Credits: AI generated image; https://deepai.org/machine-learning-model/text2img

Performance and Power: Systematic Evaluation of AI Workloads on Accelerators with CARAML

Performance, Portability, and Productivity in HPC Workshop (SC24)

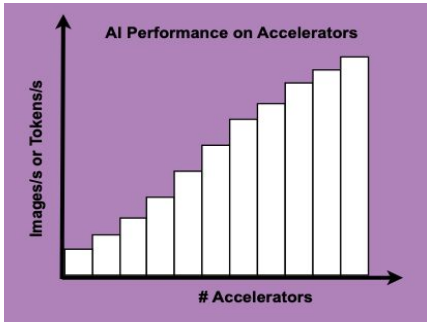
18.11.2024 I Chelsea Maria John | Forschungszentrum Jülich, Jülich Supercomputing Centre

CARAML BENCHMARK SUITE

Introduction

- **Motivation:** Automated, reproducible AI hardware assessment; quickly determine strengths
- **C**ompact **A**utomated **R**eproducible **A**ssessment of **M**achine **L**earning workloads (CARAML[1]) on novel accelerators
- Compact, automation and reproducibility using
  - **JUBE**[2] benchmarking environment
  - **Apptainer** containers
- AI Performance assessment with
  - **PyTorch** with torch.distributed
  - **TensorFlow** with Horovod
- Power measurement using **jpwr**[3]

JUBE BENCHMARKING ENVIRONMENT + PyTorch TensorFlow + jpwr



CARAML BENCHMARK SUITE

Introduction

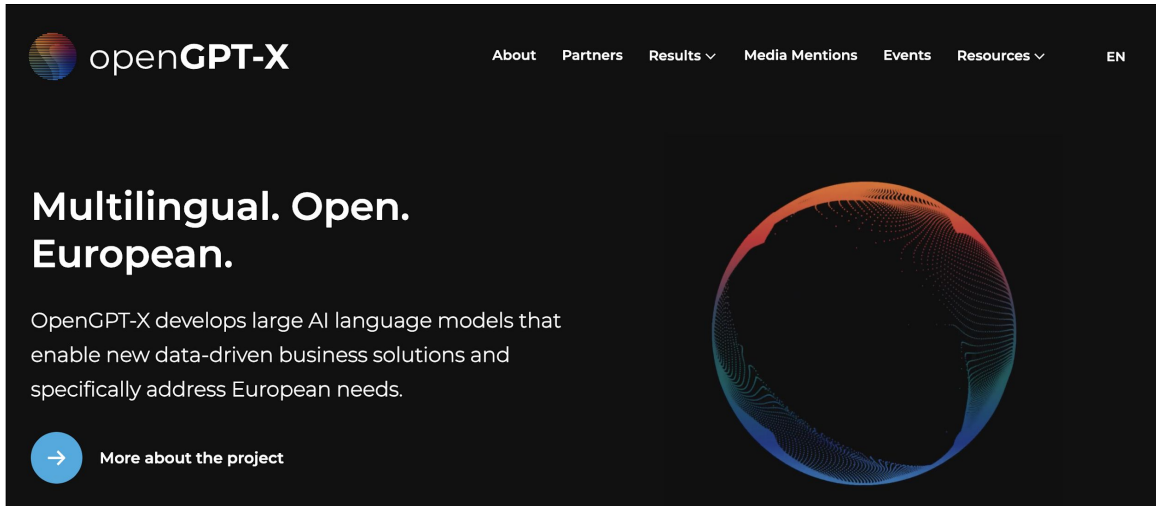
- **Motivation:** Automated, reproducible AI hardware assessment; quickly determine strengths
- **C**ompact **A**utomated **R**eproducible **A**ssessment of **M**achine **L**earning workloads (CARAML[1]) on novel accelerators
- Compact, automation and reproducibility using
  - **JUBE**[2] benchmarking environment
  - **Apptainer** containers
- AI Performance assessment with
  - **PyTorch** with torch.distributed
  - **TensorFlow** with Horovod
- Power measurement using **jpwr**[3]

Open Source Frameworks:
[1]: https://github.com/FZJ-JSC/CARAML
[2]: https://github.com/FZJ-JSC/JUBE
[3]: https://github.com/FZJ-JSC/jpwr

