COMPUTING
INSIGHT UK 2025
Computing Unites
4 – 5 DECEMBER 2025
Manchester Central, UK
www.ukri.org/CIUK
UKRI Science and Technology Facilities Council

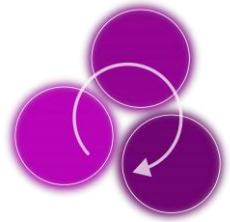
The Performance of GPU-enabled Molecular Dynamics (MD) Community Codes

Martyn Guest and Jose Munoz Criollo

Advanced Research Computing @ Cardiff (ARCCA) & Supercomputing Wales

Performance of GPU-enabled Molecular Dynamics Community Codes

- This presentation looks to demonstrate the performance of a number of widely used **GPU-enabled Molecular Dynamics (MD) community codes** important in understanding biological systems and drug interactions. This field was an early and significant adopter of GPU acceleration, with the featured codes here – **AMBER, GROMACS, LAMMPS** and **NAMD** – widely used on both local and regional clusters.
- Our objective is to **demonstrate the performance improvements of the selected codes on GPUs** compared, for example, to those seen on **CPU systems**, **understanding** the impact on performance related to the availability of various GPU (& CPU) features. **Attention focused here on systems hosting NVIDIA's V100, A100, L40S and H100 GPUs.** Additional insight provided from use of **Linaro's Performance Reports.**
- Note that this work was in part undertaken under the auspices of the UKRI funded **SHAREing project** ("*Skills Hub for Accelerated Research Environments Inspiring the Next Generation*") **https://shareing-dri.github.io/**



https://developer.nvidia.com/hpc-application-performance

Featured GPUs – 1. NVIDIA P100 accelerator nodes

The **13 x P100 accelerator nodes** are Dell PowerEdge R740 servers, with 2 x Intel Xeon Gold 6148 20-core 2 2.4GHz processors, 384GB memory (12 x 32GB ECC DDR4 2666MT/s dual rank RDIMMs), dual redundant power supplies, and an Infiniband host fabric interface card. The server is fitted with a GPU enablement kit and **2 x NVidia tesla P100 16GB PCIe GPU cards**.

Each node contains:

- **Intel Xeon Gold 6148 (Skylake) 2.40GHz processors**
- **20 cores/socket** (2.40GHz, 10.4GT/s, Turbo+, 150W) giving 40 cores per node
- 4.8GB RAM per core (192GB ECC DDR4 memory 2666MHz)
- 120GB SSD disk
- Single Infiniband EDR/PCIe Gen3-x8 interface embedded in the motherboard.
- **Two x NVidia tesla P100 16GB PCIe GPU cards**

Featured GPUs – 2. NVIDIA V100 accelerator nodes

The **14 x V100 accelerator nodes** are PowerEdge R740 servers, with 2 x Intel Xeon Gold 6248 2.5G, 20C/40T, 10.4GT/s, 27.5M Cache, Turbo, HT (150W) processors, 384GB memory (12 x 32GB DDR4-2933 2933MT/s dual rank RDIMMs), dual redundant power supplies, and an Infiniband host fabric interface card. The server is fitted with a GPU enablement kit and **2 x NVidia tesla V100 16GB PCIe GPU cards.**

- Each node contains:
  - **Intel Xeon Gold 6248 2.5GHz processors**
  - **20 cores/socket** (2.0GHz, 10.4GT/s, Turbo+, 150W) giving 40 cores per node
  - 4.8GB RAM per core (192GB ECC DDR4 memory 2933MHz)
  - 240GB SSD disk
  - Single Infiniband EDR/PCIe Gen3-x8 interface embedded in the motherboard.
  - **Two x NVidia tesla V100 16GB PCIe GPU cards**

The AccelerateAI System at Swansea 3. NVIDIA A100 GPUs

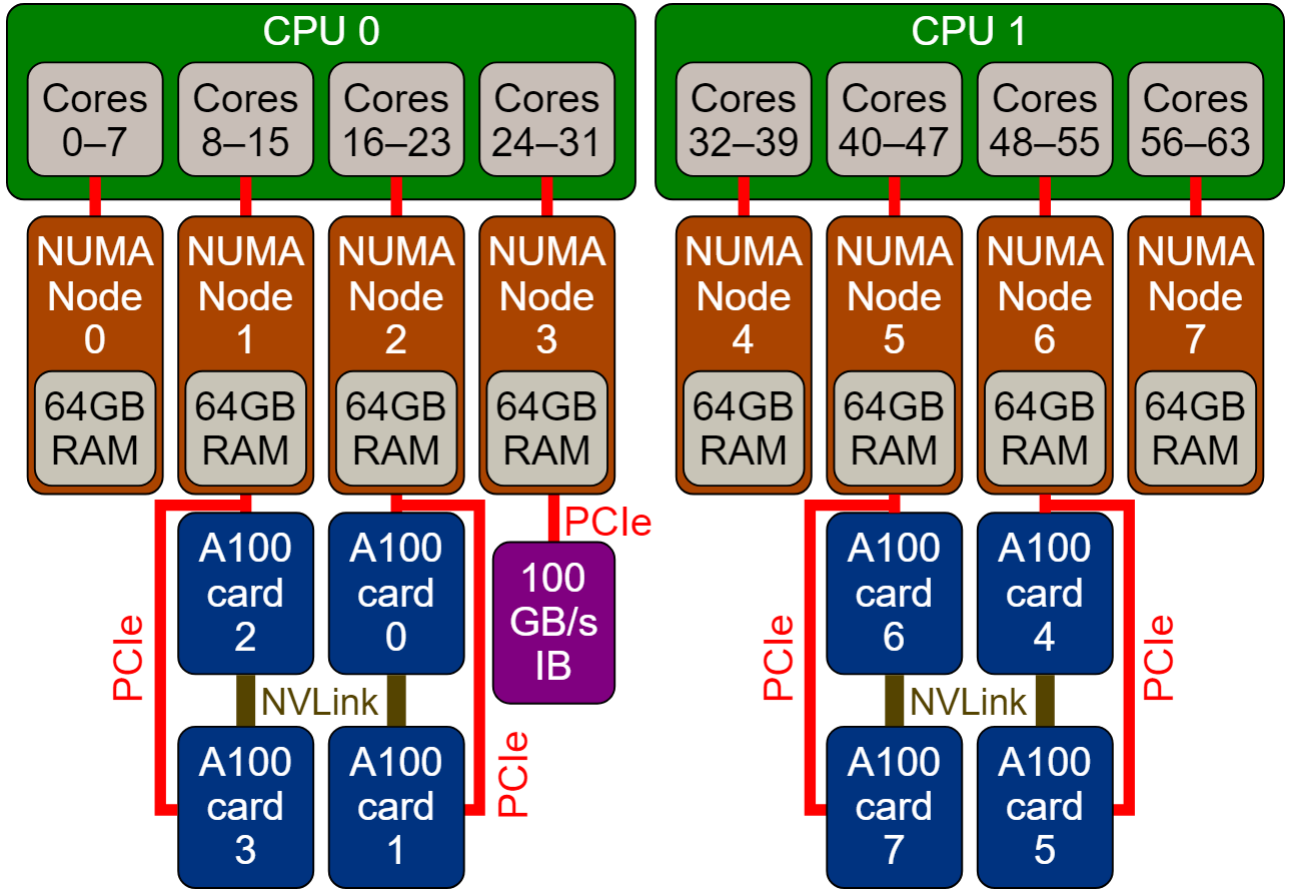
- **AccelerateAI** is a compute cluster designed to enable high-speed AI and ML computations, by making use of **NVIDIA A100 GPUs.** It is available to all researchers at Swansea, Aberystwyth, Bangor, and Cardiff Universities, free at the point of use.
- **The cluster** comprises **six BullSequana X410-A5 nodes**, connected to each other (and to the Supercomputing Wales SUNBIRD hardware) via NVIDIA Networking Infiniband EDR.
- Each node comprises:
  - ❖ 2 × AMD EPYC Rome 7452 32-core CPUs
  - ❖ **8 × NVIDIA A100 PCIe 40GB GPUs**
    - ❑ 4 x NVIDIA NVLink bridges between pairs of GPUs
  - ❖ **512GB RAM**
  - ❖ 1 × NVIDIA Networking **Infiniband EDR 100GB/s link**

