

Performance Analysis of Deep Learning Workloads on Leading-edge Systems

Yihui (Ray) Ren

Shinjae Yoo

Adolfy Hoisie

Computational Science Initiative

BROOKHAVEN
NATIONAL LABORATORY

 U.S. DEPARTMENT OF
ENERGY

BROOKHAVEN SCIENCE ASSOCIATES

Outline

- Hardware Systems
- Communication Bandwidth Test
- Deep Learning Workloads
- Training Throughput Results

GPUs and NVLink

- NVIDIA Tesla V100 (32GB HBM2 memory, 6 NVLinks, 15.7 TFLOPS)
- NVIDIA RTX 2080Ti (11GB GDDR6 memory, 2 NVLinks, 13.4TFLOPS)
- Each NVLink has bandwidth of 25GB/s in and 25GB/s out.

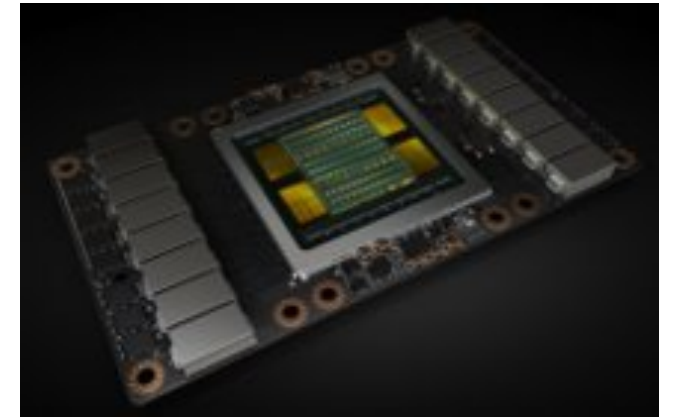
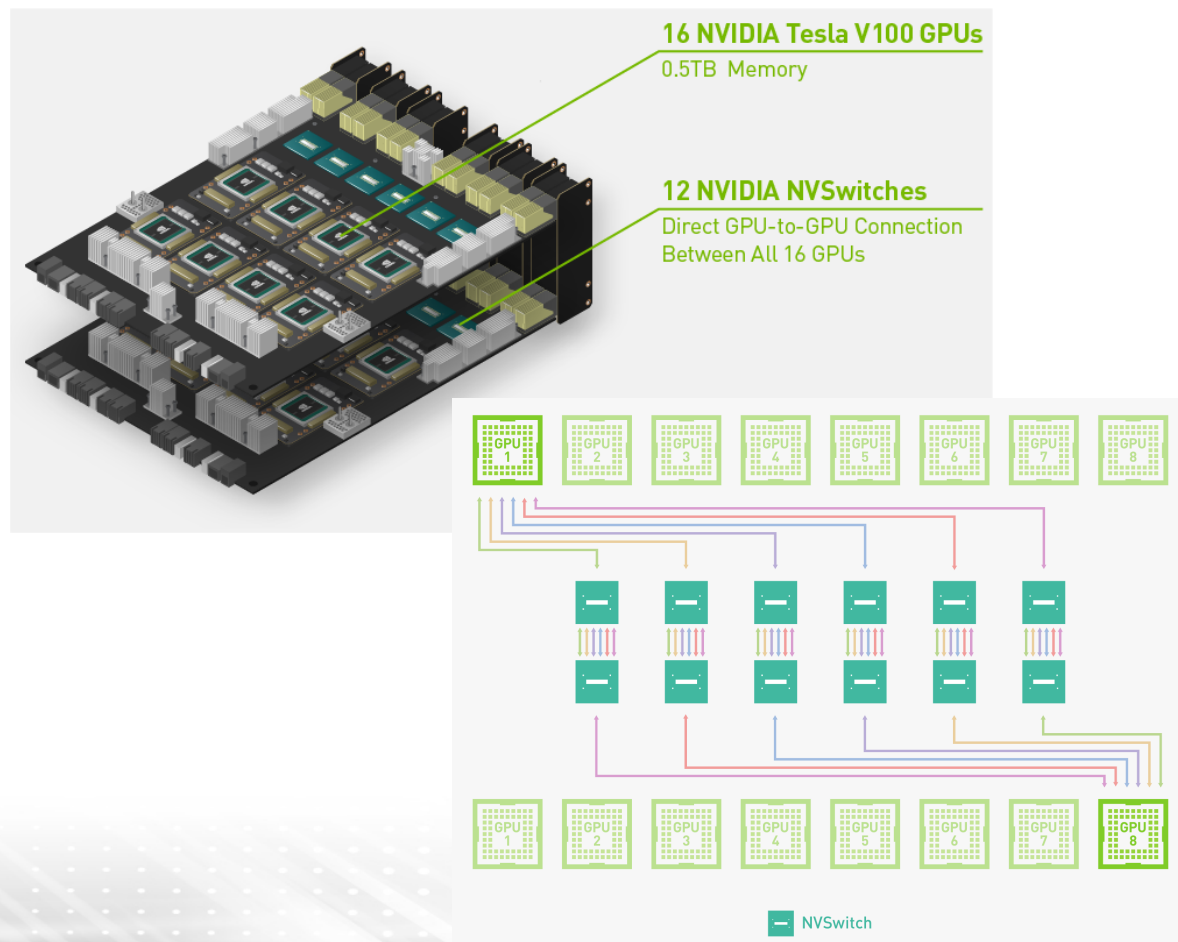