CARAML BENCHMARK SUITE

Natural Language Processing (NLP) Benchmark

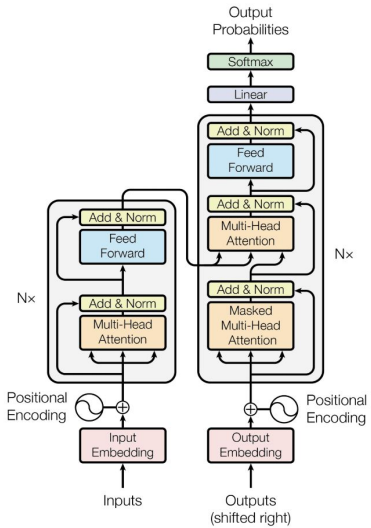
- GPT-based LLM training (OpenGPT-X: https://opengpt-x.de/en/)



CARAML BENCHMARK SUITE

Natural Language Processing (NLP) Benchmark

- GPT-based LLM training (OpenGPT-X: https://opengpt-x.de/en/)

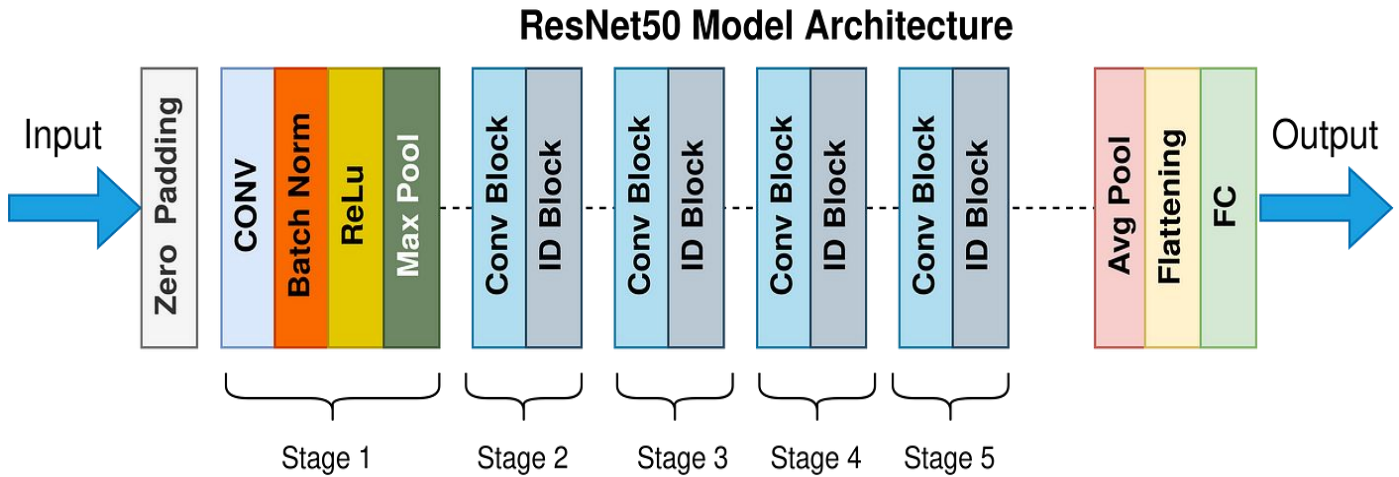


GPT Model architecture: Generative Pre-trained Transformer model uses stack of **encoders/decoders** to process data, weighted by **attention** AttentionIsAllYouNeed:https://arxiv.org/abs/1706.03762

CARAML BENCHMARK SUITE

Computer Vision (CV) Benchmark

- ResNet50 model training



ResNet50: Residual Networks (ResNet) with 50 Convolutional Neural Network layers used for image classification tasks, https://commons.wikimedia.org/wiki/ResNet50

CARAML BENCHMARK SUITE

Benchmark Models

[4]:https://huggingface.co/bigscience/oscar-1GB.jsonl.xz
[5]: https://image-net.org/download

Benchmark	NLP Training	CV Training
Model	GPT-based LLM model training on OSCAR[4] / synthetic data	ResNet50 model training on ImageNet[5] / Synthetic data
Performance Metrics	• Tokens/s • Energy/GPU • Tokens/Energy	• Images/s • Energy/Epoch (1 Epoch: 10^6 images) • Images/Energy
Reference Codes	• NVIDIA/Megatron-LM • Bigcode-project/Megatron-LM (ported for AMD) • Graphcore/examples/nlp • FZJ-JSC/jubench-megatron-lm	• Tensorflow/benchmarks • Graphcore/examples/vision • FZJ-JSC/jubench-resnet

ACCELERATORS

JURECA, JEDI & WestAI Systems

[6]: https://apps.fz-juelich.de/jsc/hps/jureca
[7]: https://apps.fz-juelich.de/jsc/hps/jedi
[8]: https://westai.de/