Understanding the AccelerateAI architecture

Node structure

Each CPU has four “chiplets” inside, each with eight cores and a dedicated NUMA node with 64GB of RAM.

Each GPU, and the IB adapter, are also associated with one of the eight NUMA nodes. Pairs of GPUs are located on the same NUMA node, and these pairs are also linked together with NVLink for higher-speed transfers.



This high-level structure is illustrated in the above diagram

Falcon GPU Infrastructure - 4. System with 8 x NVIDIA L40S 48GB GPUs

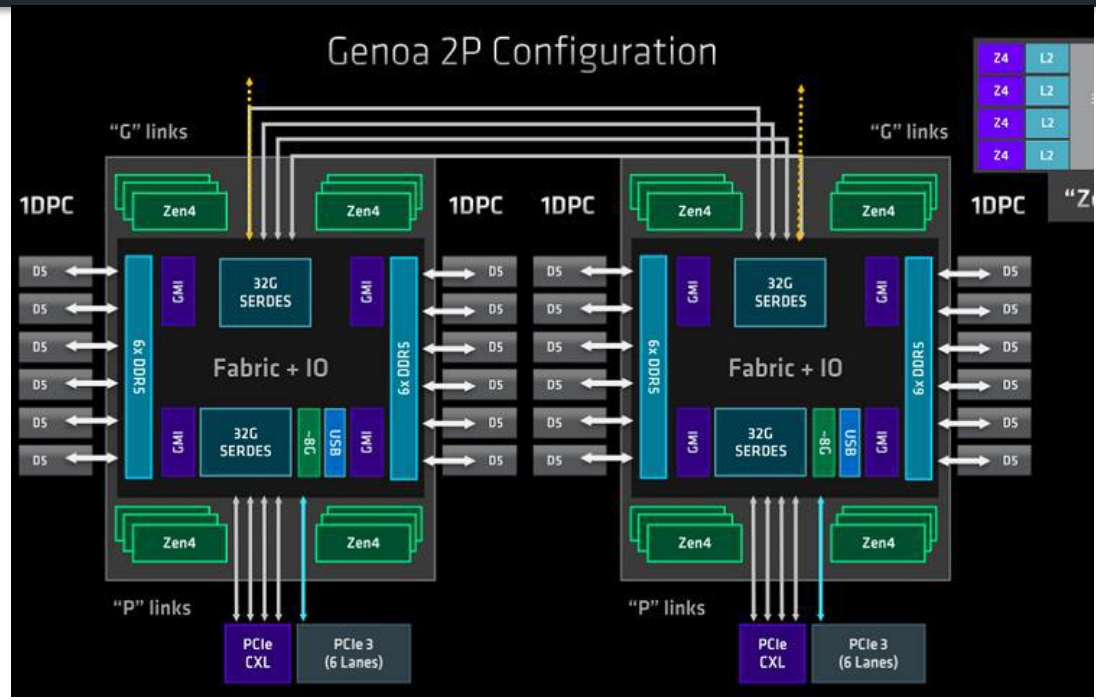
- A GPU development pool featuring **8 x NVIDIA L40S 48GB GPUs.**
- One 8-way system (**Supermicro SYS-521GE-TNRT server**), with NVIDIA L40S 48GB GPUs to create a development environment capable of supporting a range of AI projects.
- Memory Capacity: The **48GB memory / GPU** strikes a balance between **cost and capability**. Sufficient for many development scenarios, with flexibility in resource allocation – a single GPU for smaller projects or scale up to use multiple GPUs for more demanding tasks.
- **Power Efficiency:** The L40S has a significantly lower Thermal Design Power (TDP) compared to the H100 GPU. The L40S has a TDP of 300W, while the H100 PCIe has a TDP of 350W, while the SXM5 version reaches up to 700W.
- **2 x SPR 6430 2P 32C 2.1G 270W**; 1 TB memory 13 x PCIe Gen 5.0 X16 FHFL Slots
- 8 x 2.5" Hot-swap SATA drive bays, 8x2.5" Hot-swap NVMe drive bays
- **BUT .. Lack of fp64 Tensor Cores !!**

Falcon GPU Infrastructure - 5. System with 4 x H100 80GB NVIDIA GPUs

- Supermicro HGX SYS-221GE-TNHT-LCC server hosting a **system** with **4 x H100 80GB NVIDIA GPUs (NVIDIA SXM HGX H100 4-GPU 80GB)**
- 80GB memory capacity / GPU enables work on AI models requiring extensive memory resources e.g., LLM or complex computer vision applications.
- **NVLink Integration**: The inclusion of NVLink technology in this system significantly enhances GPU-to-GPU communication. Highest GPU communication using NVIDIA® NVLINK High density 2U system with NVIDIA®
- **2 x Intel EMR 6530 32C 2.1G 160MB 270W**, 1 TB memory DDR4
- 4 PCIe Gen 5.0 X16 LP
- Direct-To-Chip Liquid Cooling solution 2x 5250W(1+1) Redundant Power Supplies
- Additional networking : NVIDIA ConnectX-6 VPI , 2 x 25G Broadcom.

