Cost-Effective High Performance Distributed GPU Cluster for Deep Learning Tasks

Kirill Karpov, Dmitry Kachan, Maksim Iushchenko, Ivan Luzianin and Eduard Siemens
Department of Electrical, Mechanical and Industrial Engineering, Anhalt University of Applied Sciences,
55 Bernburger Str., Köthen, Germany
{*kirill.karpov, dmitry.kachan, maksim.iushchenko, ivan.luzianin, eduard.siemens*}*@hs-anhalt.de*

Keywords: Tensor Flow, DNN Training, Performance, Horovod, HPC, High-Performance Computing.

Abstract: The expenses on computational resources for modern Deep Learning computing can be extremely large. However, most of them are spent on the chassis and not on the GPU units themselves. Since modern mass market graphic cards are usually chipper and have huge performance for video games, it was hypothesized, that a low cost cluster, made of several graphic cards, can reach the same performance for computational tasks as ready-made enterprise GPU-server with significantly lower p rice. The concept of distributed GPU cluster based on mass market GPU units is presented in the article. During the experiments, performance of a cluster with two mass market GPU units was compared with performance of enterprise GPU-server with 8 GPU-units on the Deep Learning bench mark. The results shows benefits and limitations of use proposed distributed c luster. It describes cases, when this solution is up to 7 times more effective than enterprise one in terms of cost savings for chassis itself as well as for, additional equipment and maintenance.

1 INTRODUCTION

The rapid development of GPU-accelerated computing systems makes Deep Learning (DL) algorithms the most powerful Machine Learning solutions. At the same time, DL models require complex and expensive enterprise-level High-Performance Computing (HPC) hardware for training. They require additional equipment for operating such as racks, air conditioning, power supply systems, etc. Moreover, these solutions require high-qualified personnel. Finally, enterprise GPU units are generally more expensive than usual servers.

In opposite, mass-market graphic cards do not require special conditions and maintenance to operate. Modern ones are good enough to perform DL computational tasks. Therefore, potentially, if several mass-market GPU modules are incorporated together in a single cluster, then the overall cluster's performance will be the same as for the enterprise solution but with a lower price. This may be useful if there is a need to create a GPU-accelerated computing cluster that is simple to use and does not require special conditions to operate.

It is possible to create a computing cluster of multiple single nodes using technologies like RDMA [1], MPI [2], and NCCL [3]. Furthermore, the frameworks like Horovod [4, 5] provide a simple interface to their technologies for deep learning tasks.

This work aims to confirm the hypothesis that it is possible to create a cost-effective cluster of multiple mass-market GPU nodes with a higher performance than an enterprise solution has.

It is necessary to perform the following tasks to reach the goal:

- to prepare the testbeds for mass-market and enterprise solutions;
- to provide performance measurements on both testbeds;
- to analyze and conclude the results of the performance measurements.

The remainder of this paper is structured as follows: Section 2 describes the hardware and software equipment of the current work. Section 3 presents the results of the performance measurements of experiments. Finally, Section 4 discusses the results, followed by the conclusion in Section 5.

2 EXPERIMENTAL SETUP

The experiments were carried out on two testbeds: Single-Chassis GPU Infrastructure and PC Cluster-

Based GPU Infrastructure with the same software components.

2.1 Hardware Setup

2.1.1 Single-Chassis GPU Infrastructure

The enterprise-level solution for AI and Deep Learning data centers is represented by the Supermicro SYS-4029GP-TVRT server that has eight Tesla V100 GPUs. The system is shown in the Figure 1. In the current work, this testbed is named as **Single-Chassis GPU Infrastructure**. The server supports Nvidia's Volta V100 SXM2 form factor GPUs that benefit from Nvidia's NVLINK architecture to deliver GPU to CPU data rates of up to 300GB/s compared to using PCIe based GPUs which offer only up to 32GB/s data rates. GPUDirect Remote Direct Memory Access (RDMA) technology allows direct peer-to-peer (P2P) data exchanges between other devices in the network, bypassing the CPU and reducing GPU to GPU latency.