Platform	Accelerator (Acc)	CPU	CPU - ACC Interconnect	Acc - Acc Intraconnect	Acc - Acc Interconnect	Memory	TDP / Device
GH200 (JEDI)	4 x NVIDIA GH200 - 120 GB (1 x 72 core Grace CPU + 1 x H100 GPU)		NVLink-C2C 900 GB/s	NVLink4 900 GB/s	4 x IB NDR (4 x 200 Gbit/s)	4 x **120 GB** LPDDR5X (CPU), 4 x 96 GB HBM3 (GPU)	**680 W** (Superchip)
GH200 (JURECA)	**1** x NVIDIA GH200 - 480 GB (1 x 72 core Grace CPU + 1x H100 GPU)		NVLink-C2C 900 GB/s	-	-	**480 GB** LPDDR5X (CPU), 96 GB HBM3 (GPU)	**700 W** (Superchip)
H100 (JURECA)	4 x NVIDIA H100 GPU (**PCIe**)	2 x **72 core** Intel Xeon Platinum 8452Y	PCIe Gen5 128 GB/s	GPU 0-1 and GPU 2-3 are linked via bridge with 12 NVLink4 (25 GB/s) (Total: 600 GB/s)	-	512 GB DDR5-4800 (CPU), **80 GB** HBM2e (GPU)	**350 W**
H100 (WestAI)	4 x NVIDIA H100 GPU (**SXM5**)	2 x **32 core** Intel Xeon Platinum 8462Y	PCIe Gen5 128 GB/s	NVLink4 900 GB/s	2 x IB NDR (2 x 400 Gbit/s)	512 GB DDR5-4800 (CPU), **94 GB** HBM2e (GPU)	**700 W**

ACCELERATORS

[6]: https://apps.fz-juelich.de/jsc/hps/jureca
[7]: https://apps.fz-juelich.de/jsc/hps/jedi
[8]: https://westai.de/