Systematic Approach

- For each application and associated data sets, contrast the **measured performance** across the variety of available **GPUs – V100, A100, L40S and H100** –with the scaling performance observed on **CPU system**(s).
- Key metric used through is not elapsed time but reported **ns/day** figures - the number of nanoseconds (ns) of simulation) that can be achieved in a day of computation.
- Main focus is on **single GPU performance** comparisons (though not exclusively)



- **CPU Performance** is based on **Falcon's Core MPI Partition** that features AMD's 4th-generation EPYC "**Genoa**" processors.
  - ❑ **30 x AMD EPYC Genoa 9654 Zen 4 processors.**
  - ❑ **96 cores, 2 sockets (2.40GHz, 48.0GT/s, Turbo+, 360W) giving 192 cores per node.**
  - ❑ 4.0GB RAM / core (DDR5 ECC RAM 4800MT/s on 24 x 32 GB modules), 384 MB L3 cache, giving 768 GB RAM per node. NVIDIA ConnectX-6 VPI Adapter Card HDR/200GbE

Software Deployments and EasyBuild Modules

AMBER 24

V100, A100 : Amber/24.3-foss-2023a-AmberTools-24.10-**CUDA-12.1.1**

L40S, H100 : Amber/24.3-foss-2023a-AmberTools-24.10-**CUDA-12.1.1**

GROMACS [2025.3]

V100, A100 : GROMACS/2025.3-foss-2023b-**CUDA-12.4.0**

L40S, H100 : GROMACS/2025.3-foss-2023b-**CUDA-12.4.0**

NAMD [3.0.2]

V100, A100 : NAMD/3.0.2-foss-2025a-**CUDA-12.8.0**

L40S, H100 : NAMD/3.0.2-foss-2025a-**CUDA-12.8.0**

BUT: AMBER 22

V100, A100 : 22.4-foss-2022a-AmberTools-22.5-**CUDA-11.7.0**

Limitations that Amber/22 doesn't officially support newer CUDA versions

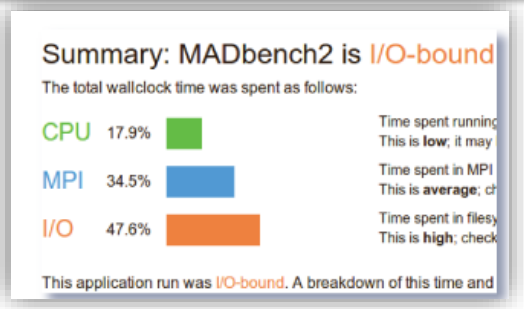
LAMMPS [29 Aug 2024 - Update 2]

V100, A100 : LAMMPS/29Aug2024_update2-foss-2023b-kokkos-**CUDA-12.4.0**

L40S, H100 : LAMMPS/29Aug2024_update2-foss-2023b-gpu-mixed-**CUDA-12.4.0**

Analysis Software - Allinea|ARM|Linaro Performance Reports

Provides a mechanism to characterize and understand the performance of HPC application runs through a single-page HTML report.



- Based on Linaro MAP's adaptive sampling technology that keeps data volumes collected and **application overhead low**.
- Modest application slowdown (ca. 5%) even with 1000's of MPI processes.
- **Runs on existing codes: a single command added to execution scripts.**
- If submitted through a batch queuing system, then the submission script is modified to load the Linaro module and add the 'perf-report' replacing the associated mpirun command.

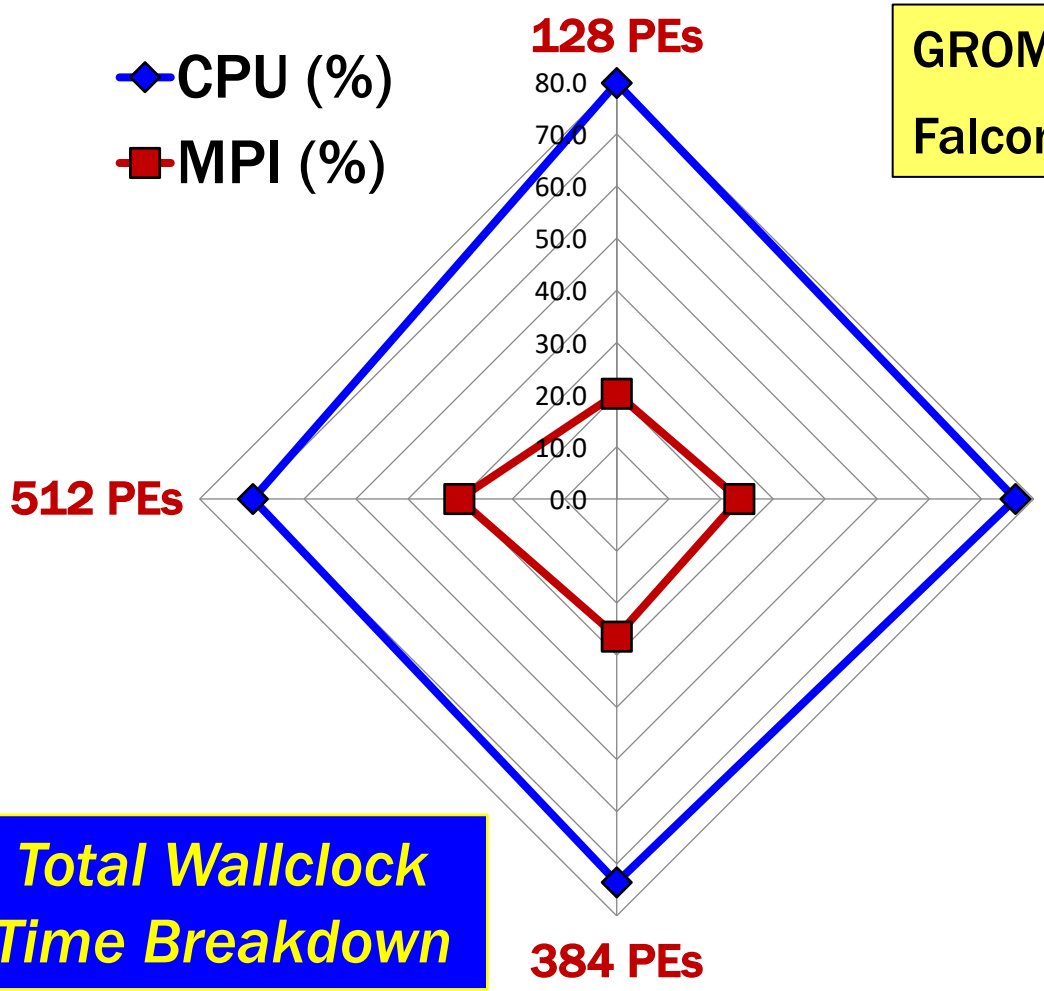
```
perf-report --output=filename.perf-report $code
```