Image Ref:

<https://devblogs.nvidia.com/using-cuda-warp-level-primitives/>

DGX-2 and NVSwitch

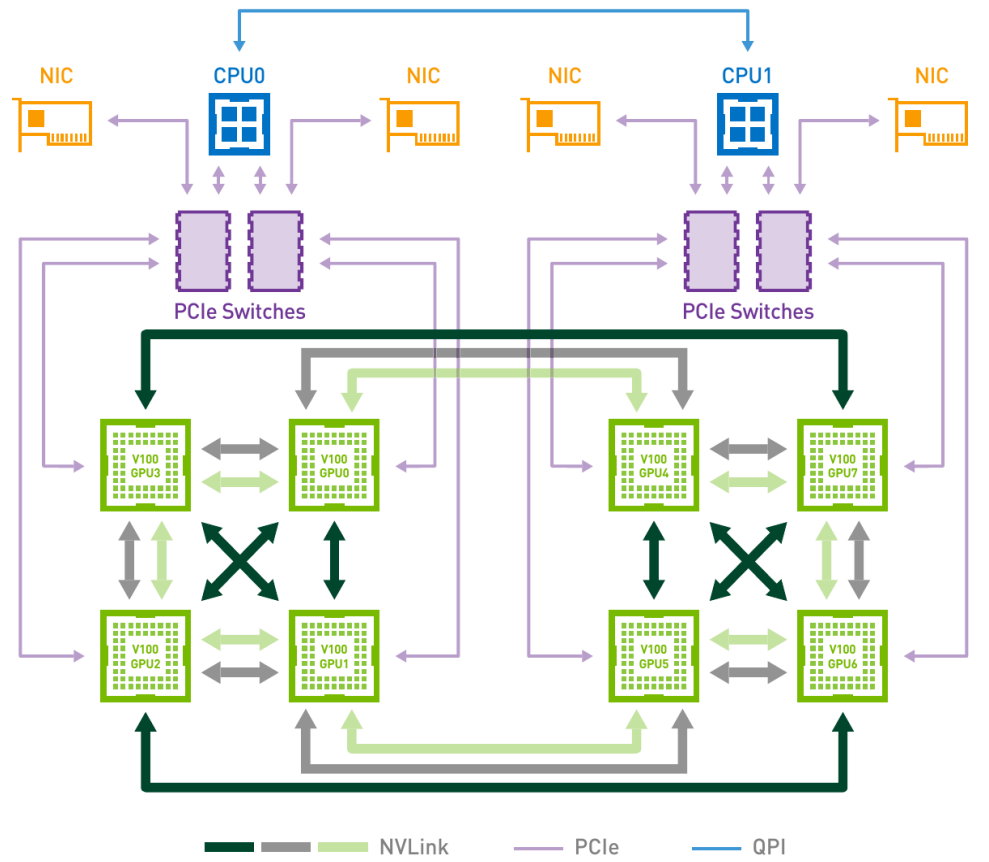
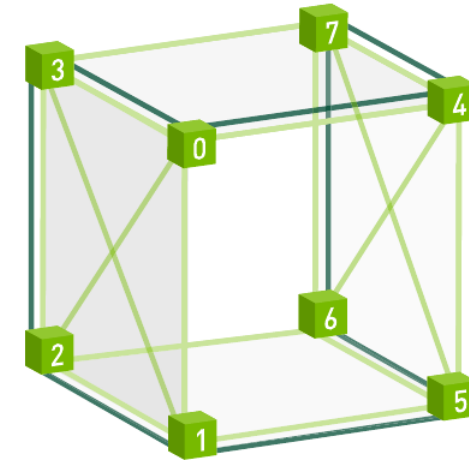


- 16x V100 (32GB) GPUs
- 12x on-node NVSwitches
- Each NVSwitch has 18 NVLink ports (16 in use).
- 2x 24-core Xeon 8186 (96 logic cores in total)
- 1.5 TB system memory
- 30 TB NVMe SSD in 8-way RAID0

Image Ref:

<https://images.nvidia.com/content/pdf/nvswitch-technical-overview.pdf>
<https://www.nvidia.com/en-us/data-center/hgx/>

AWS p3dn.24xlarge (DGX-1)



- 8x V100 (32GB) GPUs
- Hybrid cube-mesh topology
- 2x 24-core Xeon 8175M (96 logic cores in total)
- 768 GB system memory
- 2 TB NVMe SSD
- 2x AWS P3 (16 GPUs in total)
- Connected through 1.25GB/s Ethernet.