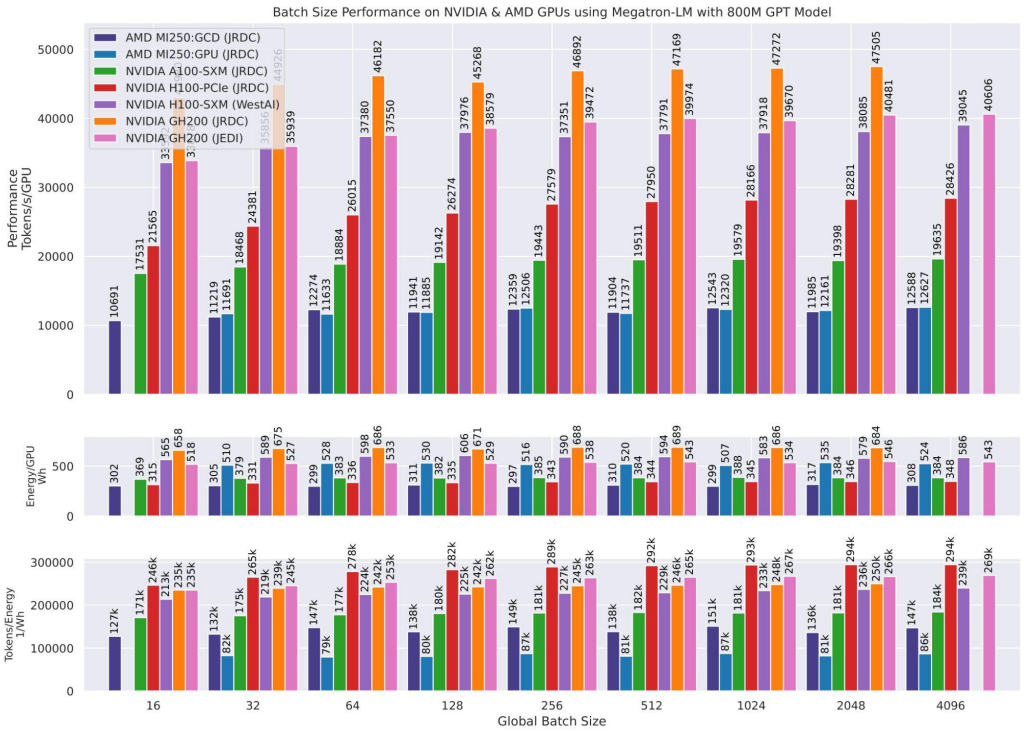
JURECA (JRDC), JEDI & WestAI Systems

Platform	Accelerator (Acc)	CPU	CPU - ACC Interconnect	Acc - Acc Intraconnect	Acc - Acc Interconnect	Memory	TDP / Device
A100 (JURECA)	4 x NVIDIA A100 GPU (SXM4)	2 x **64 core** AMD EPYC 7742	PCIe Gen4 64 GB/s	NVLink3 **600 GB/s**	2 x IB HDR (2 x 200 Gbit/s)	512 GB DDR4-3200 (CPU), 4 x 40 GB HBM2e(GPU)	400 W
MI200 (JURECA)	4 x AMD MI250 GPU (OAM)	2 x 48 core AMD EPYC 7413	PCIe Gen4 64 GB/s	Infinity Fabric 500 GB/s	2 x IB HDR (2 x 200 Gbit/s)	512 GB DDR4-3200 (CPU), 4 x **128 GB** HBM2e (GPU)	560 W
IPU-M2000 (JURECA)	4 x Graphcore GC200 IPU	2 x 48 core AMD EPYC 7413	**Ethernet 100 GbE**	IPU-Links Single IPU connects to 2 other IPUs with 2 links each and one with 4 links with 32GB/s bi-direction/link (Total: 256 GB/s)	-	512 GB DDR4-3200 (CPU)	**300 W**

RESULTS

NLP Benchmark on GPU

NLP Benchmark

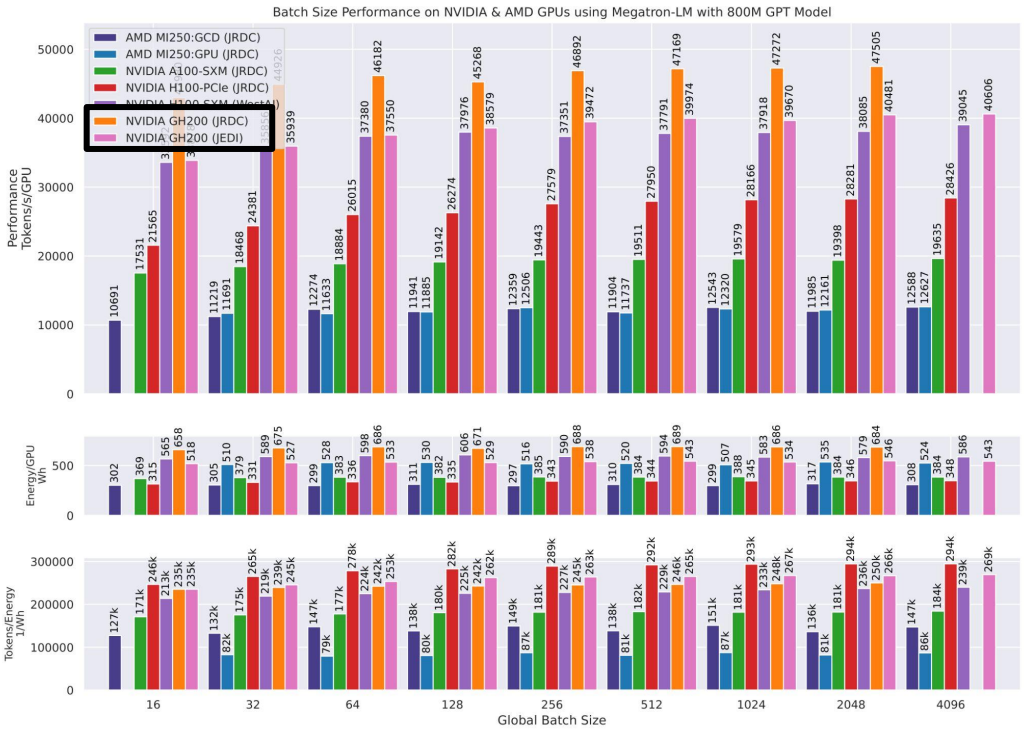


- Batch size performance in tokens/s/GPU
- **800M GPT** model trained on OSCAR using Megatron-LM
- Single node with Data Parallelism (DP) of 4 (except **GH200 JRDC**)