- ***A Report Summary:*** This characterizes how the application's wallclock time was spent, broken down into **CPU, MPI and I/O**

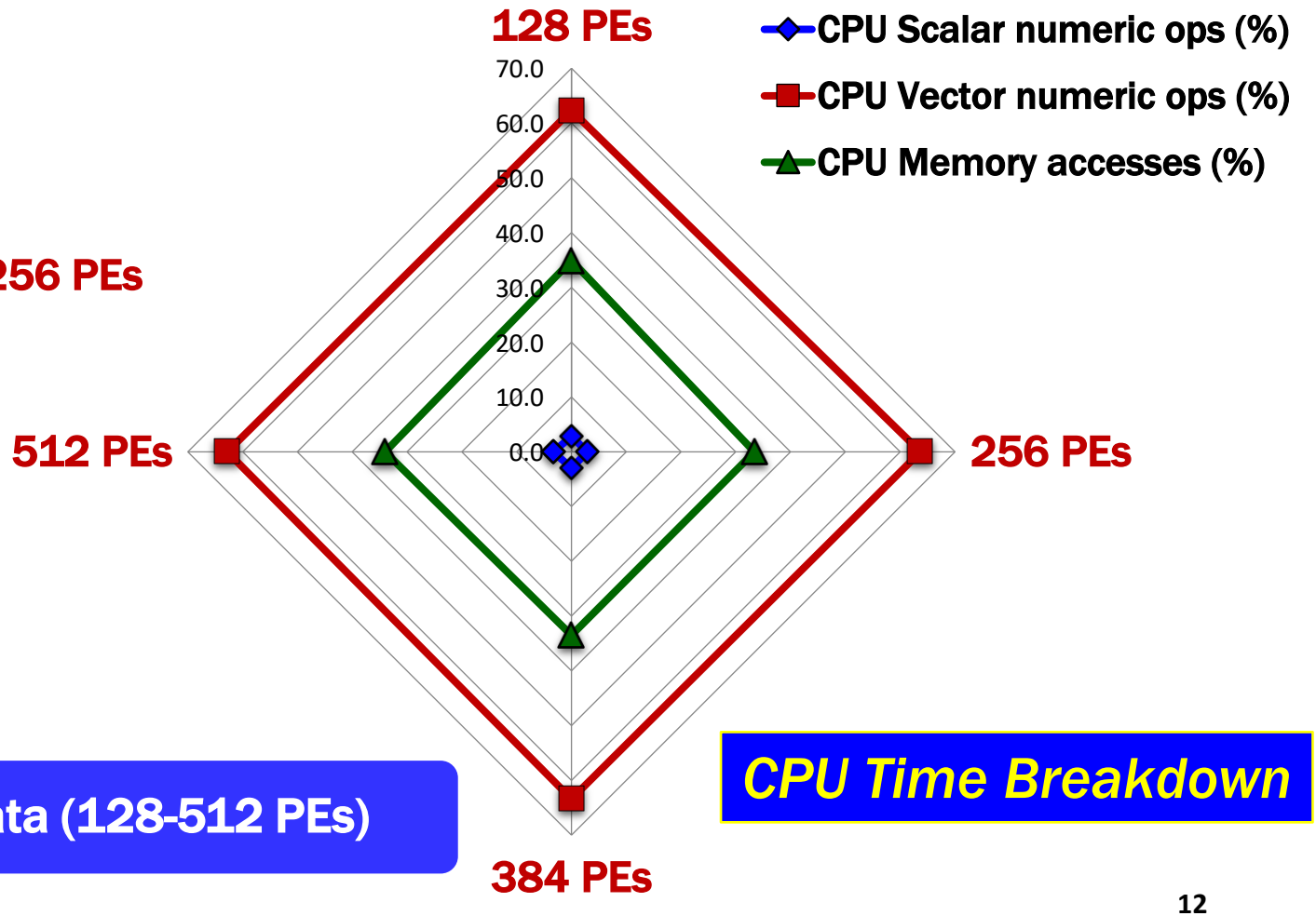
GROMACS 2024 – HECBioSim Performance Report