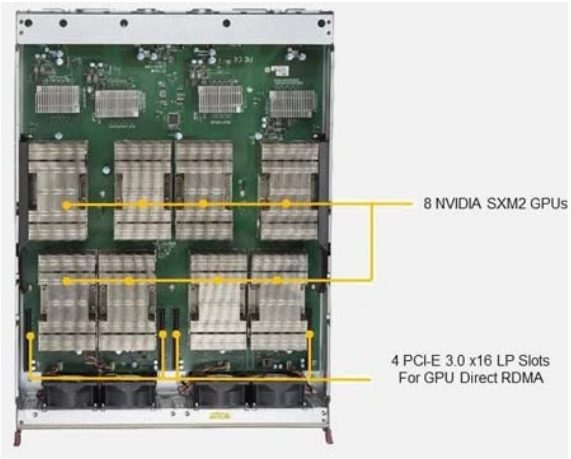

Figure 1: Supermicro SYS-4029GP-TVRT.

The detailed configuration of the Single Chassis GPU Infrastructure testbed is provided in the Table 1.

Table 1: Single-Chassis GPU Infrastructure Configuration.

System Type	Supermicro SYS-4029GP-TVRT
CPU	2x Inel Xeon Platinum 8268, 24-Core, 2.9 GHz, HT, 35,75MB Cache
Motherboard	Dual Socket P (LGA 3647) intel Xeon Scalable, X11DGO-T
Chipset	Intel C621, UPI upt to 10.4 GT/sec
GPU	8x Nvidia Tesla V100, 32 GB CoWoS HBM2, SXM2 - NVLink 2.0, CUDA Cores: 5120, Core Clock: 1455 MHz, FP32 Computing Performance: 15.0 TF, Memory Bandwidth: 900 GB/s, Memory Type: 4096-bit 16 GB HBM2, GPU: GV100 (Volta),
RAM	24x 64 GB DDR4, PC2933, ECC registered
Price	112,000 € (2020)

2.1.2 PC-Cluster Based GPU Infrastructure

The PC-based testbed consists of two nodes. Each node is composed of a small-form-factor computer Intel NUCs Ghost Canyon 9 Extreame (Figure 2) with Intel-i7 CPU and an external GPU module Aorus gaming Box RTX 3090 (Figure 3) connected to the computer via Thunderbolt 3 (TB3).

Figure 2: Intel Ghost Canyon 9 Extreme PC.

Since a NUC 9 computer has an additional PCIe3 16x slot, it is equipped with Mellanox ConnectX-5 100G NIC for high-bandwidth and low-latency network communication that is a crucial part of distributed computing.

Figure 3: Aorus Gaming Box RTX 3090 eGPU.

The scheme of PC Cluster Based GPU Infrastructure setup is shown in the Figure 4. The NUC computers, each equipped with Mellanox 100G network adapter (NIC), connect to their respective external GPU modules via TB3 at the speed up to 40Gbps. In this work the NUCs' network adapters are connected via 100G network switch. However it is possible to connect them directly in point-to-point manner.

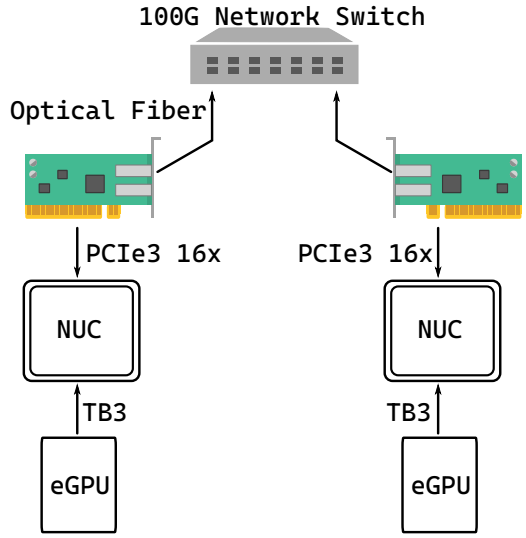

Figure 4: PC-Cluster Based GPU Infrastructure Setup.