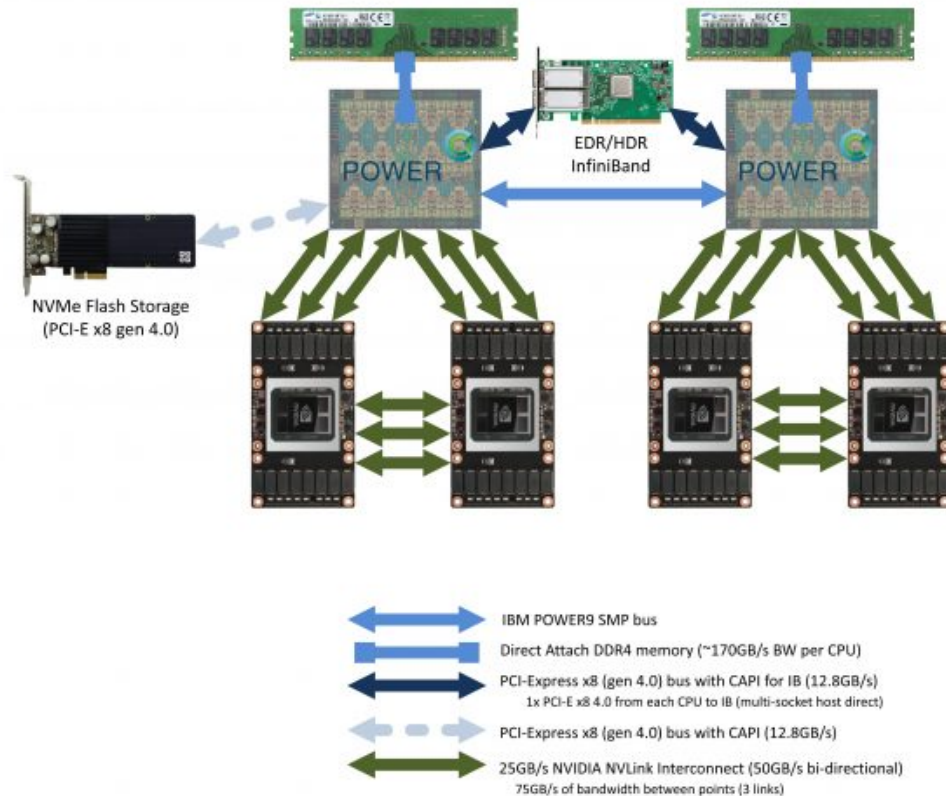
Image Ref:

<https://images.nvidia.com/content/pdf/dgx1-v100-system-architecture-whitepaper.pdf>

IBM Power System AC922 (8335-GTH)

Server Block Diagram

Power Systems AC922 with NVIDIA Tesla V100 with Enhanced NVLink GPUs



- 4x V100 (32GB) GPUs
- 2x IBM 20-core Power9 CPU (160 logic cores in total)
- Each IBM Power9 CPU has 6 NVLinks.
- Two CPUs are connected by a SMP bus (32GB/s).
- 4x IBM P9 systems (16 GPUs in total)
- Connected through InfiniBand (24 GB/s).
- The tested system uses GPFS (remote filesystem) with block size of 16 MB and bandwidth ~18 GB/s.

Image Ref:

<https://www.microway.com/product/ibm-power-systems-ac922/>

Exact TensorEX TS4 GPU Server



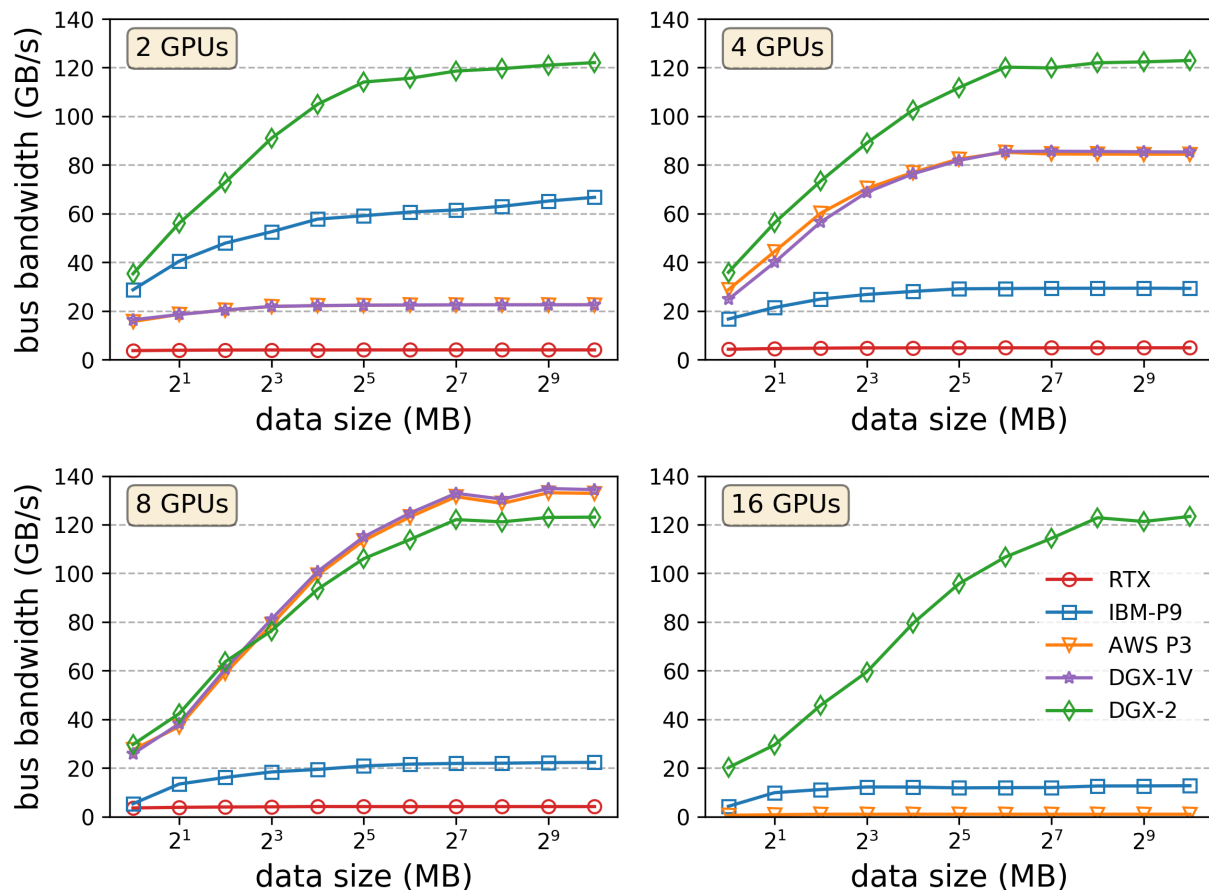
- (TS4-1598415-DPN)
- 8x RTX 2080 Ti GPU
- All GPUs are connected by a PCIe bus.
- 2x 12-core Xeon 4116 CPUs (48 logic cores in total)

Image Ref:

<https://www.exactcorp.com/Exact-TS4-1598415-E1598415>

Inter-device Communication Bandwidth

NCCL All-reduce



- All-reduce operation is performed at the end of every iteration during training.
- **DGX-2** has consistent peak unidirectional bus bandwidth of 120 GB/s.
- **DGX-1** and **AWS P3** have the same NVLink Topology.
- **IBM-P9** has better 2-GPU communication bandwidth. (3 NVLinks)
- **RTX** uses a PCI-e bus.
- When communicate across nodes, **IBM-P9** and **AWS P3** are bottlenecked by the Infiniband / Ethernet.

Source code Ref:

<https://github.com/NVIDIA/nccl-tests>

Deep Learning workloads

Model Name	Param.
AlexNet	61.10 M
ResNet18	11.69 M
ResNet50	25.56 M
ResNet101	44.55 M
ResNet152	60.19 M
BERT-SWAG	109.5 M
BERT-SQuAD	109.5 M

High-throughput

Large Models

- PyTorch 1.0
- Computer Vision (AlexNet and ResNet)
 - Image Classification (ImageNet)
- Natural Language Processing (BERT)
 - SWAG
 - SQuAD

Performance Factors:

- Model Complexity (number of operations)
- Number of Parameters (affects the communication cost)
- GPU memory size (affects batch size, therefore, consequently the number of synchronizations.)

Paper Ref:

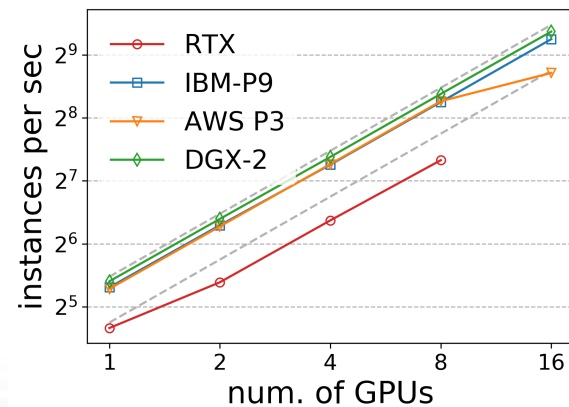
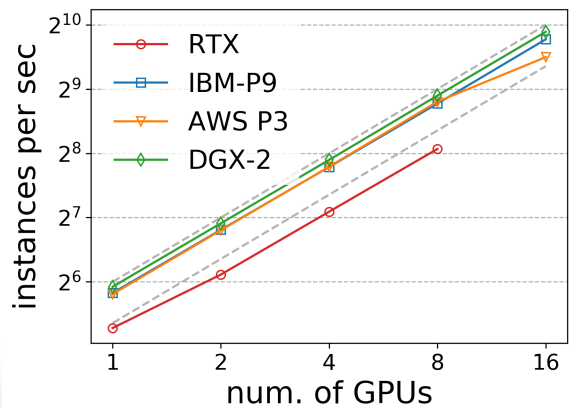
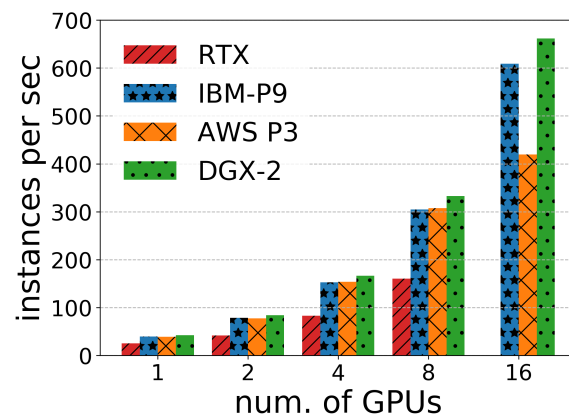
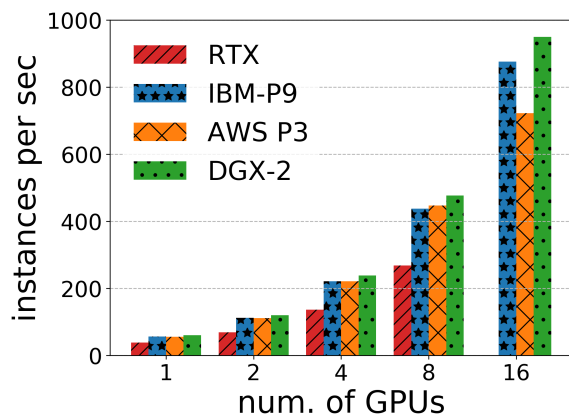
ResNet: <https://arxiv.org/abs/1512.03385>

BERT: <https://arxiv.org/abs/1810.04805>

SWAG: <https://arxiv.org/abs/1808.05326>

SQuAD: <https://rajpurkar.github.io/SQuAD-explorer/explore/1.1/dev/>

Results: BERT (Large Models)

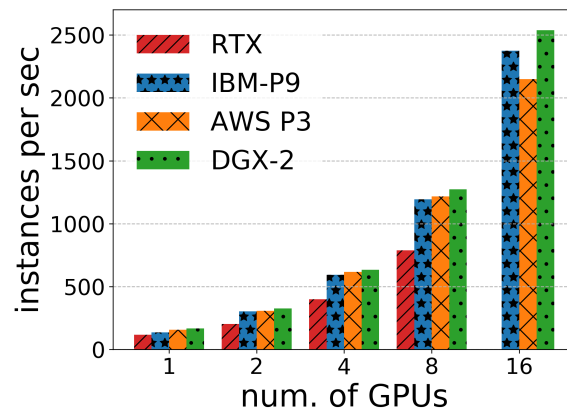
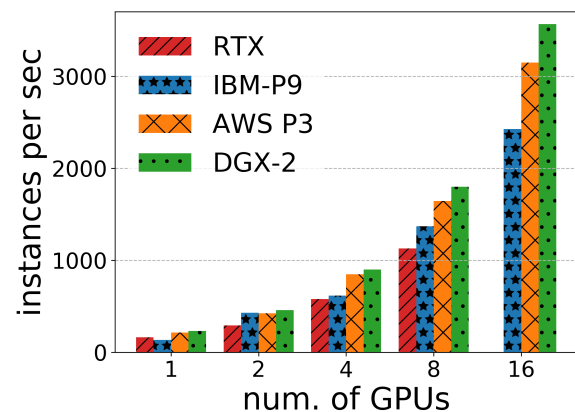


BERT-SWAG

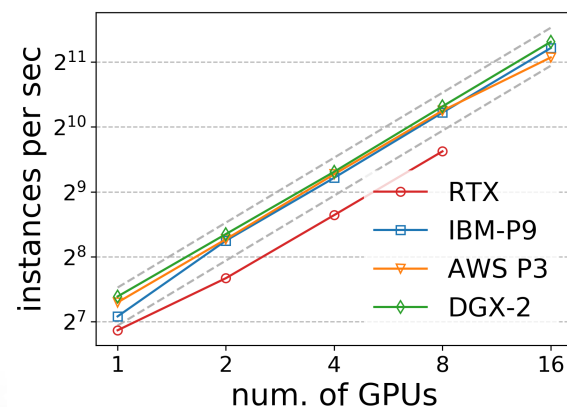
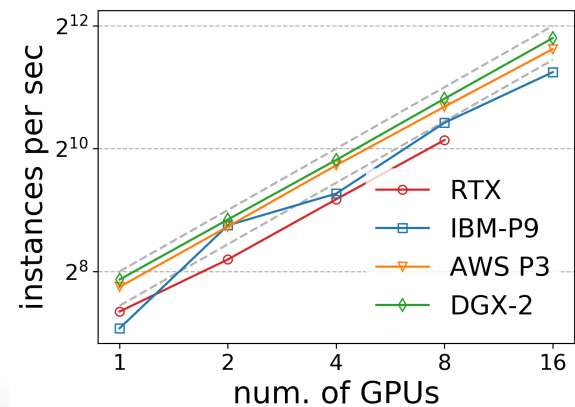
BERT-SQuAD