GROMACS/2024.4-foss-2023b; Linaro Forge 25.1.1

Falcon AMD Genoa 9654 / 2.4 GHZ



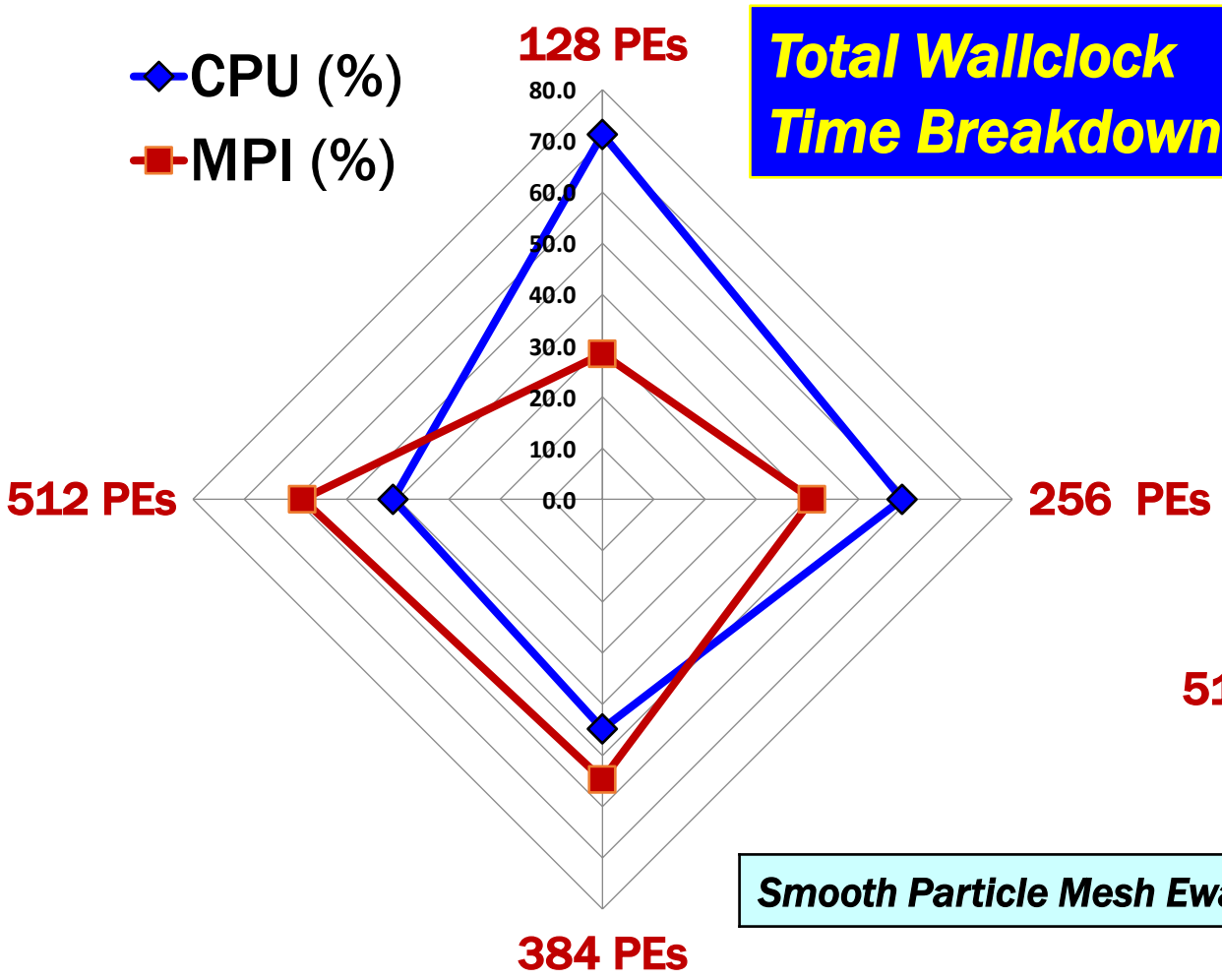
Total Wallclock Time Breakdown



CPU Time Breakdown

Performance Data (128-512 PEs)

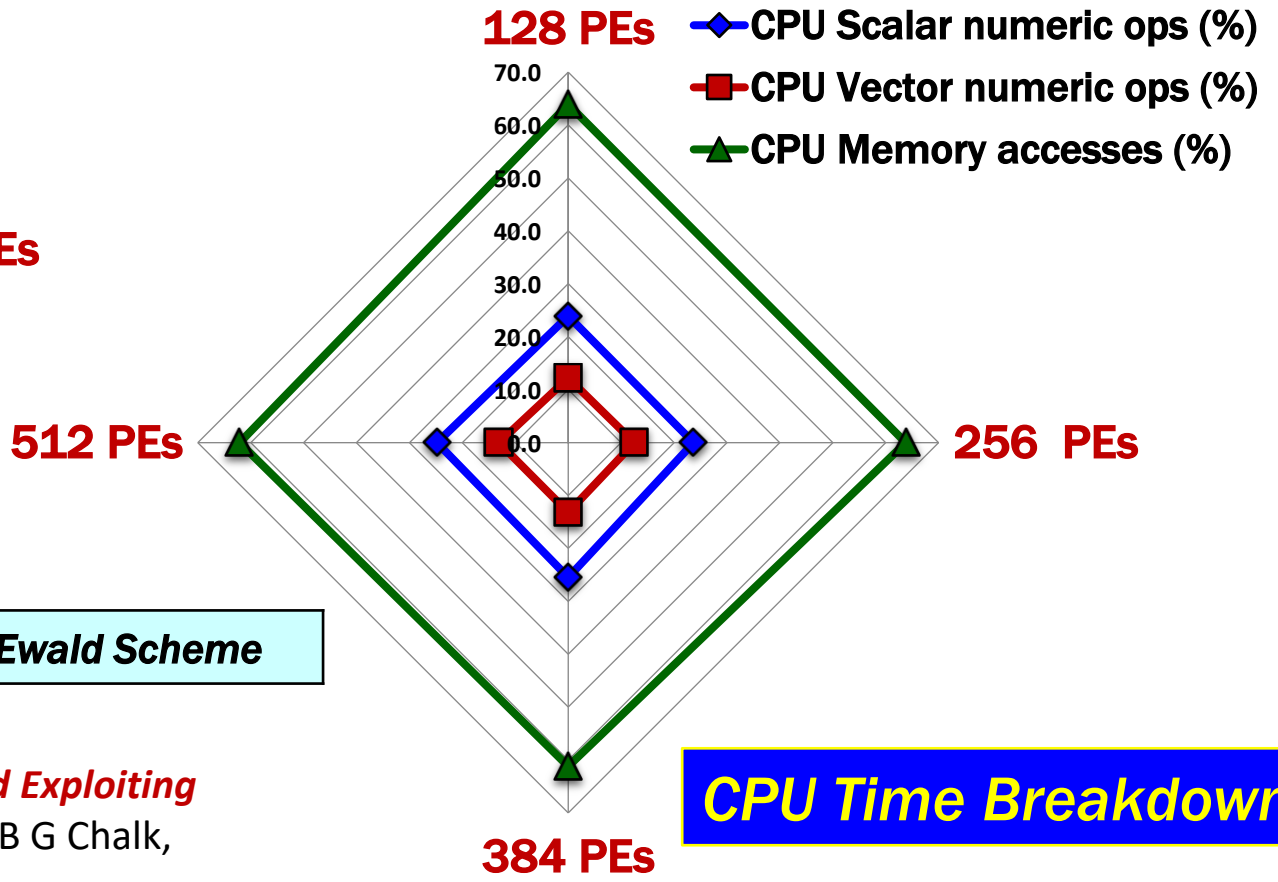
DLPOLY4 – Performance Report: NaCl Simulation



