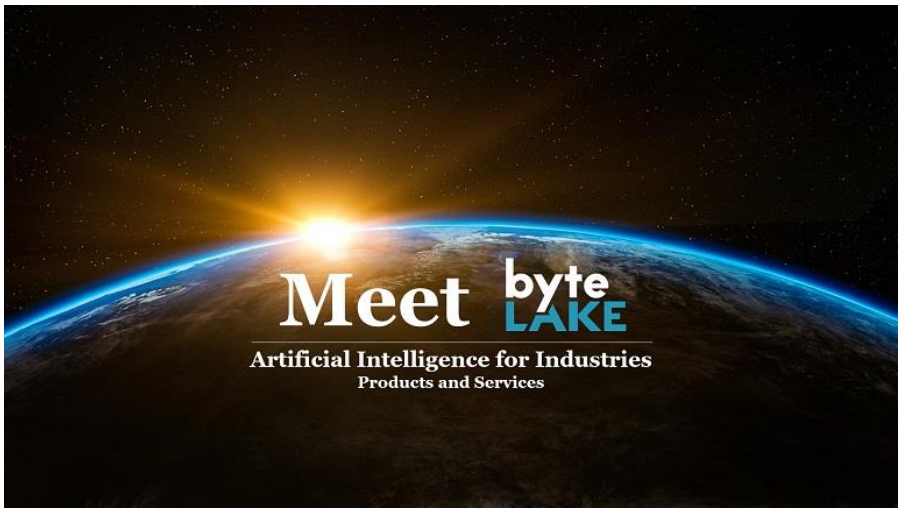


CFD Suite
(AI-accelerated CFD)

AI TRAINING BENCHMARK

This report summarizes byteLAKE's benchmark results of the recommended hardware platform for byteLAKE's CFD Suite's AI training at the Edge. Key takeaways are at the end of the report.

Artificial Intelligence

AI-accelerated CFD

AI-assisted Visual Inspection

AI-powered Big Data Analytics

Complex Tasks Automation

Chemical Industry

Paper Industry

Manufacturing

Industry 4.0

byteLAKE

Europe & USA

+48 508 091 885
+48 505 322 282
+1 650 735 2063

Foreword

Last year we published CFD Suite's scalability report. The goal was to address our clients' and partners' fundamental question of how CFD Suite's AI Training performs when deployed in a multi-node HPC (High-Performance Computing) configuration. The focus at that time was to help them understand the benefits of performing the AI Training on GPU-enabled architectures vs. CPU-only clusters which at that time were more common in the CFD (Computational Fluid Dynamics) industry. The conclusion was that: "we can assume that 4 nodes with Intel Xeon Gold CPUs will give comparable performance as a single NVIDIA V100 Tensor Core GPU node" and one can read more at byteLAKE's blog (link: https://marcrojek.medium.com/bytelakes-cfd-suite-ai-accelerated-cfd-hpc-scalability-report-25f9786e6123) or by downloading the full report from byteLAKE's website (link: www.byteLAKE.com/en/CFDSuite or directly at: https://www.bytelake.com/en/download/4013/).

As we progressed with the product development and related research efforts, we significantly changed the underlying AI architectures to better address our clients' and partners' needs. This effort led to improved accuracy of AI predictions but also better performance and increased automation. Previously, CFD Suite's users had to manually configure the number of iterations that the traditional CFD solver had to perform before AI could take over and generate the prediction. Now AI has taken over that task as well and the users no longer need to worry about how to properly calibrate CFD Suite for optimal results. That allowed us to implement a mechanism that could find the best tradeoff between performance and accurate predictions, ultimately improving the overall quality of predictions. Previously, CFD Suite's AI Training phase's performance could only be improved by adding more nodes within a multi-node HPC architecture. Thanks to our latest upgrade, it can now benefit from many NVIDIA GPU cards within a single node. A much-awaited feature for those who prefer such setups.

„We have significantly changed the architecture of the underlying AI within the CFD Suite. With the mechanisms like dynamic generating of the learning samples, we are now able to fully utilize multiple GPU cards within one node and provide better accuracy. Unlike in the previous versions, where CFD Suite's AI training performance could only be increased by adding more nodes. Now we can greatly benefit from having more accelerators within a single node.", said Krzysztof Rojek, DSc, PhD, CTO at byteLAKE.

Having said the above, let us warmly invite you to read another report about how we benchmarked the performance of CFD Suite's AI Training with a special focus on our partners and clients who prefer performing it on the edge server type of configurations. With that in mind, we have also selected a byteLAKE's recommended hardware platform, validated specifically for that purpose. Enjoy!

Marcin Rojek, Mariusz Kolanko, byteLAKE's co-founders.

Introduction

A few years ago, when we started extensive research at byteLAKE in the space of CFD simulations (Computational Fluid Dynamics), we naturally drifted towards experiments with various different hardware configurations. Our focus has always been on performance though. It is no different today although we think we might expand our research in the future towards other comparably exciting aspects and a broader context of how AI (Artificial Intelligence) can deliver value for industries performing CFD simulations. We have been describing these efforts and results including case studies in a blog post series which can be found here: www.byteLAKE.com/en/AI4CFD-toc. This report, however, contains byteLAKE's conclusions and recommendations about edge hardware configuration for the AI Training phase, required for the byteLAKE's CFD Suite to deliver its predictions and eventually significantly reduce time to results in CFD simulations.

What is byteLAKE's CFD Suite?

It is a Collection of innovative AI Models for Computational Fluid Dynamics (CFD) acceleration. It is a Deep Learning, data-driven solution which is currently available for the chemical industry, reducing mixing simulation time from hours to minutes. While one can read more about the product on byteLAKE's website at www.byteLAKE.com/en/CFDSuite or find us listed in various independent benchmarks i.e. Discover 5 Top Startups working on Computational Fluid Dynamics (startus-insights.com), CFD Suite works in the following way:

- first, it needs to be trained. We typically need some number of historic simulations to train the embedded AI models about the physical phenomenon, its parameters, corner cases, etc. The exact number of such historic simulations varies across phenomena but the rule of thumb is that 30–100 of such simulations is enough to be able to address various possible input configurations (speed of mixing, viscosity, pressure, etc.) and geometries;
- once the AI Training completes, CFD Suite can be used for inference purposes, meaning it can predict the results of the CFD simulations. CFD Suite starts by calling a traditional CFD solver first. CFD Suite analyzes its initial results and when its AI "feels" it is ready to predict, it takes over the simulation and generates its final result (steady-state). It does so within the seconds and including the overhead, CFD Suite has been able to reduce the time of chemical mixing simulations from 4–8 hrs. to 10–20 mins and keep the accuracy of predictions north from 93%.

Example results of the CFD Simulation performed by a traditional CFD Solver and CFD Suite are visible on the image below.