RESULTS

NLP Benchmark on GH200 Superchip

NLP Benchmark

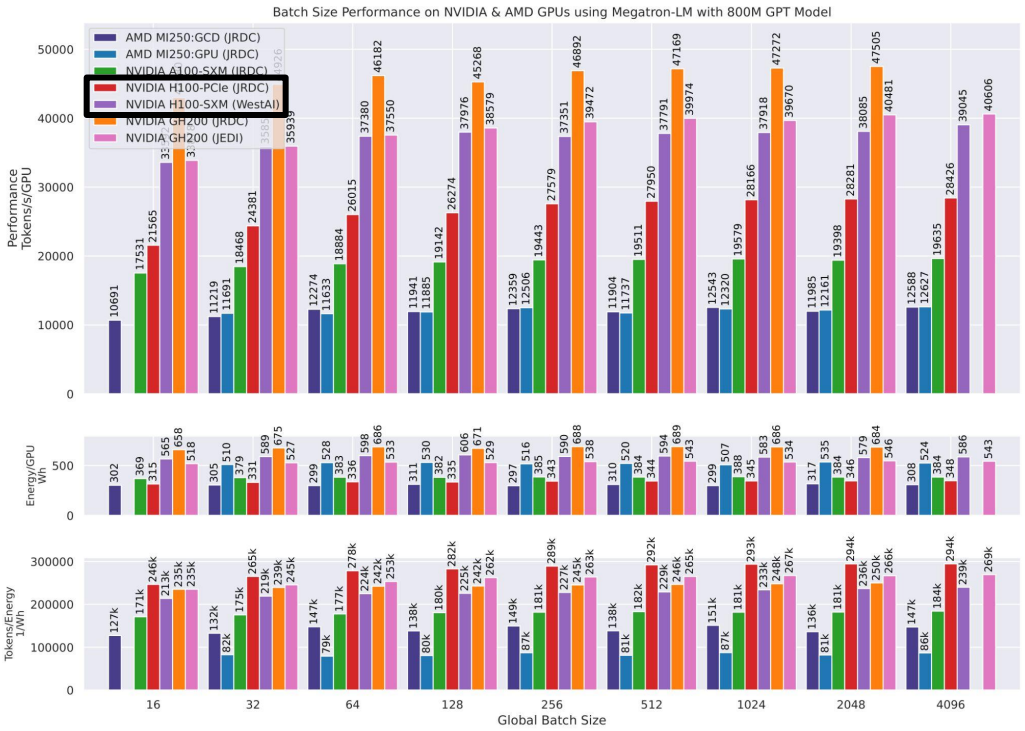


- Tokens/s/GPU:
  - JRDC > JEDI by 20% since no parallelisation
  - 2.45× than A100
- Energy/GPU:
  - JRDC > JEDI by 20% due to higher TDP (700W vs. 680W)
- Tokens/Energy:
  - JEDI slightly better

RESULTS

NLP Benchmark on H100 GPU

NLP Benchmark

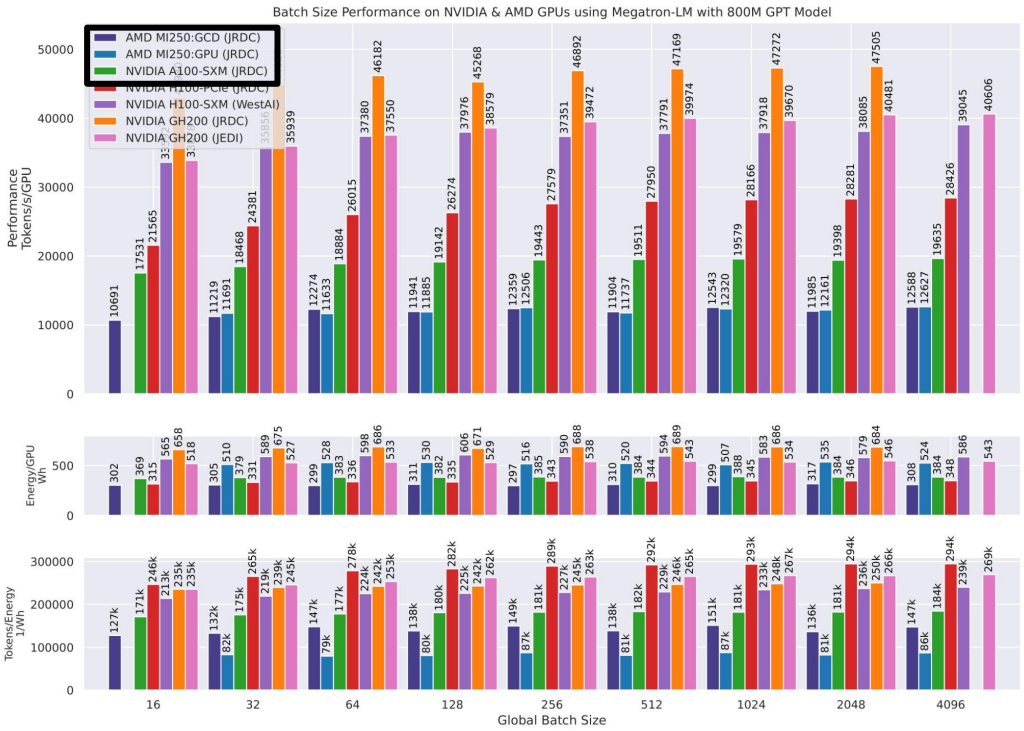


- Tokens/s/GPU:
  - **SXM** > **PCIe** by 1.3x with 1.5x NVLink bandwidth and SXM5 GPU form factor
- Energy/GPU:
  - **SXM** > **PCIe** by 1.6x
  - **SXM** TDP is 2x **PCIe** TDP
- Tokens/Energy
  - **H100 PCIe** outperforms all devices upto 25%