The detailed configuration of PC-Cluster Based GPU Infrastructure is provided in the Table 2.

Table 2: PC-Cluster Based GPU Infrastructure Configuration.

PC Model	NUC 9 Extreme Ghost Canyon [6, 7]
CPU	i7-9750H
Motherboard	Dual Socket P (LGA 3647) intel Xeon Scalable, X11DGO-T
Chipset	Intel C621, UPI upt to 10.4 GT/sec
eGPU	Aorus Gaming Box RTX 3090
RAM	16 GB DDR4
NIC	Mellanox ConnectX-5, 2x100GbE, QSFP28 [8]
100G Switch	Extreme Networks x870-32c
Price for a node	3,900 € (2021)
Price for a switch	27,000 € (2020)

2.2 Software Setup

All tests were performed with the same software set for both PC Cluster-Based GPU Infrastructure and Single Chassis GPU Infrastructure testbeds.

The computers are equipped with Ubuntu 20.04 operating system with 5.4.0.97-lowlatency kernel. Low latency kernel contains optimizations, such as Preempt-RT, to achieve the lowest possible latency for applications.

Since the network is the bottleneck that significantly affects the scaling factor [9], the operating system's TCP stack was tuned to achieve the bandwidth closest to 100 Gbps with the following configuration:

```
net.core.rmem_max = 16777216
net.core.wmem_max = 16777216
net.core.rmem_default = 16777216
net.core.wmem_default = 16777216
net.ipv4.tcp_rmem = 4096 87380 16777216
net.ipv4.tcp_wmem = 4096 87380 16777216
net.ipv4.tcp_mem = 1638400 1638400 1638400
net.ipv4.tcp_sack = 0
net.ipv4.tcp_dsack = 0
net.ipv4.tcp_fack = 0
net.ipv4.tcp_slow_start_after_idle = 0
jumbo_frames=yes (default no)
```

In this work, Horovod [4] is used as a distributed learning framework. Horovod supports the Remote Direct Memory Access (RDMA) technology [1] improves its efficiency. RDMA provides access to the memory from one computer to the memory of another computer without involving either computer's operating system. This technology enables high-throughput and low-latency networking with low CPU utilization. The Mellanox ConnectX-5 NICs of the NUCs were configured with `mlnx-en-5.5-1.0.3.2` driver and MLNX OFED version 4.9-4.1.7.0. These NICs make use of RDMA over Converged Ethernet (RoCE) - a network protocol that enables remote direct memory access (RDMA) over Ethernet.

The network performance was tested using qperf [10] tool with the following result for TCP:

```
# qperf nuc2.lab tcp_bw tcp_lat
tcp_bw:
    bw  =  6.6 GB/sec (52.8 Gbps)
tcp_lat:
    latency  =  8.6 us
```

And for RDMA over Converged Ethernet:

```
# qperf -cm1 nuc2.lab rc_bw rc_lat
rc_bw:
    bw  =  12.2 GB/sec (97.6 Gbps)
rc_lat:
    latency  =  5.3 us
```

This network configuration meets the Horovod [5] requirements. Horovod is a distributed deep learning training framework for TensorFlow, Keras, PyTorch, and Apache MXNet. The goal of Horovod is to make distributed deep learning fast and easy to use. The Horovod version 0.22.1 deployed on both NUCs as a Docker [11] container provided by developers. It is configured with the following components: TensorFlow [12], PyTorch, MXNet, MPI, Gloo, NCCL, and CUDA 11.0.

Tensorflow 2.4 [12] is the framework to help develop and run DL-based solutions and is used to estimate the performance of distributed learning. TensorFlow is one of the most popular machine learning frameworks, and it has been used in a wide variety of applications and to conduct AI research.

3 EXPERIMENTAL RESULTS

The performance measurement experiments were provided using Horovod-adopted Tesnsorflow2 syn-

thetic benchmark with ResNet101 model and batch size 64. The estimated metric is Images/sec provided by the Tensorflow2 benchmark. Each run was performed 10 times for each number of GPUs.