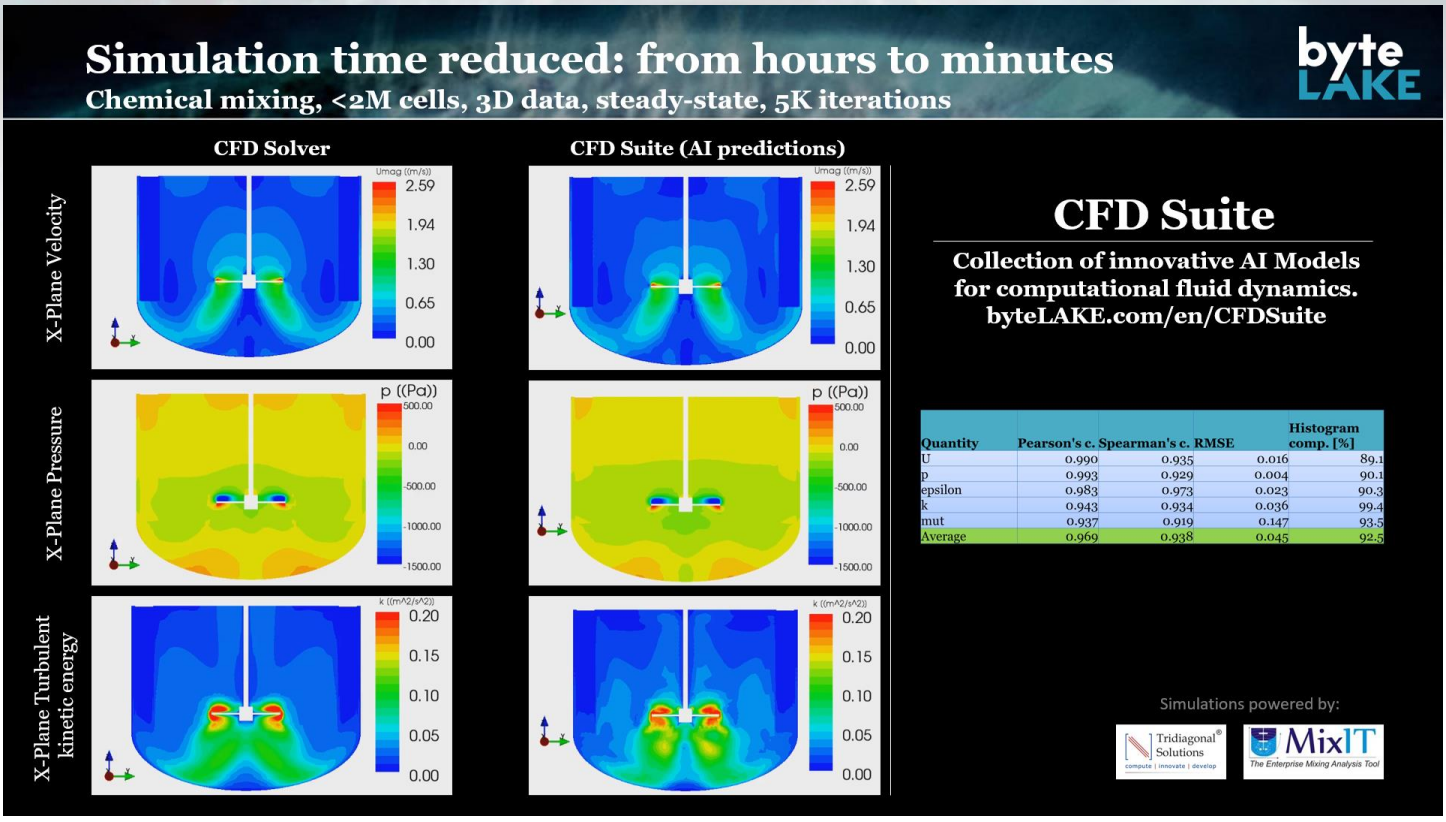


Figure 1. CFD simulation results vs. byteLAKE's CFD Suite's AI predictions.

Industry of focus and a reference hardware

Our current focus is on the chemical industry and how AI can deliver value there through the acceleration of various CFD workloads. While considering various hardware options to perform the AI Training part, we decided to start with an edge configuration that would meet the following criteria:

- **a standalone configuration** that does not require any server-related specific or advanced infrastructure (racks, server rooms) — something you just plug into a power outlet and is ready to work;
- **edge-type of the device** but something definitely stronger than laptops and definitely more flexible than desktop PCs (many CPUs, many GPUs, huge amount of RAM, large and fast storage);
- **ready for some of the most powerful GPUs** as the main workload here is the AI training. NVIDIA was our preferred vendor as we have had a great experience with almost all of their GPUs and APIs.

In addition, we wanted also:

- an optimal performance per value (installed at the edge, edge-sized, edge-priced);
- to ensure security to keep the data in-house without a need to send anything to the Cloud;
- to ensure no external data transfer and enable all computing locally;
- compatibility for a variety of communications ports for connection flexibility;
- a modular structure for easy hardware upgrades.

After all, we needed a single hardware option that we could recommend to the clients who need something more than a laptop and are not yet ready or might not need complex server configurations, not to mention the HPC (High-Performance Computing) as such. We did not want to consider Cloud options at that time but we might do so in the future as we expand our offering towards aaS (as-a-Service) options. Eventually we decided to pick Lenovo ThinkEdge SE450 Edge Server (Product Guide, Press Release) powered by 2 NVIDIA A100 80GB Tensor Core GPUs (Learn More).

We picked NVIDIA's A100 80GB variant to ensure that the cards are capable of handling relatively large mesh sizes of the CFD simulations during the AI training of the CFD Suite. Although we currently focus on simulations of around 5 million cells, we will definitely go north of that number in the nearest future.

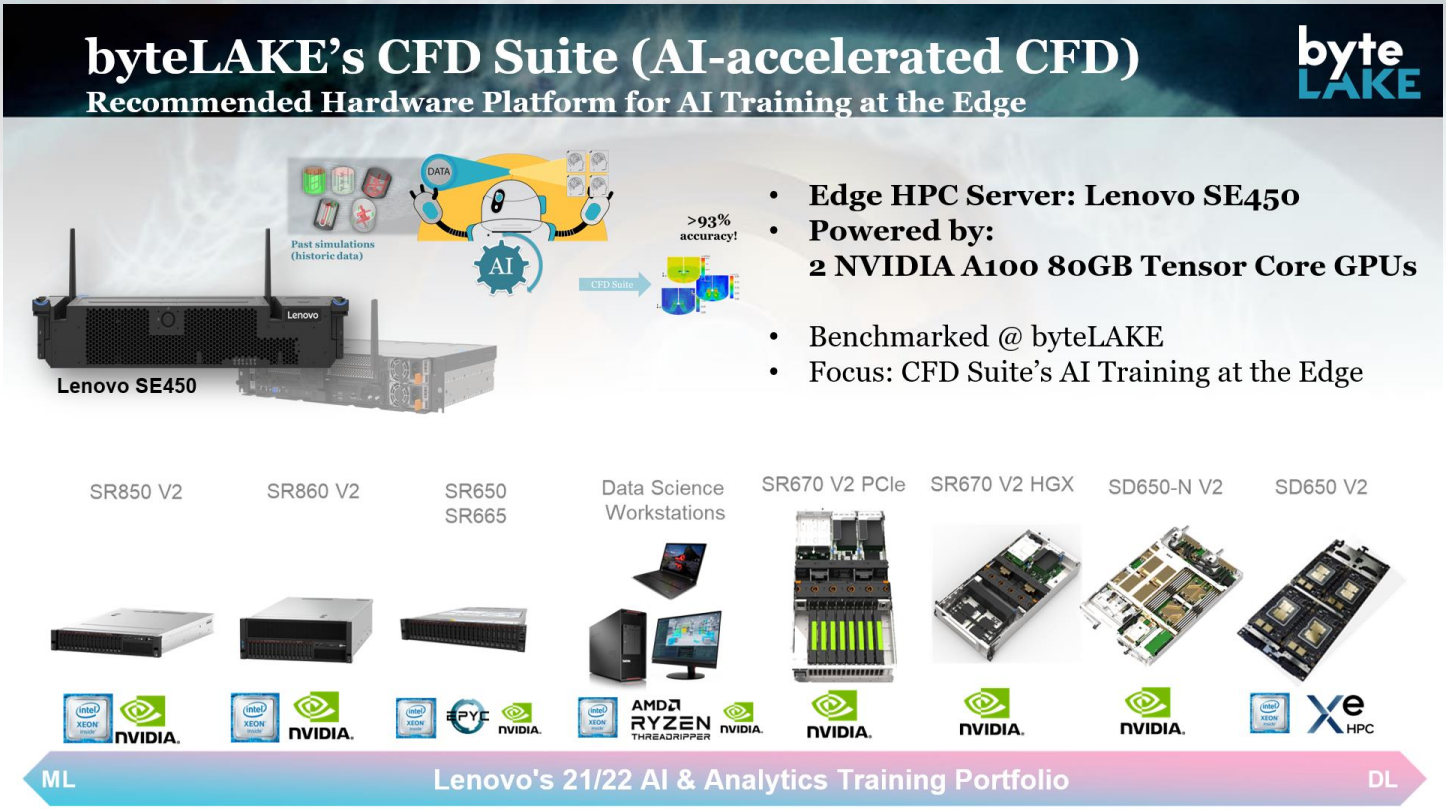


Figure 2. byteLAKE's CFD Suite (AI-accelerated CFD) — recommended hardware for AI training at the Edge

"We at byteLAKE picked Lenovo's SE450 edge HPC server as a recommended edge platform for the AI Training phase of the CFD Suite (AI-accelerated CFD) product. We love its design and flexibility and are convinced our clients in the chemical industry will appreciate how easily it enables the AI Training capabilities at the edge with NVIDIA GPUs", said Marcin Rojek, byteLAKE's Co-Founder.