- BERT has 109M parameters.
- Batch size 64 (SWAG) and 32 (SQuAD).
- Max-seq-length of 80 (SWAG) and 384 (SQuAD)
- AWS P3 does not scale well in the case of 16 GPUs

Results: ResNet-101 and 152 (Large Models)



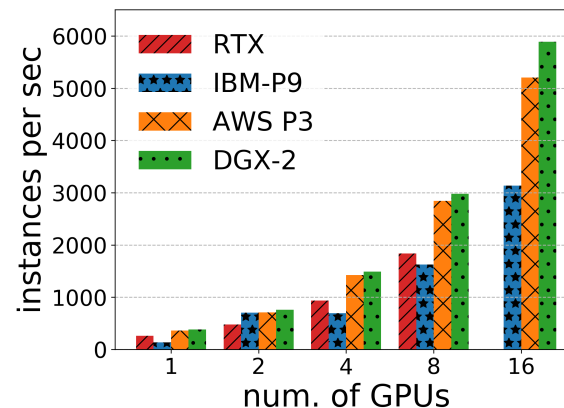
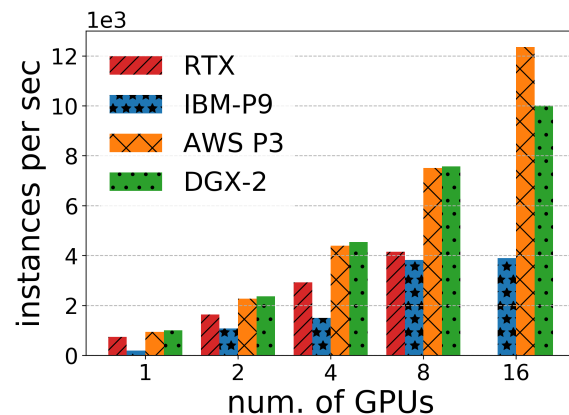
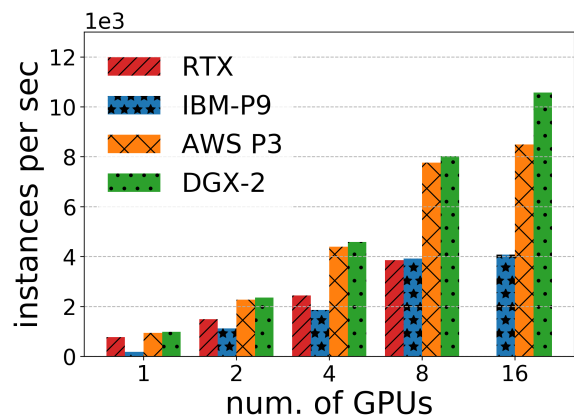
- Batch size of 128.
- ResNet101 has 44.55M parameters
- ResNet152 has 60.19M parameters



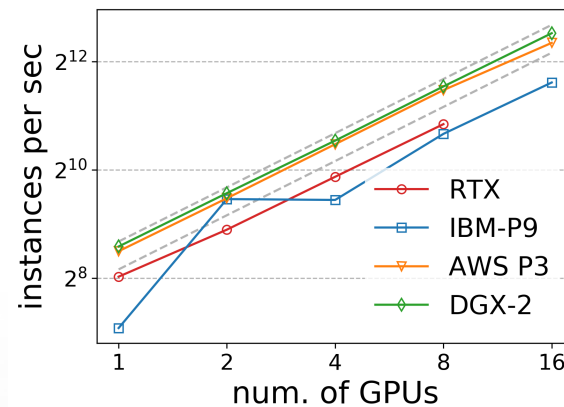
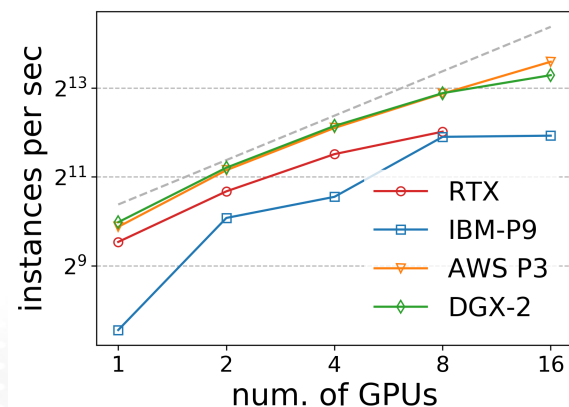
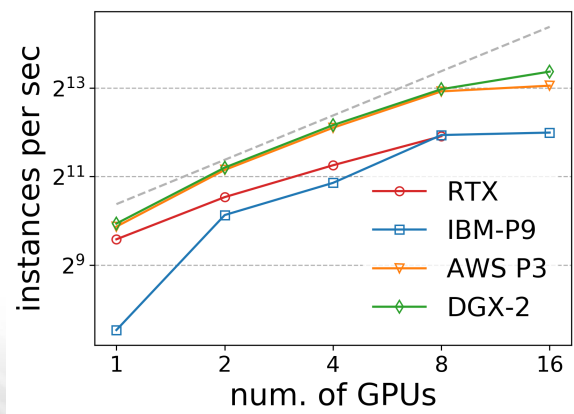
ResNet-101

ResNet-152

Results: High-throughput models



- Batch size of 256
- Extremely data-intensive (10,000 images per sec).
- The tested IBM P9 has remote filesystem and large block size (16MB).