Figure 5 shows the performance of the Single-Chassis GPU setup with 8 Tesla V100 GPUs.

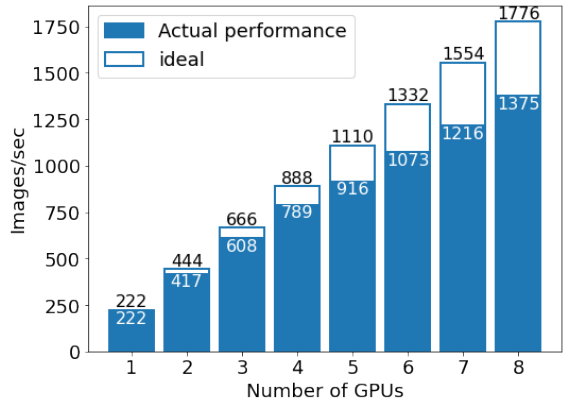

Figure 5: Multi-GPU scaling performance of Single Chassis GPU infrastructure using TensorFlow. The ideal value is the value of a single GPU performance multiplied by the corresponding number of GPUs.

As can be seen, the performance of scaling efficiency drops with each additional GPU. Whereas two parallel GPUs lose only 5 % of the ideal performance, the eight GPUs lose 22 %.

Figure 6 shows the performance of PC Cluster-Based GPU setup with two RTX 3090 GPUs.

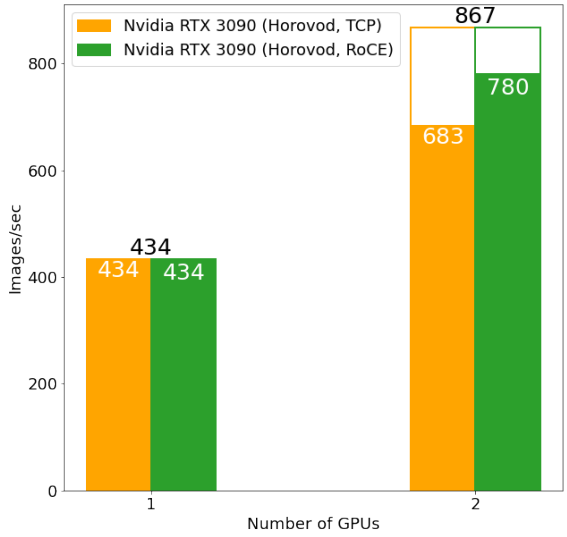

Figure 6: A comparison of the images processed per second by Horovod framework over plain 100GbE TCP and 100GbE RoCE-capable networking.

In this case, the drop of performance scaling efficiency is significantly higher than in the previous experiment. Two GPUs communicated via TCP stack lose 21% of ideal performance. RDMA brings slightly better results, the difference between the ideal value and the actual one is 10%. Nevertheless, the performance of two GPUs in absolute values in the PC Cluster-Based GPU testbed is comparable to the four GPUs in the Single Chassis testbed.

4 THE DISCUSSION OF THE RESULTS

There is a significant difference between the test scenarios in the number of parallel nodes, which makes the comparison quite challenging. Fortunately, there is a method that estimates the performance of scalable computing systems. The Universal Scalability Law [13] (USL) is an extension of Amdahl's Law that corrects the performance concerning communication latency between nodes. The USL is given by the (1).

$$S(n) = \frac{\gamma n}{1+\alpha(n-1)+\beta(n^2-n)} \qquad (1)$$

The coefficient γ represents the slope associated with linear-rising scalability in the case of ideal parallelism, α defines the serial coefficient, and β represents additional delays (see Figure 7).