Benchmark

We examined the performance and memory requirements of byteLAKE's CFD Suite that uses AI to accelerate CFD Simulations (hereinafter also referred to as a "**framework**"). CFD Suite is based on a data-driven model, where in the first step we need to train the model using CFD simulations executed with a traditional CFD solver (historic simulations build-up in a form of a training dataset). Then CFD Suite can provide the prediction that allows us to significantly reduce the simulation execution time. In this benchmark, we focus on the AI training part that requires highly parallel hardware to create an accurate model. To train the model, we used a real-life scenario, where our framework takes 10 initial iterations generated by the CFD solver as an input and returns the final (steady-state) iteration. Our dataset includes 50 such CFD simulations. From each simulation, the framework generates 20 different packages of inputs and a single iteration as an output. As a result, we use 1000 packages containing 10 input and 1 output iteration.

All the simulations are generated with the 3-dimensional rhoSimpleFoam solver. The rhoSimpleFoam is a steady-state solver for compressible, turbulent flow, using the SIMPLE (Semi-Implicit Method for Pressure Linked Equations) algorithm. That means that a pressure equation is solved and the density is related to the pressure via an equation of state. We generate 3 different mesh configurations:

- the mesh of size 32 768 cells;
- the mesh of size 262 144 cells;
- the mesh of size 884 736 cells.

The training of our model generates a set of sub-models for each quantity used by the solver. We use here 2 types of quantities: scalar quantities (pressure, temperature, …), and the vector quantity (velocity). Our model trains them sequentially one by one, so for the purpose of this benchmark, we focus on analyzing a single scalar and vector quantity.

Hardware and software environment

This benchmark has been executed on the Lenovo SE450 node. The node is equipped with a single Intel Xeon Gold CPU and 2xNVIDIA A100 GPUs. Moreover, the performance results are compared with a single node equipped with 2xNVIDIA V100 GPUs. The server node includes 128GB of the host memory. All the platforms that we have benchmarked will be further referred to as:

- **A100** – NVIDIA A100 GPU with 80GB of GPU global memory;
- **V100** – NVIDIA V100 GPU with 16GB of GPU global memory;
- **CPU or Gold** - Intel Xeon Gold 6330N CPU clocked 2.20GHz with 28 physical (56 logical) cores.

The software environment of the SE450 node includes:

- Ubuntu 20.04.4 LTS (GNU/Linux 5.13.0-51-generic x86_64) OS;

- NVIDIA Driver version: 510.73.05;
- CUDA Version: 11.6;
- cuDNN version: 8.4.0;
- TensorFlow version: 2.9.0;
- Keras: the Python deep learning API version: 2.9.1;
- Dataset: float32 data type (single-precision arithmetic).

A100 performance results and host memory requirements for a scalar quantity

In the table below we included the performance results for the scalar quantity training. The performance results include a different batch (the number of samples that are propagated through the network) and mesh size. We examined here the host memory requirements, and execution time using a single A100 GPU and 2xA100 GPUs. To train the network we used 1000 epochs. We have also included here the average execution time per epoch and a speedup of 2xA100 over 1xA100 GPU.

			1xA100		2xA100		
batch	mesh [cells]	host memory [GiB]	time [s]	time/epoch [s]	time [s]	time/epoch [s]	Speedup
1	32768	9.7	19020	19.02	10316	10.32	1.84
2	32768	9.7	10021	10.02	5634	5.63	1.78
4	32768	9.8	6025	6.03	3428	3.43	1.76
8	32768	10.0	3028	3.03	1618	1.62	1.87
16	32768	11.0	2340	2.34	1316	1.32	1.78
32	32768	12.0	1460	1.46	791	0.79	1.85
64	32768	16.0	1750	1.75	982	0.98	1.78
128	32768	25.0	1126	1.13	628	0.63	1.79
1	262144	30.0	22023	22.02	11809	11.81	1.86
2	262144	31.0	14029	14.03	7879	7.88	1.78
4	262144	33.0	9037	9.04	4790	4.79	1.89
8	262144	36.0	6053	6.05	3258	3.26	1.86
16	262144	41.0	5910	5.91	3303	3.30	1.79
32	262144	52.0	5166	5.17	2917	2.92	1.77
64	262144	67.0	5332	5.33	2927	2.93	1.82
128	262144	109.0	5634	5.63	3093	3.09	1.82
1	884736	89.0	30031	30.03	16901	16.90	1.78
2	884736	92.0	22046	22.05	12165	12.16	1.81
4	884736	96.0	18074	18.07	9818	9.82	1.84
8	884736	104.0	17142	17.14	9620	9.62	1.78
16	884736	116.0	16278	16.28	9184	9.18	1.77

Table 1. Performance results for the scalar quantity training.

The best performance results for each mesh are included in the table below:

			1xA100		2xA100		
batch	mesh [cells]	host memory [GiB]	time [s]	time/epoch [s]	time [s]	time/epoch [s]	Speedup
128	32768	25	1126	1.13	628	0.63	1.79
32	262144	52	5166	5.17	2917	2.92	1.77
16	884736	116	16278	16.28	9184	9.18	1.77

Table 2. Best performance results for each mesh (scalar quantity training).

The performance results for a single A100 are plotted in the figure below (the lower the better):

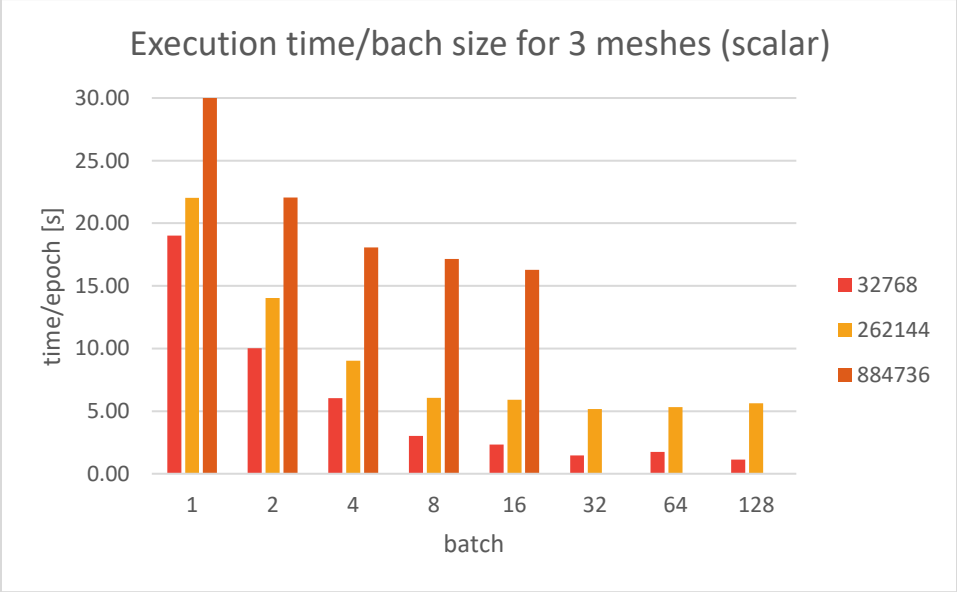


Figure 3. A100 performance results (scalar quantity training).

The performance results for 2 x A100 are plotted in the figure below (the lower the better):

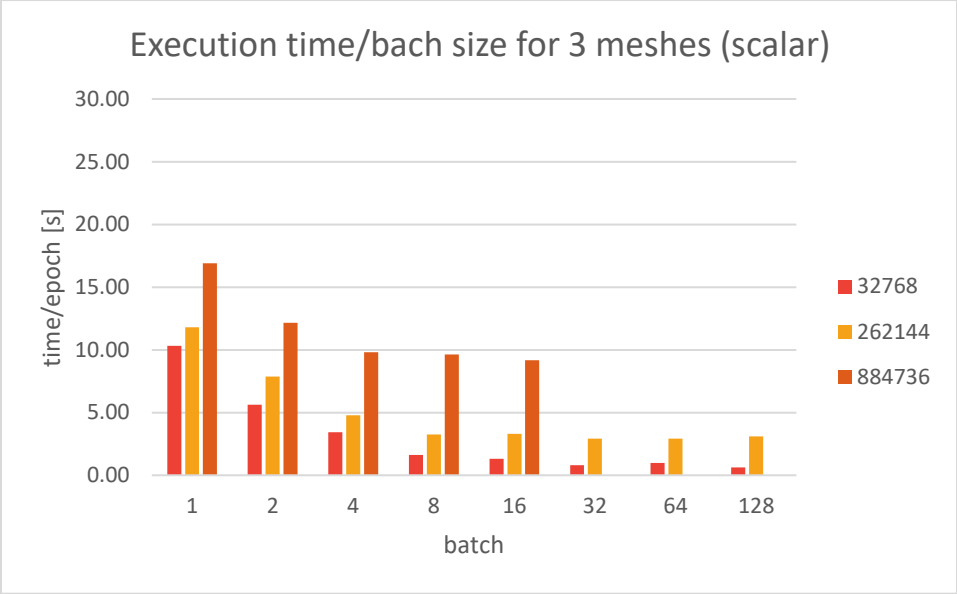


Figure 4. Two A100 performance results (scalar quantity training).