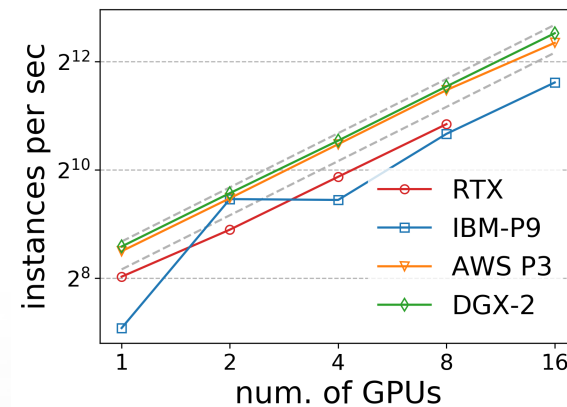
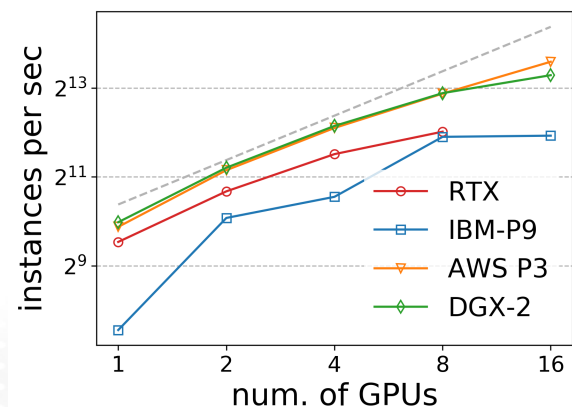
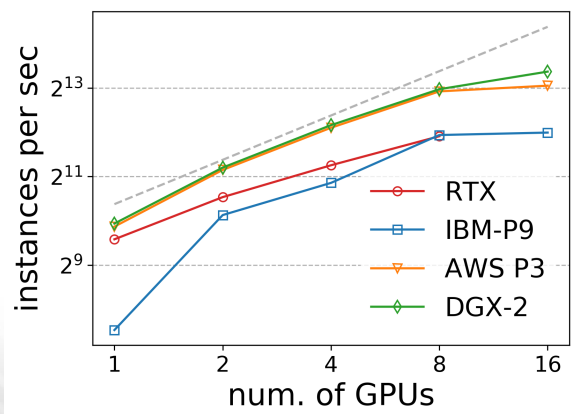
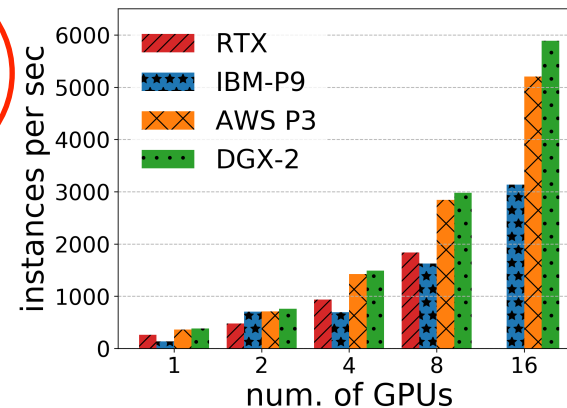
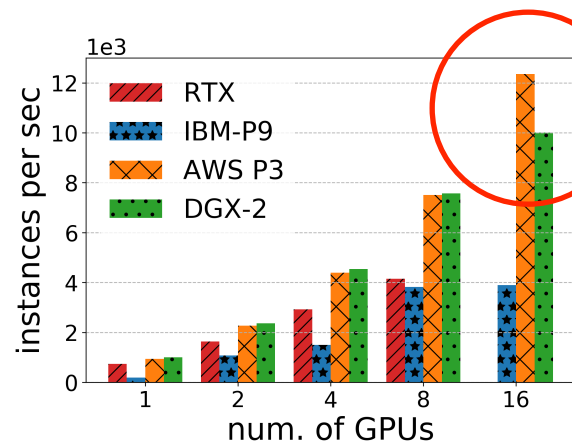
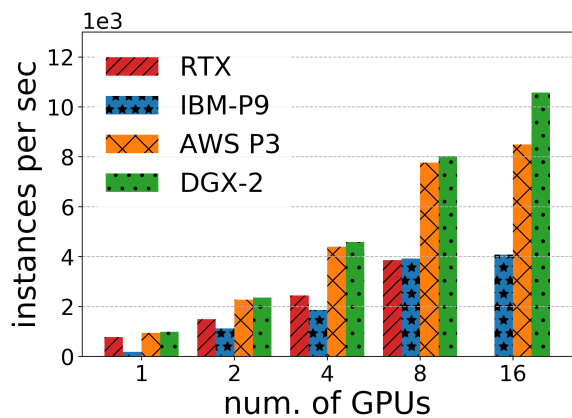