Total Wallclock Time Breakdown

DL_POLY_4/5.1.0; Linaro Forge 25.1.1

Falcon AMD Genoa 9654 / 2.4 GHZ



CPU Time Breakdown

Smooth Particle Mesh Ewald Scheme

"DL_POLY - A Performance Overview. Analysing, Understanding and Exploiting available HPC Technology", Martyn F Guest, Alin M Elena and Aidan B G Chalk, Molecular Simulation, (2019) 10.1080/08927022.2019.1603380

Monitoring, Analysing & Optimising usage of GPUs

1. **cluster-smi**: Useful for users to analyse workflows on multiple GPUs and administrative GPU monitoring tasks from the command line. https://github.com/PatWie/cluster-smi.
2. Custom script: Deployment of a custom Python script which **extracts the job SM (GPU core) & Memory ultilisation from the command line**. Currently, users use JupyterLabs via OnDemand to extract such metrics which are extremely important to know how they're utilising our GPUs.
3. **GPU-FPX**: Low-overhead tool to detect & analyse floating-point exceptions for NVIDIA GPUs. Can be incorporated into the build process/runtime monitoring for applications to improve numerical robustness and aid understanding where floating point exceptions occur within the system. https://github.com/LLNL/GPU-FPX enabled on JupyterLab ARCCA OnDemand.
4. Use of **nvtop** (https://github.com/Syllo/nvtop) added to the documentation.
5. Usage of **gpustat** (pip package, https://github.com/wookayin/gpustat) and **memory_viz** (PyTorch memory vis. tool https://pytorch.org/blog/understanding-gpu-memory-1/).

Analysis Software - Linaro Performance Reports - Accelerators

Command:	/shared/apps/easybuild/x86_64/intel/sapphirerapids/software/Amber/24.3-foss-2023a-AmberTools-24.10-CUDA-12.1.1/bin/pmemd.cuda.MPI -O -o /shared/home1/m.guest/Falcon.BenchMarks.Jun24/GPUS/AMBER/M45/amber243.M45.H100.ebfoss23.gpus1.n1.cpus1.ppn1.out.17594.1
Resources:	1 node (64 physical, 64 logical cores per node) 1 GPU per node available
Memory:	1008 GiB per node
Tasks:	1 process
Machine:	ccigh0001.falcon.supercomputingwales.ac.uk
Architecture:	x86_64
CPU Family:	emeraldrapids
Start time:	Thu Dec 25 02:55:15 2025
Total time:	99 seconds (about 2 minutes)
Full path:	/shared/apps/easybuild/x86_64/intel/sapphirerapids/software/Amber/24.3-foss-2023a-AmberTools-24.10-CUDA-12.1.1/bin

Summary: pmemd.cuda_SPFP.MPI is Compute-bound in this configuration

Accelerators

Breakdown of how CUDA accelerators were used:

GPU utilization	85.0%
Mean GPU memory usage	4.2%
Peak GPU memory usage	4.3%

GPU utilization

% of time during which one or more kernels were executing on the GPU, averaged across available GPUs.

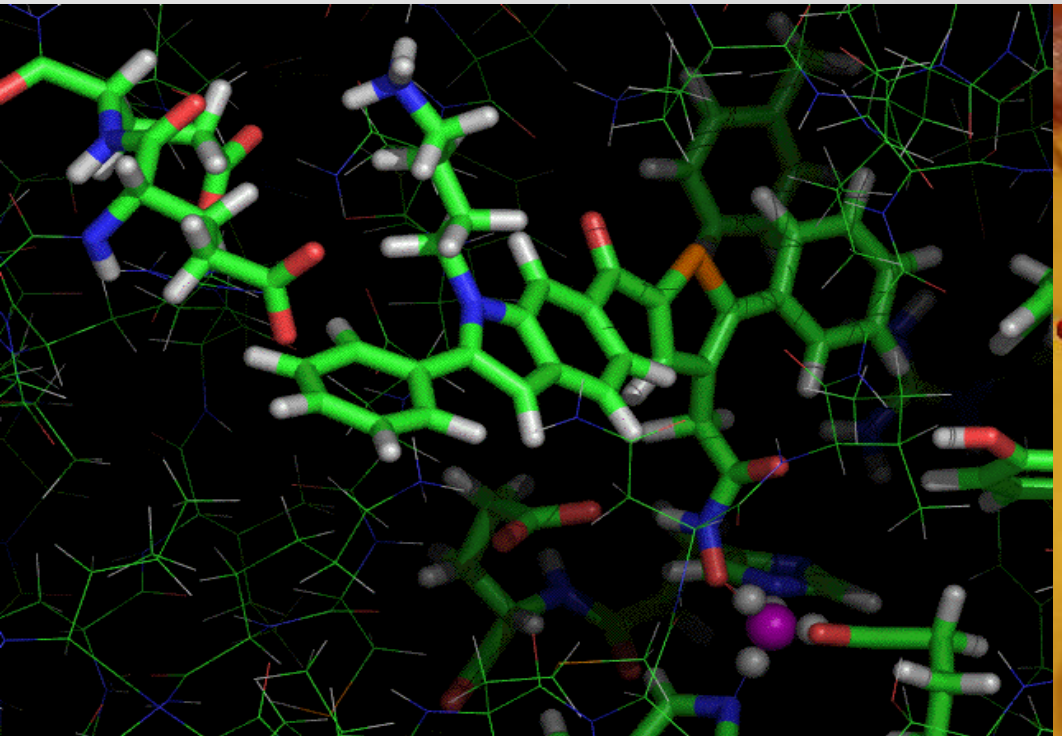
Mean GPU memory usage

Average amount of memory in use on GPU cards.

Peak GPU memory usage

Maximum amount of memory in use on GPU cards.