The host memory requirements depending on a batch size are listed below (the lower the better):

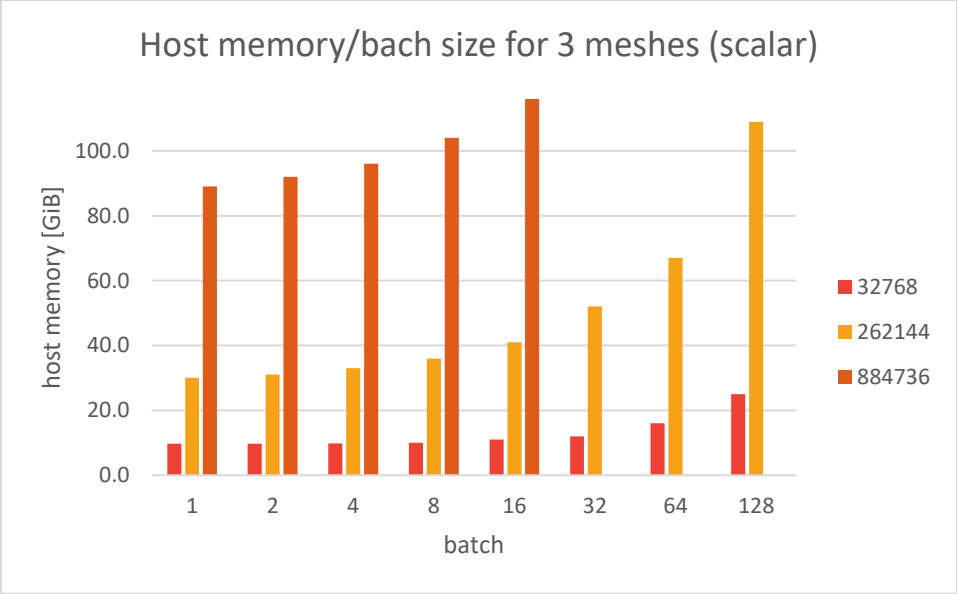


Figure 5. Host memory requirements for various batch sizes (scalar quantity training).

The performance comparison for each mesh between 1x and 2x A100 is plotted below (the lower the better):

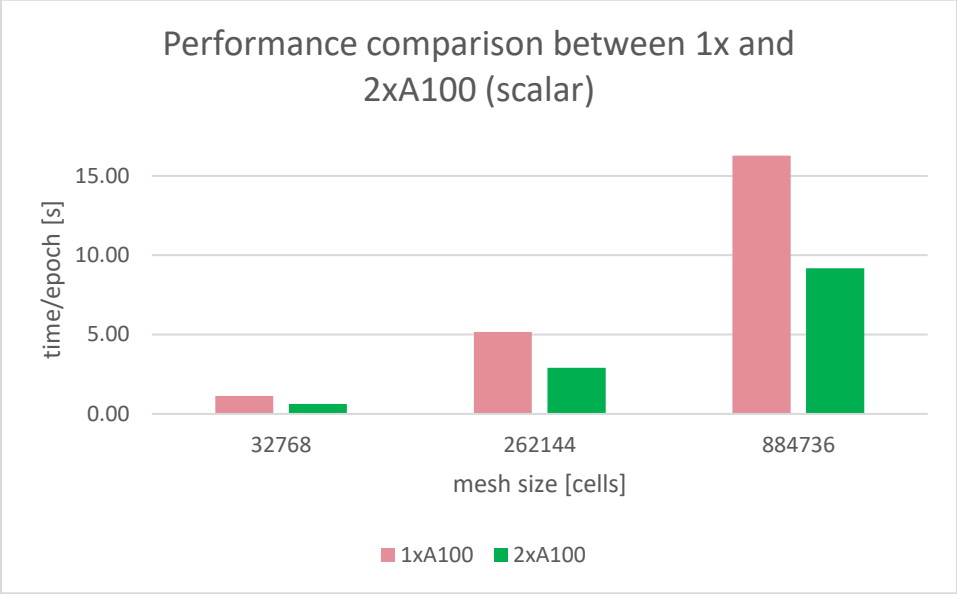


Figure 6. Performance comparison: one vs. two A100 GPUs (scalar quantity training).

Conclusions:

- **The speedup between 1xA100 and 2xA100 is stable for all 3 meshes and varies from 1.76 to 1.89.**
- **It gives the efficiency of up to 0.95 which shows good scalability of the framework.**
- The best performance is achieved using a batch of sizes 128, 32, and 16 for meshes of sizes 32768, 262144, and 884736, respectively.
- We observe that memory requirements increase when the batch increases and for the mesh of size 884 736, the batch >16 exceeds the available host memory.

A100 performance results and host memory requirements for a vector quantity

In the table below, we included the performance results of the vector quantity training. The performance results include a different batch and mesh size. We examined here the host memory requirements, and execution time using a single A100 GPU and 2xA100 GPUs. As before, to train the network we used 1000 epochs. We have also included here the average execution time per epoch and speedup of 2xA100 over 1xA100 GPU.

			1xA100		2xA100		
batch	mesh [cells]	host memory [GiB]	time [s]	time/epoch [s]	time [s]	time/epoch [s]	Speedup
1	32768	15	20021	20.02	10980	10.98	1.82
2	32768	15	12025	12.03	6825	6.83	1.76
4	32768	16	7028	7.03	3884	3.88	1.81
8	32768	17	4033	4.03	2238	2.24	1.80
16	32768	17	3048	3.05	1736	1.74	1.76
32	32768	23	2076	2.08	1140	1.14	1.82
64	32768	28	2141	2.14	1162	1.16	1.84
128	32768	48	2315	2.32	1235	1.24	1.87
1	262144	79	28030	28.03	15697	15.70	1.79
2	262144	82	19040	19.04	10552	10.55	1.80
4	262144	87	16067	16.07	8940	8.94	1.80
8	262144	93	15125	15.13	8425	8.42	1.80
16	262144	109	14237	14.24	8072	8.07	1.76

Table 3. Performance results for the vector quantity training

The best performance results for each mesh are included in the table below:

			1xA100		2xA100		
batch	mesh [cells]	host memory [GiB]	time [s]	time/epoch [s]	time [s]	time/epoch [s]	Speedup
32	32768	23	2076	2.08	1140	1.14	1.82
16	262144	109	14237	14.24	8072	8.07	1.76

Table 4. Best performance results for each mesh (vector quantity training)

The performance results for a single A100 are plotted in the figure below (the lower the better):