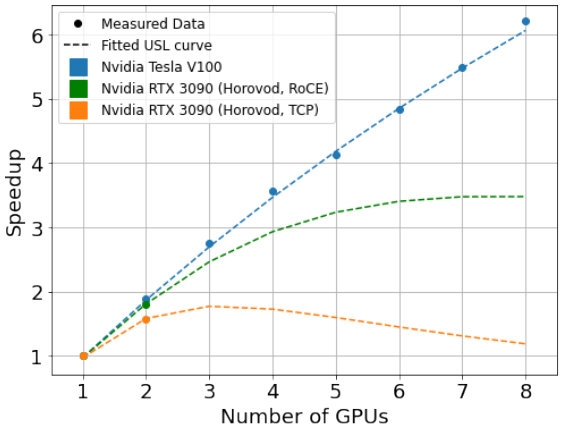

Figure 7: The result of fitting the measured data to the USL formula.

Using the Universal Scalability Law (USL) fit technique described in [14] the following parameters for the equation 1 were obtained: $\alpha = 0.04$, $\beta = 0$, $\gamma = 0.97$.

In order to evaluate the performance of the PC-Cluster Based GPU Infrastructure testbed the only parameter of additional delay was variated to fit the first

two points. The result of estimation is shown in the Figure 8.

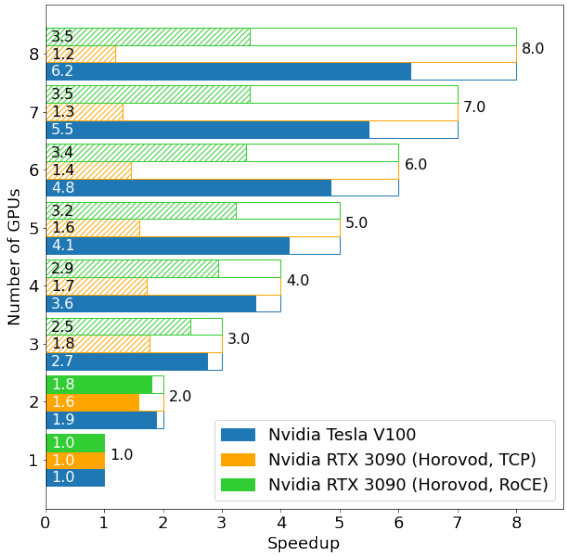

Figure 8: The dependency of speedup coefficient o n the number of GPUs. The colored frame is an ideal speedup value. The stroked column is an extrapolated value of the speedup coefficient.

With known speedup coefficients it is possible to estimate the absolute performance of the PC-based setup. The result of estimation is shown in the Figure 9.

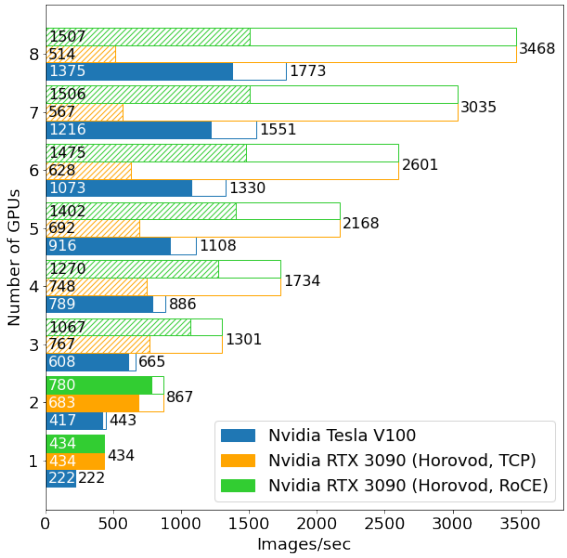

Figure 9: The dependency plot of learning performance on the number of GPUs. The colored frame is an ideal performance value. The stroked column is an extrapolated value of the performance.

Relying on the estimation it is possible to conclude that PC-Cluster Based GPU Infrastructure testbed consisting of five nodes orchestrated by Horovod and communicated via RoCE protocol can surpass the Single Chassis setup in terms of the number of processed images/sec.

5 CONCLUSIONS

The experimental results clearly show that the distributed setup with only two GPUs provides half of the performance of the modern and professional solution that costs ca. 14 times higher. However, the scalability of PC-based solutions is limited. The estimation shows that the performance of PC Cluster-Based solution with 5 GPUs can surpass Single-Chassis, though, the following scaling is unreasonable. In this case, the cost-efficiency drops significantly, since the setup with more than two GPUs requires a 100G network switch. Here the estimated cost win is supposed to be 2.5 times.