The Performance of Community GPU Codes in Life Sciences and Genomics



Molecular Simulation - AMBER

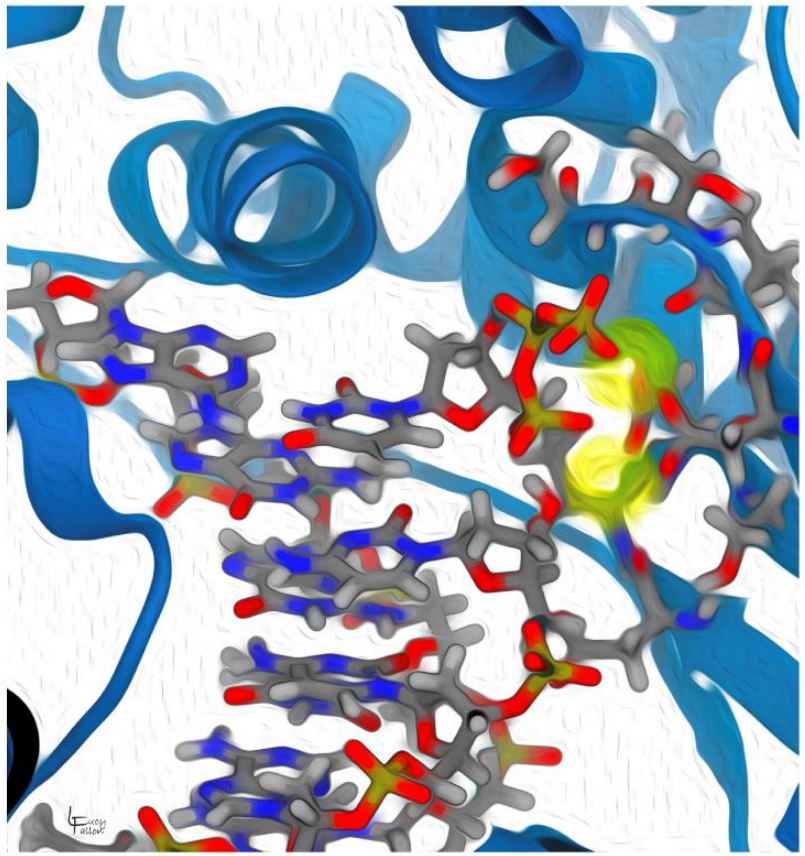
- **Amber** is a suite of biomolecular simulation programs. It began in the late 1970's and is maintained by an active development community.
- The term "Amber" refers to two things. First, it is a set of molecular mechanical force fields for the simulation of biomolecules; these force fields are in the public domain, and are used in a variety of simulation programs. Second, it is a package of molecular simulation programs which includes source code and demos.
- Amber is distributed in two parts: AmberTools25 and Amber24.
- Amber is developed in an **active collaboration** of David Case at Rutgers University, Tom Cheatham at the University of Utah, Ray Luo at UC Irvine, Ken Merz at Michigan State University, Maria Nagan at Stony Brook, and many others. Amber was originally developed under the leadership of **Peter Kollman**.

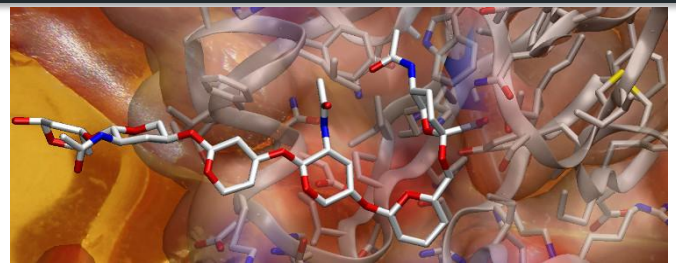
D.A. Case, H.M. Aktulga, K. Belfon, D.S. Cerutti, G.A. Cisneros, V.W.D. Cruzeiro, N. Forouzesh, T.J. Giese, A.W. Götz, H. Gohlke, S. Izadi, K. Kasavajhala, M.C. Kaymak, E. King, T. Kurtzman, T.-S. Lee, P. Li, J. Liu, T. Luchko, R. Luo, M. Manathunga, et al. ***AmberTools***. *J. Chem. Inf. Model.* **63,** 6183-6191 (2023).

R. Salomon-Ferrer, D.A. Case, R.C. Walker. ***An overview of the Amber biomolecular simulation package****. WIREs Comput. Mol. Sci. 3, 198-210 (2013).*

D.A. Case, T.E. Cheatham, III, T. Darden, H. Gohlke, R. Luo, K.M. Merz, Jr., A. Onufriev, C. Simmerling, B. Wang and R. Woods. ***The Amber biomolecular simulation programs****. J. Computat. Chem. 26, 1668-1688 (2005).*

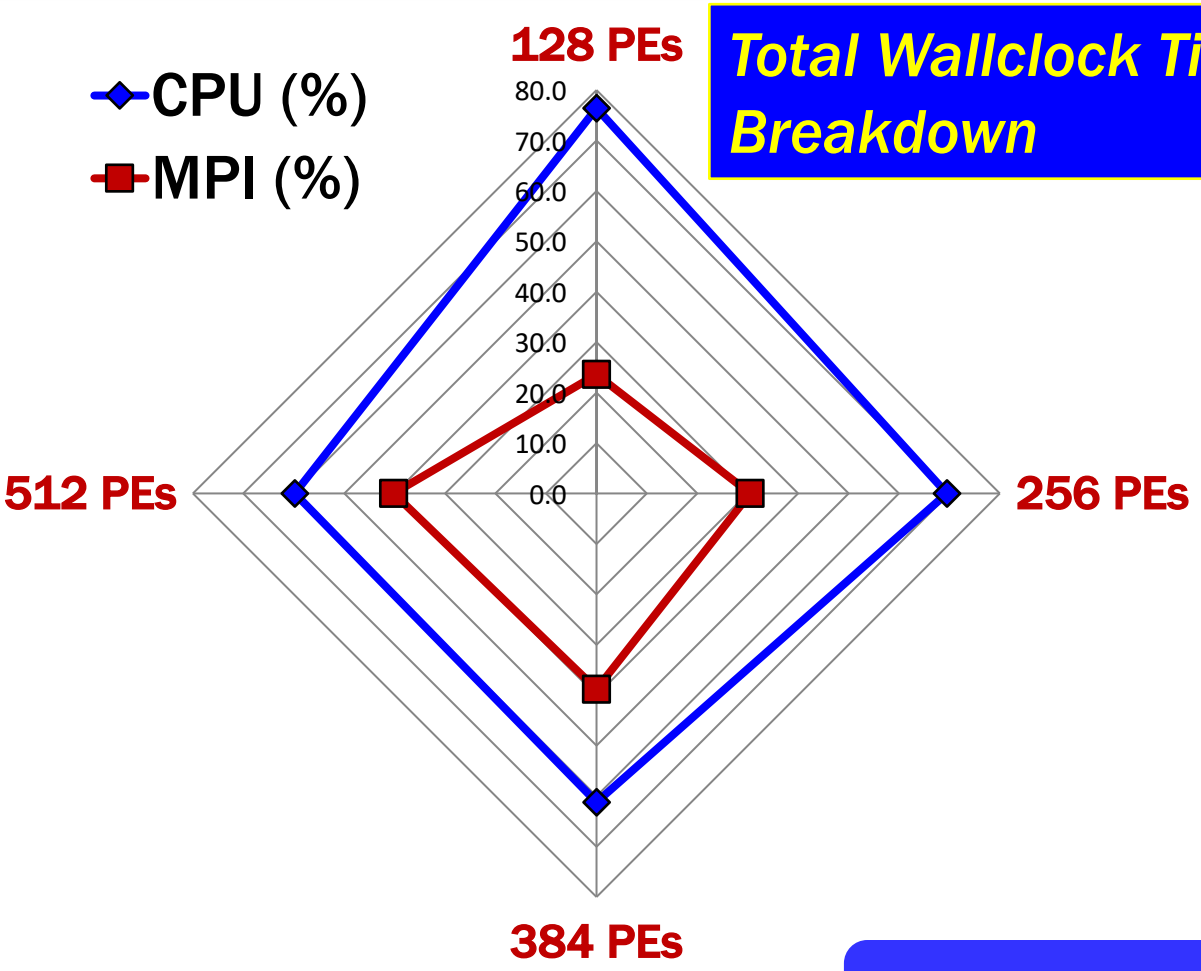
Early Experience with AMBER 18 and AMBER 22