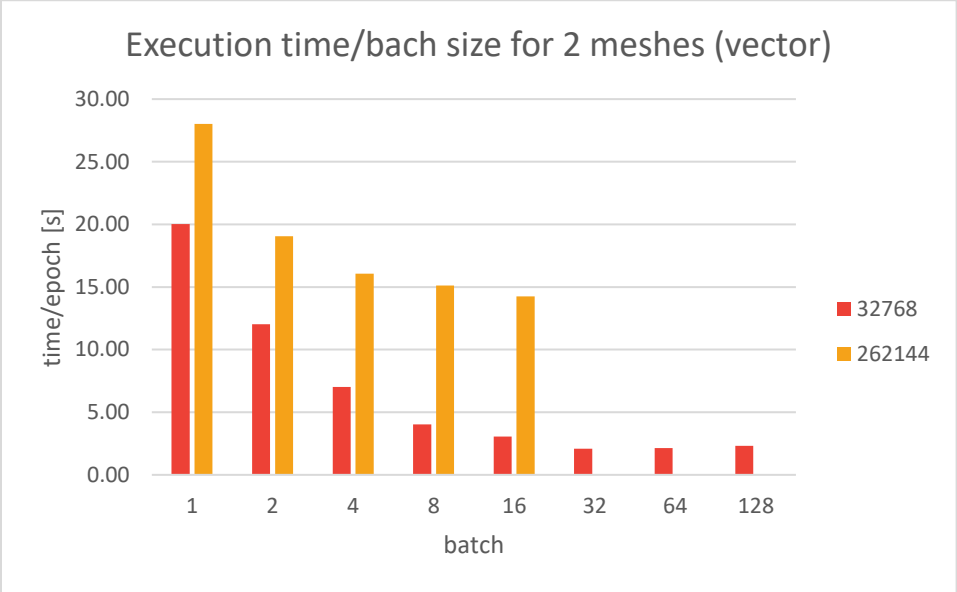


Figure 7. A100 performance results (vector quantity training)

The performance results for 2 x A100 are plotted in the figure below (the lower the better):

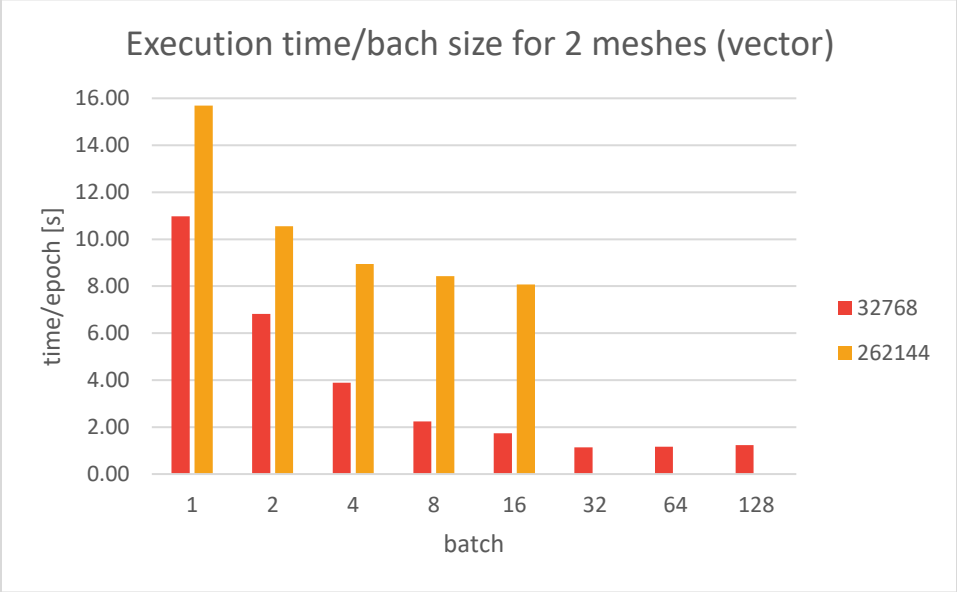


Figure 8. Two A100 performance results (vector quantity training)

The host memory requirements depending on a batch size are listed below (the lower the better):

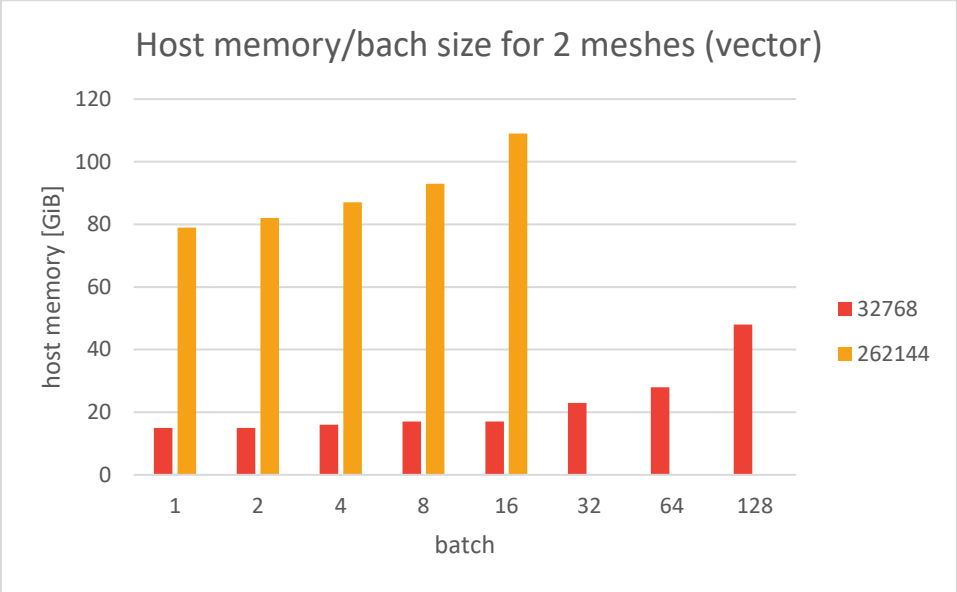


Figure 9. Host memory requirements for various batch sizes (vector quantity training).

The performance comparison for each mesh between 1x and 2x A100 is plotted below (the lower the better):

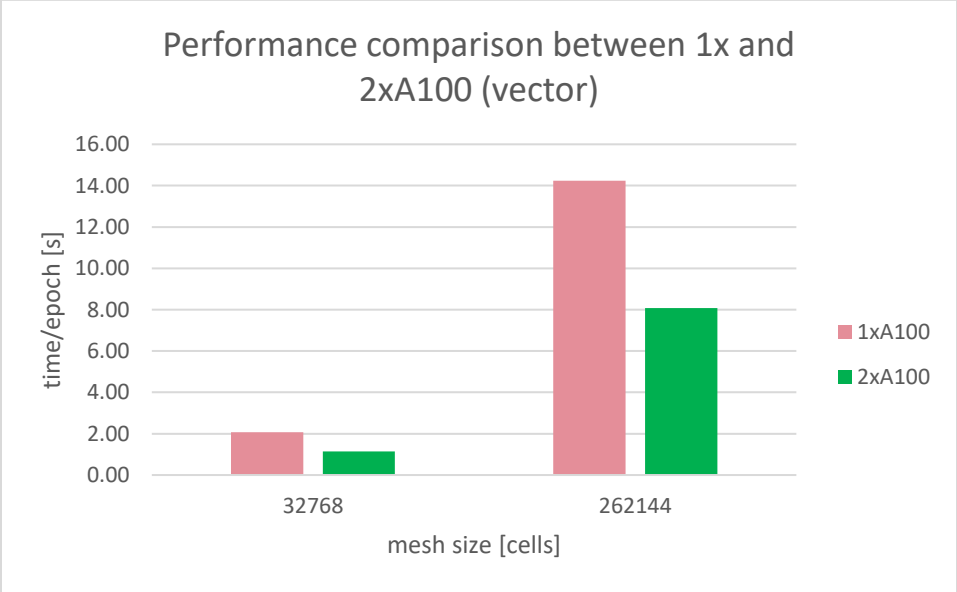


Figure 10. Performance comparison: one vs. two A100 GPUs (vector quantity training).

Conclusions:

- **The speedup between 1xA100 and 2xA100 is stable for all 3 meshes and varies from 1.76 to 1.87.**
- **It gives the efficiency up to 0.95 which confirms good scalability of the framework regardless of the type of quantity (scalar, vector).**
- The best performance is achieved using a batch of sizes 32, and 16 for meshes of sizes 32768, and 262144, respectively.
- The tested scenario requires more than 128GB of the host memory to train the network for the mesh of size 884736 with the vector quantity.

- Comparing this benchmark with our previous one (available here: https://marcrojek.medium.com/bytelakes-cfd-suite-ai-accelerated-cfd-hpc-scalability-report-25f9786e6123) or by downloading the full report from byteLAKE's website www.byteLAKE.com/en/CFDSuite here: https://www.bytelake.com/en/download/4013/), we observe that in the current version of the byteLAKE's CFD Suite the **scalability within a node is much more profitable**. This has resulted from the fact, that the current AI model is much more compute-intensive - includes more layers. **In this version of our framework, we provided a mechanism that dynamically generates a set of training samples from a single input simulation, which also improves the dataset size and reduces the memory transfer**.

