td>
<td>6.28 (377)</td>
<td>1.19</td>
<td>-</td>
</tr>
<tr>
<td>GEMS-MASTER</td>
<td>1 (4)</td>
<td>7</td>
<td>0.001</td>
<td>4.21 (252)</td>
<td>1.56</td>
<td>1</td>
</tr>
<tr>
<td>GEMS-Hybrid</td>
<td>2 (8)</td>
<td>7</td>
<td>0.001</td>
<td>2.51 (151)</td>
<td>2.99</td>
<td>1.9</td>
</tr>
<tr>
<td>GEMS-Hybrid</td>
<td>4 (16)</td>
<td>7</td>
<td>0.001</td>
<td>1.34 (80)</td>
<td>5.65</td>
<td>3.6</td>
</tr>
<tr>
<td>GEMS-Hybrid</td>
<td>8 (32)</td>
<td>10</td>
<td>0.0025</td>
<td>0.98 (59)</td>
<td>10.69</td>
<td>6.8</td>
</tr>
<tr>
<td>GEMS-Hybrid</td>
<td>16 (64)</td>
<td>12</td>
<td>0.0025</td>
<td>0.65 (39.5)</td>
<td>19.18</td>
<td>12.3</td>
</tr>
<tr>
<td>GEMS-Hybrid</td>
<td>32 (128)</td>
<td>15</td>
<td>0.004</td>
<td>0.46 (27.8)</td>
<td>34.51</td>
<td>22.1</td>
</tr>
</tbody>
</table>

**Table II**
Scaling ResNet-110 v2 on 1024 × 1024 image tiles extracted from histopathology data (cf. Figure 2)
Outline

• Motivation
• Existing solutions and limitations
• Proposed Model parallelism systems
  • GEMS-MAST
  • GEMS-MASTER
  • GEMS-HYBRID
• Performance evaluation
• Summary/Critique
Summary / Critique

• Speeding up a deep learning model by 22 times is great performance optimization.
• How Batch Normalization is performed on proposed parallelism approaches is not discussed.