RESULTS

NLP Benchmark on A100 & MI250

NLP Benchmark



- Tokens/s/GPU:
  - **A100** > **MI250** by 1.6x
  - MI250: **GCD** slightly better than **GPU** due to lesser DP (4 v/s 8)
- Energy/GPU:
  - **MI250** > **A100** by 1.3x due to higher TDP (560 W v/s 400 W)
  - MI250: **GPU** > **GCD** by 1.7x due to higher DP
- Tokens/Energy
  - MI250: **GCD** > **GPU** by 1.7x
  - **A100** > **MI250 GPU** by 2x

RESULTS

NLP Benchmark

NLP Benchmark on Graphcore IPU M2000 POD4

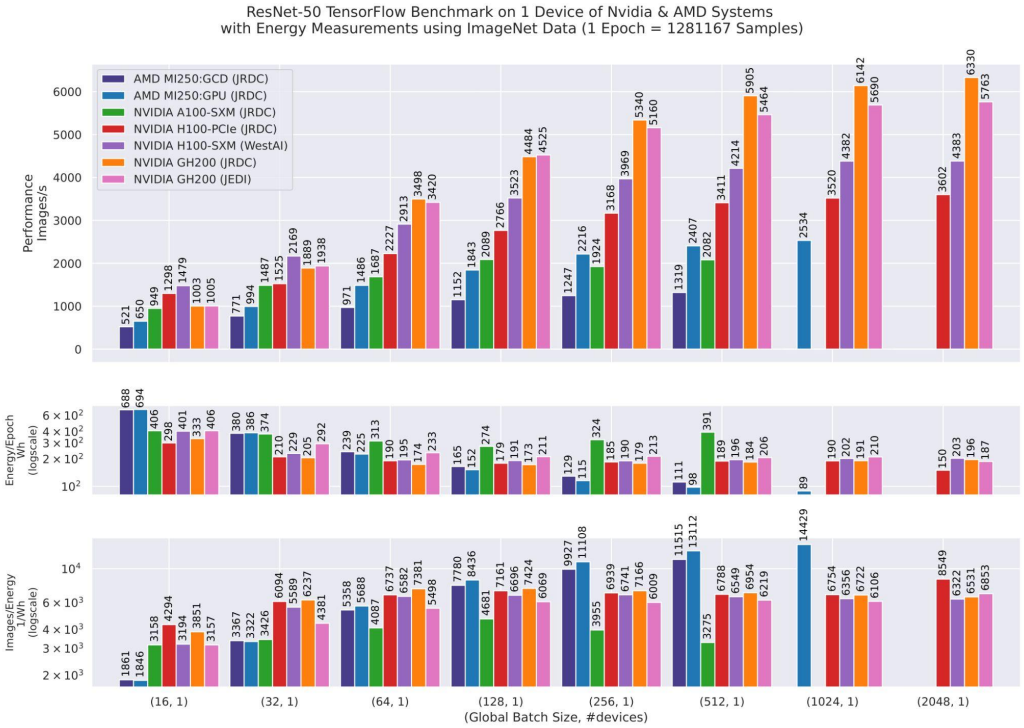
Batch Size	Tokens/Time 1/s	Energy/Epoch/IPU Wh	Tokens/Energy 1/Wh
64	64.99	15.68	4.08
128	97.21	18.20	7.03
256	129.96	18.37	13.93
512	155.72	18.56	27.60
1024	172.94	19.07	53.71
2048	183.37	20.05	102.13
4096	188.88	21.88	187.22
8192	191.86	25.47	321.34
16384	193.41	33.00	496.43

- Batch size performance in tokens/s (1 Epoch = 4*Batch size)
- **117M GPT** model trained on synthetic data using vendor benchmark
- Single node with Pipeline Parallelism (PP) of 4
- Performance saturates with large batch sizes due to limited SRAM (3.6 GB) and pipeline bubbles
- Max energy of 33 Wh/IPU