Comparison of performance and memory requirements between vector and scalar quantities

The table below contains the host memory requirements between vector and scalar quantity depending on batch size. This comparison is performed for a mesh of size 32 768 cells.

	host memory [GiB]		
batch	scalar	vector	Ratio
1	9.7	15.0	1.55
2	9.7	15.0	1.55
4	9.8	16.0	1.63
8	10.0	17.0	1.70
16	11.0	17.0	1.55
32	12.0	23.0	1.92
64	16.0	28.0	1.75
128	25.0	48.0	1.92

Table 5. Host memory requirements. Mesh size: 32 768 cells.

The table below contains the host memory requirements between vector and scalar quantity depending on batch size. This comparison is performed for a mesh of size 262 144 cells.

	host memory [GiB]		
batch	scalar	vector	Ratio
1	30.0	79.0	2.63
2	31.0	82.0	2.65
4	33.0	87.0	2.64
8	36.0	93.0	2.58
16	41.0	109.0	2.66
32	52.0		
64	67.0		
128	109.0		

Table 6. Host memory requirements. Mesh size: 262 144 cells.

The performance comparison between the vector and scalar quantities is shown in the figure below (the lower the better):

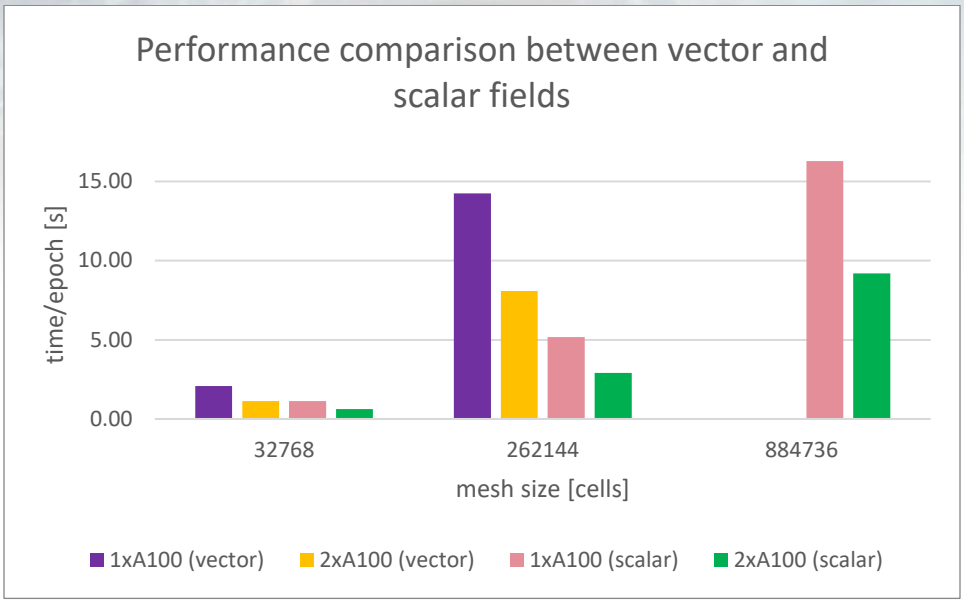


Figure 11. AI Training performance for various mesh sizes (vector and scalar @ 1*A100 vs. 2*A100)

The host memory requirements for a scenario with the vector and scalar quantities are shown in the figure below. The results include the mesh of size 32768 cells (the lower the better).

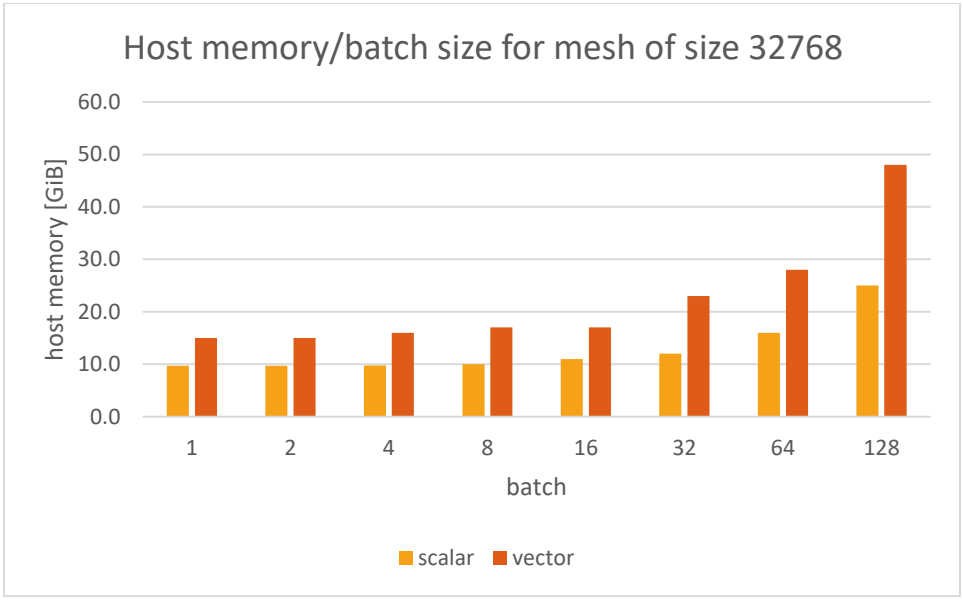


Figure 12. Host memory requirements (vector, scalar) for AI Training. Mesh size: 32 768.

The host memory requirements for a scenario with the vector and scalar quantities are shown in the figure below. The results include the mesh of size 262144 cells (the lower the better).

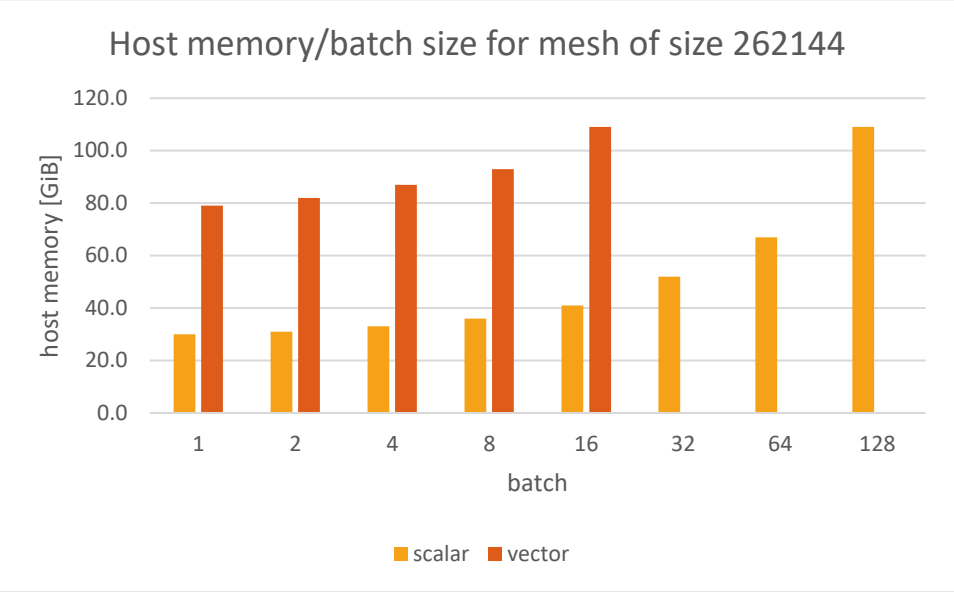


Figure 13. Host memory requirements (vector, scalar) for AI Training. Mesh size 262 144.

Conclusions:

- **Both, scalar and vector quantities, are efficiently distributed across 2 x A100 GPUs with an efficiency of up to 0.95.**
- Vector quantity is executed from 1.81x to 2.77x slower than the scalar quantity, depending on the mesh size.

- The memory requirements of the vector quantity are from 1.55x to 1.92x higher than the scalar quantity for the mesh of size 32768 cells.
- The memory requirements of the vector quantity are more than 2.6x than of the scalar quantity for the mesh of size 262144 cells.

- Based on byteLAKE's case study, the full AI Training, assuming a single vector quantity, and 6 scalar quantities takes ~1h36min (1140s+791s*6) for a mesh of size 32 768, ~7h6min (8072s+2917s*6) for a mesh of size 262 144, and approximately up to 24h (30000s+9184s*6) for a mesh of size 884 736 (the execution time of the vector quantity is approximated based on other results) using 2 x A100 GPUs.