Pros and cons of PC-based Cluster infrastructure:

+ The results show that the PC Cluster-Based solution is very cost-efficient within the defined limits.
+ The system is easily upgradable.
+ Flexibility is one of the main advantages of this setup. The PC-based setup can be assembled any time in many ways. It also might be distributed from a location perspective. Since the setup is as compact as a usual desktop, the nodes can be distributed across working tables, offices, or even building, if the optical network infrastructure allows it.
+ This is a multi-purpose solution. Since each node is basically a usual PC, with the corresponding periphery interfaces (USBs, HDMI, Ethernet, etc), it might be used as an office workstation during the day, and as a node of the cluster the rest of the time.
+ Each node or even element of the node (PC, NIC, eGPU) can be easily changed in case of malfunctioning. Also, in the event of an overcurrent in a nodal element, this highly likely will not lead to a malfunction of the node or the entire system.
- The scaling factor drops with the little number of nodes. As it was shown, in this particular case, after five nodes the scaling is meaningless.
- The system is limited in distributed computational tasks. It is necessary to use special frameworks which are able to distribute desired tasks across the nodes.

REFERENCES

[1] "RDMA Aware Networks Programming User Manual," Mellanox Technologies, Tech. Rep. Rev 1.7, 2015. [Online]. Available: https://network.nvidia.com/related-docs/prod_software/RDMA_Aware_Programming_user_manual.pdf

[2] W. Gropp, E. Lusk, and A. Skjellum, Using MPI: Portable Parallel Programming with the Message Passing Interface. The MIT Press, 11 1999. [Online]. Available: https://doi.org/10.7551/mitpress/7056.001.0001

[3] "Nvidia collective communication library (nccl) documentation," Nvidia Corporation, Specification, 2020. [Online]. Available: https://docs.nvidia.com/deeplearning/sdk/nccl-developer-guide/docs/index.html

[4] A. Sergeev and M. Del Balso, "Horovod: fast and easy distributed deep learning in tensorflow," arXiv preprint arXiv:1802.05799, 2018.

[5] Horovod, "Horovod," https://github.com/horovod/horovod, 2017.

[6] "Intel NUC 9 Extreme/Pro Kit, Techincal Product Specification," Intel Corporation, Specification 1, Dec. 2019. [Online]. Available: https://simplynuc.com/wp-content/uploads/2020/01/NUC9QN TechProdSpec.pdf

[7] "Intel NUC 9 Kit, User Guide," Intel Corporation, Specification 1, May 2020. [Online]. Available: https://www.intel.com/content/dam/support/us/en/documents/intel-nuc/nuc-kits/NUC9xyQNX UserGuide.pdf

[8] "ConnectX-5 EN Card," Mellanox Technologies, Tech. Rep., 2020. [Online]. Available: https://www.mellanox.com/files/doc-2020/pb-connectx-5-en-card.pdf

[9] Z. Zhang, C. Chang, H. Lin, Y. Wang, R. Arora, and X. Jin, "Is network the bottleneck of distributed training?" in Proceedings of the Workshop on Network Meets AI & ML, 2020, pp. 8–13.

[10] J. George, "Qperf 0.4.11 (1)-linux man page," Qperf (1): Measure RDMA/IP Performance.

[11] D. Merkel et al., "Docker: lightweight linux containers for consistent development and deployment," Linux journal, vol. 2014, no. 239, p. 2, 2014.

[12] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," arXiv preprint arXiv:1603.04467, 2016.

[13] N. J. Gunther, Guerrilla Capacity Planning: A Tactical Approach to Planning for Highly Scalable Applications and Services, 1st ed. Springer Publishing Company, 2010.

[14] F. Alkhoury, D. Wegener, K.-H. Sylla, and M. Mock, "Communication efficient distributed learning of neu-ral networks in big data environments using spark," in 2021 IEEE International Conference on Big Data (Big Data), 2021, pp. 3871–3877.