RESULTS

CV Benchmark on GPU

CV Benchmark

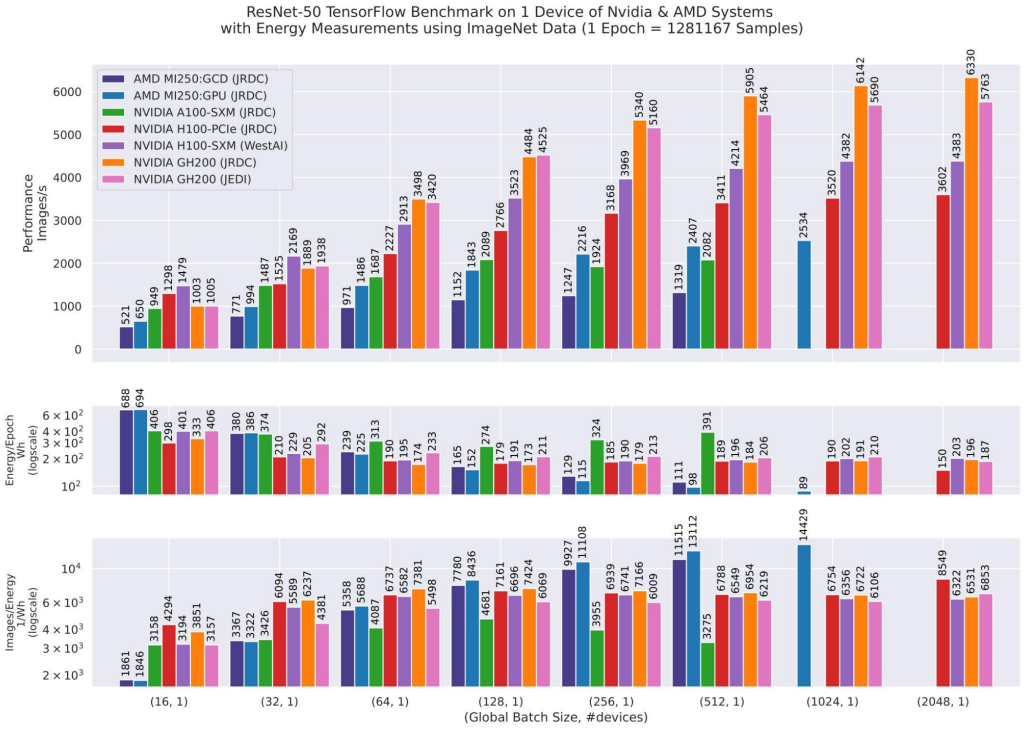


- Batch size performance in images/s on single device
- **ResNet50** model trained on ImageNet using TensorFlow benchmark

RESULTS

CV Benchmark on GPU

CV Benchmark



- Images/s:
  - GH200: **JRDC** > **JEDI**, more evident in larger batchsize
  - H100: **SXM** > **PCIe**, higher TDP SXM form factor
  - MI250: **GPU** > **GCD** due to parallelisation
  - **A100** > **MI250** for small batch size
- Images/Energy:
  - **H100-PCIe** and **GH200(JRDC)** for small batch size
  - **MI250** for large batch size

RESULTS

CV Benchmark

CV Benchmark on Graphcore IPU M2000 POD4

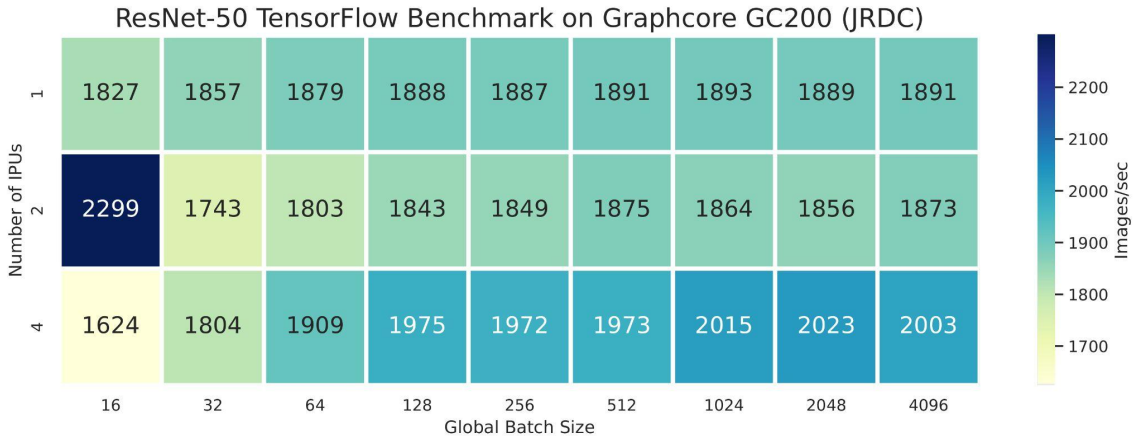
Batch Size	Images/Time 1/s	Energy/Epoch Wh	Images/Energy 1/Wh
16	1827.72	32.09	39925.87
32	1857.90	31.73	40382.19
64	1879.29	31.75	40346.18
128	1888.11	31.67	40452.50
256	1887.23	31.58	40563.65
512	1891.74	31.49	40689.85
1024	1893.07	31.50	40668.79
2048	1889.87	31.53	40636.28
4096	1891.58	31.51	40660.14