Platforms comparison: A100 vs. V100 vs. CPU

In this section, we compare the performance results of A100 GPUs with other platforms (V100 GPU, Gold CPU). **The first experiment allowed us to determine the best batch size for each platform.** The results are listed in the table below.

batch	CPU [s]	V100 [s]	Speedup (V100 vs CPU)	A100 [s]	Speedup (A100 vs CPU)
1	121.13	15.02	**8.07**	22.02	**5.50**
2	95.20	11.02	**8.64**	14.03	**6.79**
4	85.36	12.05	**7.08**	9.04	**9.45**
8	71.59	11.09	**6.46**	6.05	**11.83**
16	61.00	10.17	**6.00**	5.91	**10.32**
32	60.00	9.31	**6.44**	5.17	**11.61**
64	56.00	13.90	**4.03**	5.33	**10.50**
128	51.01	13.00	**3.92**	5.63	**9.05**

Table 7. Best batch size for each platform: Gold CPU, V100, A100

In the figure below we plot the performance comparison between A100, V100, and CPU gold depending on the batch size (the lower the better). In the red rectangles, we marked the best batch size for each platform.

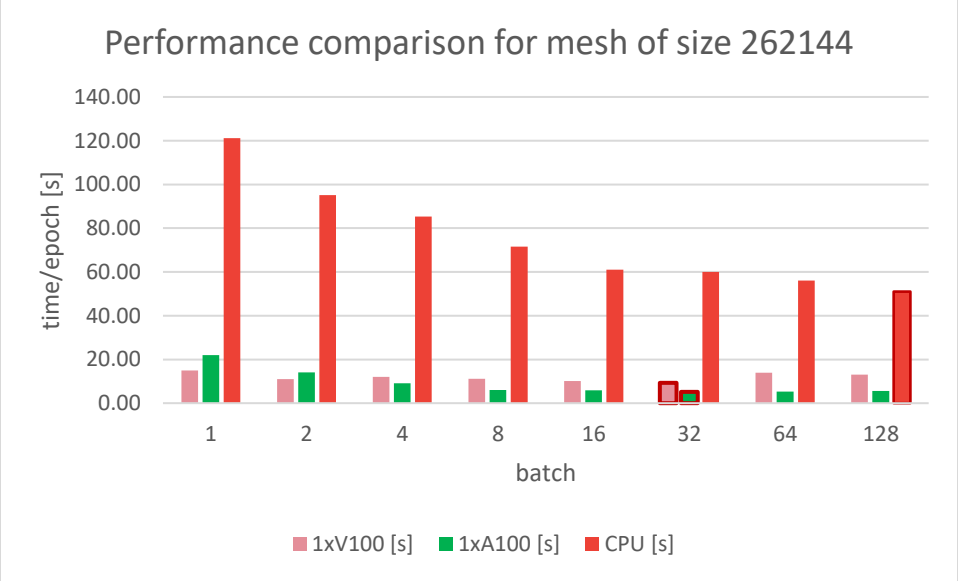


Figure 14. Performance results

We selected the best batch size for each platform and each mesh size and compared the performance results. In other words, we compared the performance of each platform's best configuration. The table below includes the platform name, selected batch size, and mesh size end execution time for each test. It also contains speedups of the GPUs over the CPU execution.

			1xGPU			2xGPU		
platform	batch	mesh [cells]	time [s]	time/epoch [s]	Speedup	time [s]	time/epoch [s]	Speedup
A100	128	32768	1126	1.13	**5.91**	628	0.63	**10.60**
V100	16	32768	2031	2.03	**3.28**	1095	1.09	**6.08**
Gold	128	32768	6658	6.66	**1**			
A100	32	262144	5166	5.17	**9.87**	2917	2.92	**17.49**
V100	32	262144	9312	9.31	**5.48**	4967	4.97	**10.27**
Gold	128	262144	51006	51.01	**1**			
A100	16	884736	16278	16.28	**10.87**	9184	9.18	**19.27**
V100	4	884736	29121	29.12	**6.08**	16576	16.58	**10.68**
Gold	128	884736	177022	177.02	**1**			

Table 8. Performance results.

The performance comparison between A100, V100, and CPU gold for a mesh of size 32768 is listed below (the lower the better):

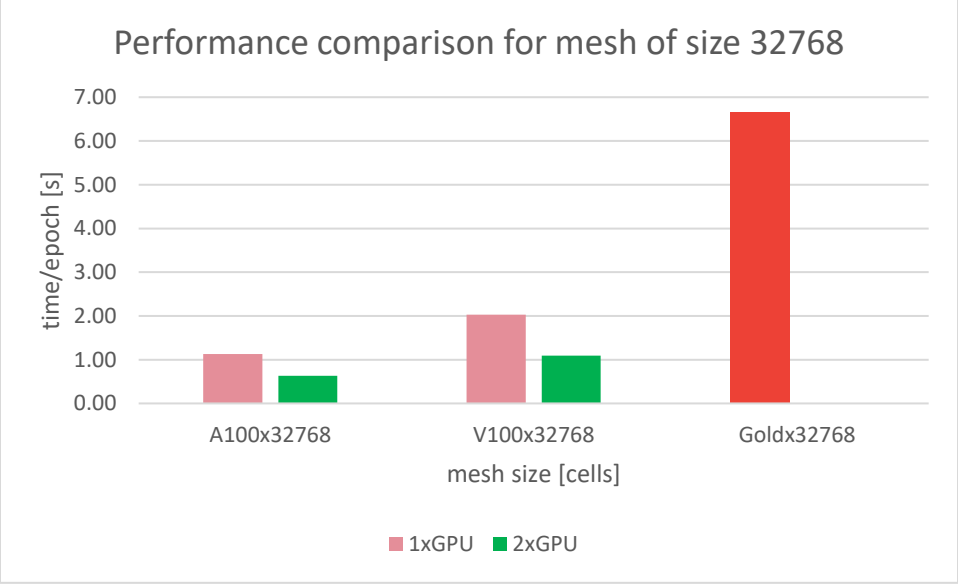


Figure 15. Performance comparison. Mesh size: 32 768.

The performance comparison between A100, V100, and CPU gold for a mesh of size 262144 is listed below (the lower the better):

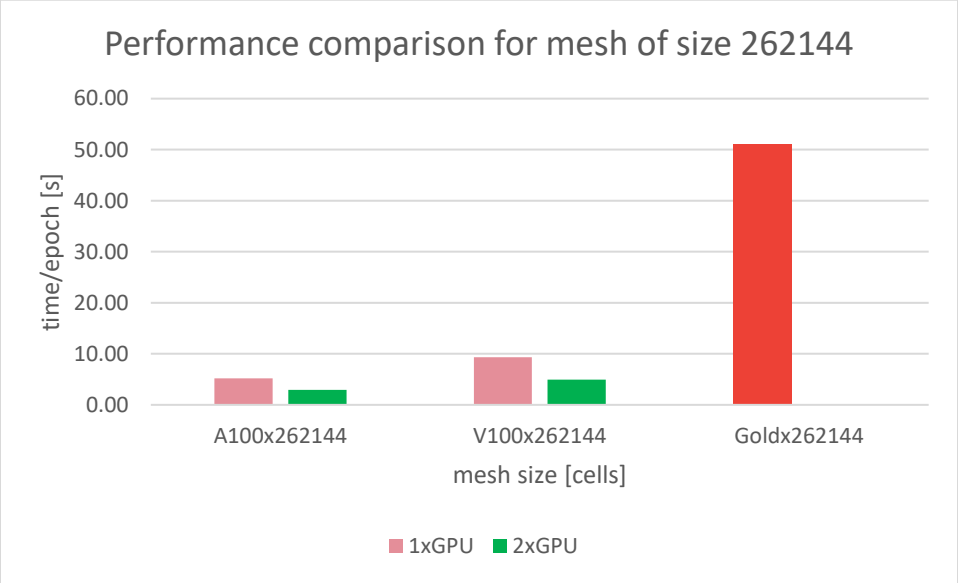


Figure 16. Performance comparison. Mesh size: 262 144.