AlexNet

ResNet18

ResNet50

Results: High-throughput models



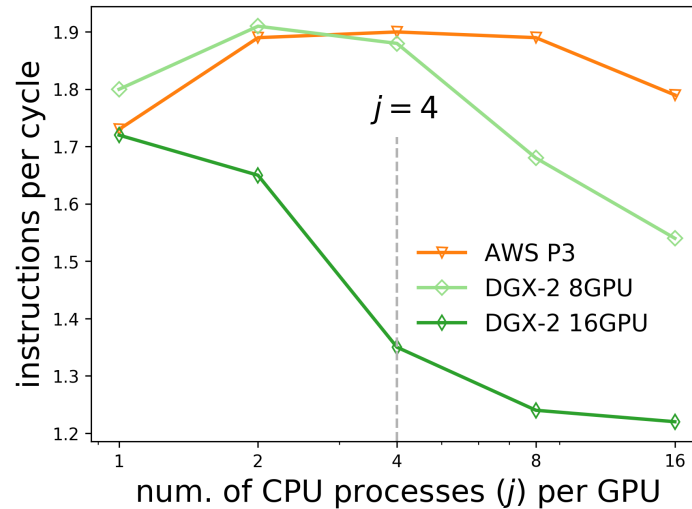
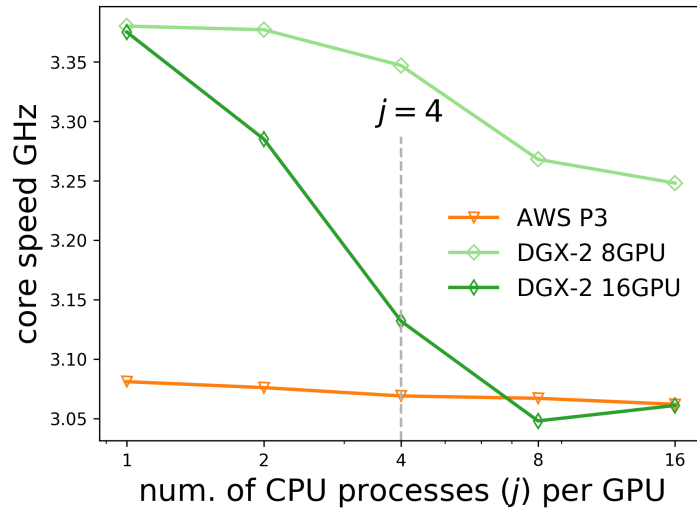
AlexNet

ResNet18

ResNet50

- Batch size of 256
- Extremely data-intensive (10,000 images per sec).
- The tested IBM P9 has remote filesystem and large block size (16MB).
- Two AWS P3 better than one DGX-2 in the case of ResNet18

Results: Investigation on CPU bottleneck



- j : the number of CPU data-fetching processes associated with each GPU.
- DGX-2 has better CPU (higher clock speed). Larger j affects Intel turbo boost. But still better than AWS P3 at $j=4$.
- But lower IPC, indicating CPU cache bottleneck.

Results: RTX vs DGX-2

Model Name	1 GPU	2 GPUs	4 GPUs	8 GPUs
AlexNet	78.19%	63.01%	53.41%	47.95%
ResNet18	73.50%	69.13%	64.39%	54.80%
ResNet50	67.97%	62.67%	62.97%	61.75%
Average	73.22%	64.94%	60.26%	54.83%
ResNet101	69.70%	63.72%	64.15%	62.69%
ResNet152	69.73%	62.45%	62.96%	61.90%
BERT-SWAG	64.04%	57.52%	57.20%	56.25%
BERT-SQuAD	59.81%	49.79%	49.74%	48.22%
Average	65.82%	58.37%	58.51%	57.27%
Overall avg.	68.99%	61.19%	59.26%	56.22%

Conclusion

- The DGX-2 offers the best 16 GPU collective communication.
- Leading-edge systems scale well with large deep learning workloads
- RTX suits best for small-scale training and model development.

Acknowledgements

- Ethan Hereth, Anthony Skjellum (University of Tennessee)
- Xinghong He, Mladen Karcic and Douglas L. Lehr (IBM)
- Brian Barrett (Amazon Web Services)
- Craig Tierney and Louis Capps (NVIDIA)
- Zhihua Dong (BNL)
- This work was funded as part of the *Exploiting the Convergence of Research Challenges in Scientific Discovery and National Security* program within Brookhaven Lab's Computational Science Initiative.

Thank you

- Questions?