- Batch size performance on single IPU
- ResNet50 model trained on ImageNet data using vendor benchmark
- Performance does not scale well due to multiple DRAM access
- Model graph compilation ≅ 1 h, while execution≅ 10 - 15 mins

RESULTS

CV Benchmark

CV Benchmark on Graphcore IPU M2000 POD4

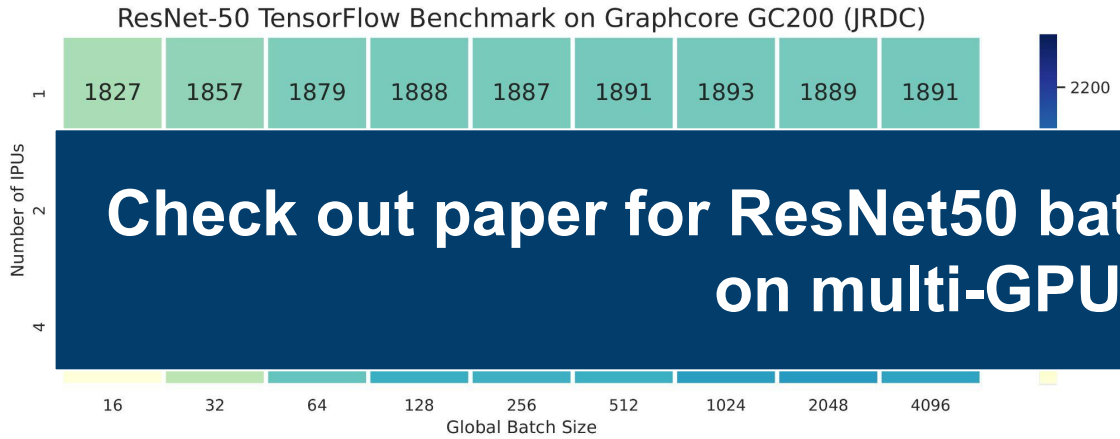


- Batch size performance on multiple IPU
- Performance increases with batch size
- Highest images/s for global batch size 16 and DP of 2

RESULTS

CV Benchmark

CV Benchmark on Graphcore IPU M2000 POD4



- Batch size performance on multiple IPU

Check out paper for ResNet50 batch size performance on multi-GPU!

- Highest images/s for global batch size 16 and DP of 2

CHALLENGES, INSIGHTS, NEXT STEPS

- Ensuring compatibility across accelerators is challenging
- Standardizing benchmarks across diverse accelerators is difficult
- Vendor support and optimizations vary
- Integrating containers with HPC and Slurm schedulers is complex
- System-specific optimizations (e.g., CPU-GPU affinity, MPI threads) are crucial
- Next
  - Expand support for new accelerators
  - Add more AI workloads, inference and communication benchmarks

CONCLUSION

Performance Trends

- Newer GPU generations show performance gains, with GH200 nodes in the lead
- H100 (SXM) surpasses H100 (PCIe) with NVLink higher bandwidth and SXM form factor

NLP Benchmark

- GH200 (JEDI) lags behind GH200 (JRDC) due to data parallelism overhead
- AMD MI250 (4 GCDs) slightly outperforms 4 GPUs, reflecting data parallelism limits
- GC200 IPU improves tokens/s with batch size but remains less efficient than GPUs due to pipeline bubbles
- H100 (PCIe) is most energy efficient, constrained by PCIe power limits

CV Benchmark

- AMD MI250 achieves higher throughput with 2 GCDs than a single GCD
- GC200 IPU reaches peak performance within SRAM limits
- AMD MI250 is energy efficient at larger batch sizes, while GH200 and H100 excel at smaller ones

ACKNOWLEDGEMENTS

- Funding
  - Funded by BMWK via OpenGPT-X (project no. 68GX21007D) and EuroHPC (Grant 955513)
  - Co-funded by BMBF through the MAELSTROM project (ref. 16HPC029)
  - Acknowledge use of JURECA-DC, JURECA-DC Evaluation Platform, WestAI infrastructure, and JUPITER platform JEDI

- CARAML Team


Stepan Nassyr


Dr. Carolin Penke


Dr. Andreas Herten

Thank you!
c.john@fz-juelich.de