The performance comparison between A100, V100, and CPU gold for a mesh of size 884736 is listed below (the lower the better):

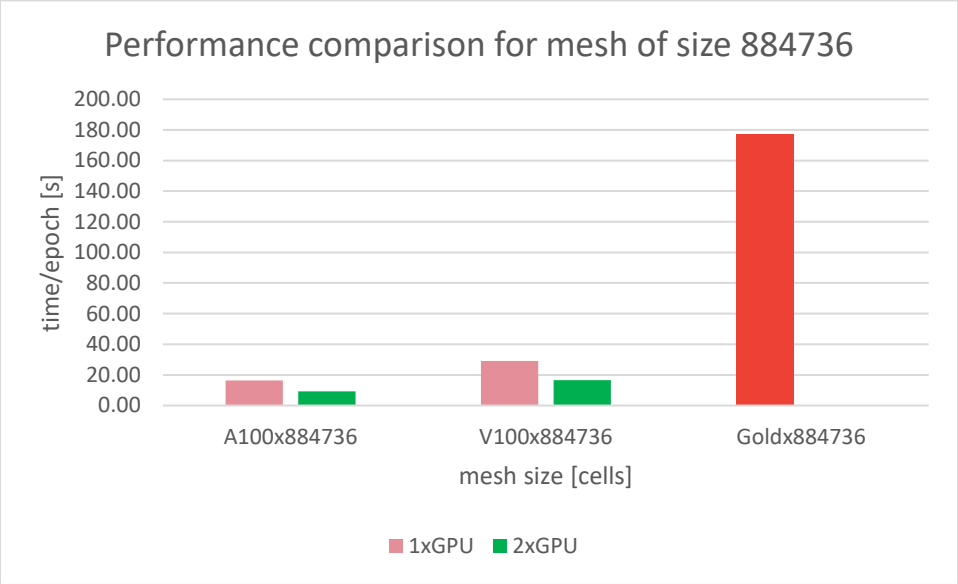


Figure 17. Performance comparison. Mesh size: 884 736.

Conclusions:

- Different platforms used individually optimized configurations for each mesh size to measure the performance of byteLAKE's CFD Suite.
- For some batch sizes (size 1 and 2 – see Table 7) the V100 GPU outperforms the A100 (1.5x speedup for a batch of size 1), which is resulted from the facts, that:
    - A100 requires a higher batch size than V100 to fully utilize the GPU compute resources;
    - A100 has a higher number of SMs (Streaming Multiprocessors) than V100, so the number of parallel tasks to utilize the resources needs to be higher;

  The base frequency of V100 is higher than of A100, so for a low GPU utilization V100 outperforms A100.
- **Overall, with the optimized configs, the A100 GPU is ~1.8x faster than V100 GPU.**
- With the optimized configs, the A100 GPU gives a speedup from 5.9x to 10.9x compared to the CPU gold, while the V100 is from 3.3x to 6x faster. This is as expected and aligned with our previous benchmarks where we concluded that GPUs were preferred for AI Training workloads.
- **Using an entire SE450 node, 2 x A100 gives from 10.6x to 19.3x speedup over CPU gold, while the V100 is from 6x to 10.7x faster than CPU gold.**
- **The higher mesh the higher speedup is achieved using GPUs over the CPU platform.**

Memory limits

The goal of the final test was to measure the maximum mesh that could be trained using A100 80GB GPU with our framework. To perform this test the following assumptions were made:

- The host memory requirements are not taken into account, since we generated an artificial dataset for GPU purposes only;
- We generate a single CFD simulation, for which 20 input packages were generated;
- We used a single A100 GPU (parallelization using 2xGPUs does not enable sharing a single batch, so it does not make it possible to train with larger meshes);
- We executed as big as possible mesh up to reducing the batch size from 8 to 1.

The results are listed in the table below:

mesh [cells]	batch	execution
11239424	8	OK
16777216	8	OOM
16777216	4	OK
21952000	4/2	OOM
21952000	1	OK

Table 9. Maximum mesh size test. OOM = out of memory.

Conclusions:

- In theory, we are able to train the model for a mesh of size 21 952 000 using A100 80GB GPU.
- With a batch of size 4 we can train a model for a mesh of size 16 777 216.
- With a batch of size 8 we can train the model for a mesh of size 11 239 424.

Key takeaways

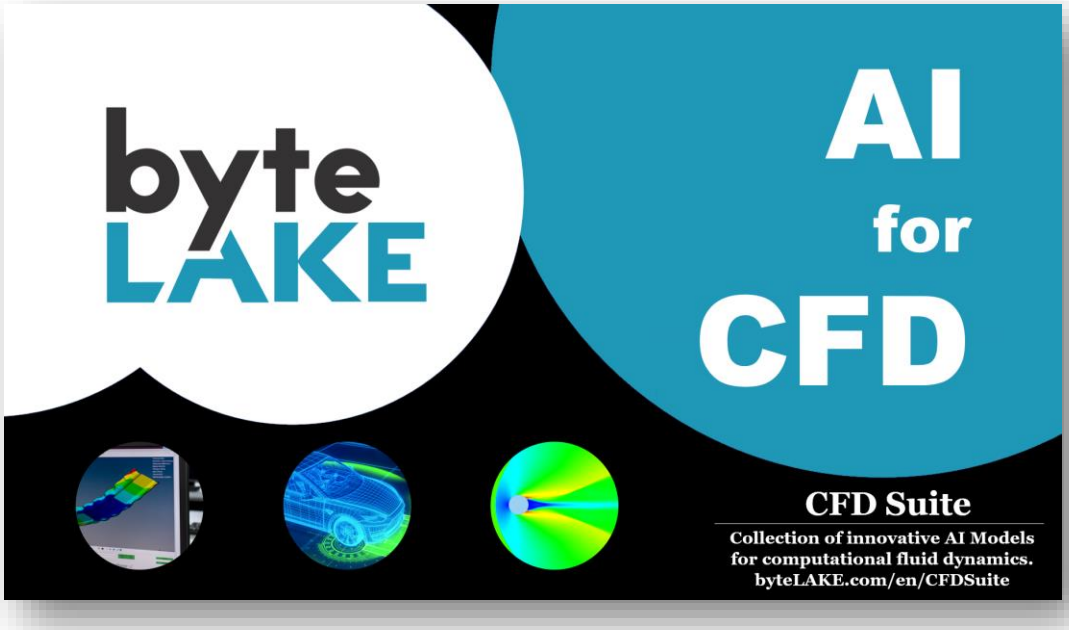
- **Lenovo ThinkEdge SE450 Edge Server (Product Guide, Press Release) powered by 2 NVIDIA A100 80GB GPUs (Learn More) is byteLAKE's recommended hardware configuration to perform CFD Suite's AI Training at the Edge.**

- **A100 GPU turned out to be ~1.8x faster than V100 GPU** in the scenarios benchmarked by byteLAKE and described in this report.
- **CFD Suite's AI Training's performance improves if we add more NVIDIA GPUs per node. The speedup between 1xA100 and 2xA100 was stable for all benchmarked meshes and varied from 1.76 to 1.87.**

- **Efficiency of the AI Training was 0.95 which confirmed the good scalability of CFD Suite.**

- SE450 node, powered by 2 x A100 gave from 10.6x to 19.3x speedup over CPU gold, while the V100 was from 6x to 10.7x faster than CPU gold. The higher the mesh size, the higher speedup was achieved using GPUs over the CPU platform. Again, the results are based on scenarios described in this report.

- In theory, we are able to train the model for a mesh of size 21 952 000 using a single A100 80GB GPU. This is based on byteLAKE's CFD Suite's current architecture and as a research in that space is ongoing, this will change in the future.



byteLAKE

Artificial Intelligence for Chemical Industry, Paper Industry and Manufacturing.

We build AI products and help design custom AI software.

About byteLAKE

byteLAKE is a software company that builds Artificial Intelligence products for the chemical industry, paper industry and manufacturing. byteLAKE's CFD Suite leverages AI to reduce CFD (Computational Fluid Dynamics) chemical mixing simulations' time from hours to minutes. byteLAKE's Cognitive Services offer AI-assisted Visual Inspection and Big Data analytics. For the paper industry, it can detect and visually inspect the so-called Water Line. For manufacturing, it helps automate complex tasks related to visual quality inspection and perform sound analytics helping efficiently detect and identify faulty engines, bearings, etc. The company also offers custom AI software development for real-time data analytics (image / video / sound / time-series). To learn more about byteLAKE's innovations, go to www.byteLAKE.com.