- **AMBER18** (released in 2018) previously used in this study. In practice we find **also identical performance of AMBER18 and AMBER22 releases** when running the M01, M06, M27 and M45 performance test cases.
- **AMBER22** released (on April 27, 2022).
- The **Amber22** package builds on AmberTools22 by adding the pmemd program, which resembles the sander (MD) code in AmberTools, but provides better performance on multiple CPUs, and **dramatic speed improvements on GPUs**.
- Presentation limited to the **M27 and M45** test cases for M01 and M06 are now too small for meaningful analysis

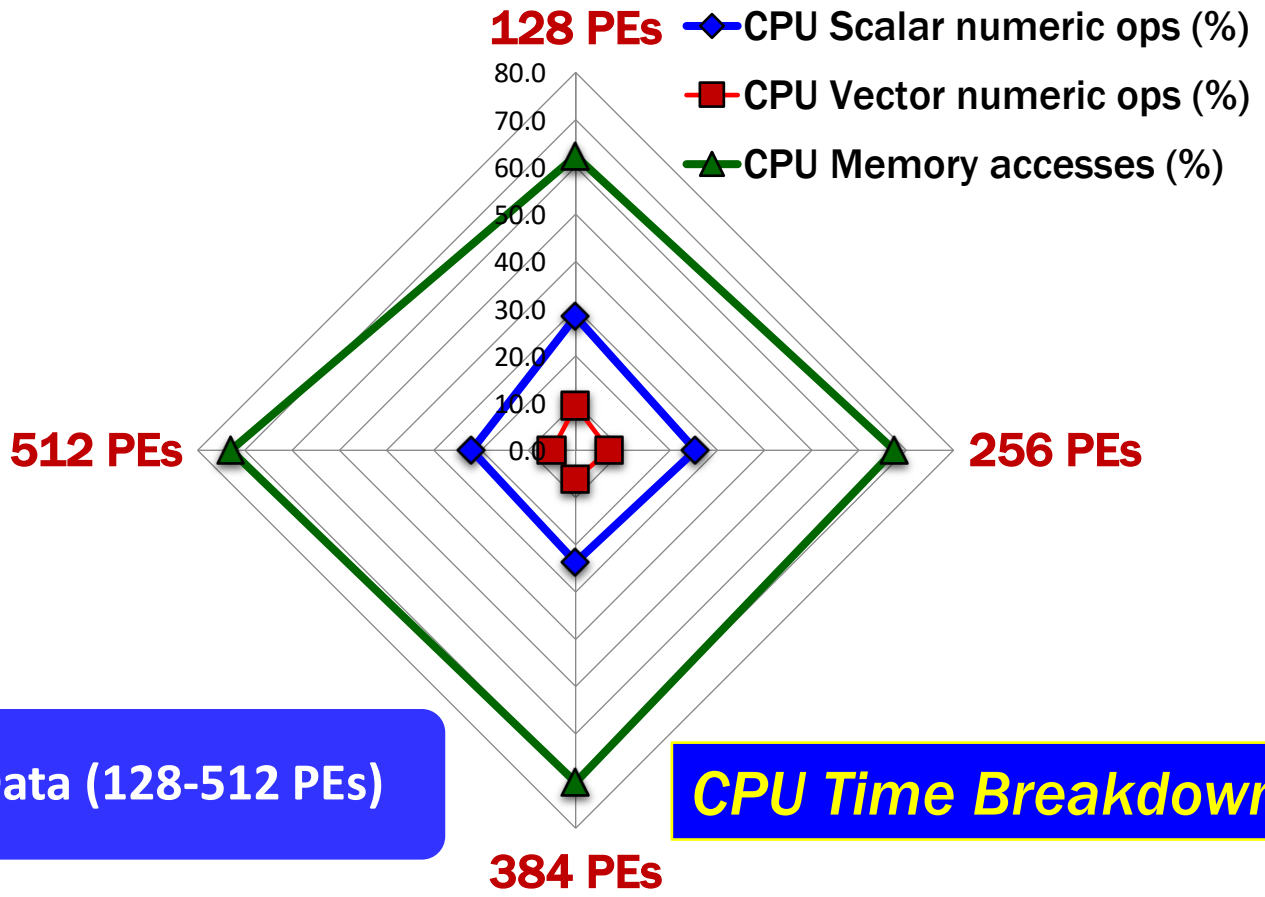
AMBER24 – M45 Benchmark Performance Report



Total Wallclock Time Breakdown

Amber/24.0-foss-2023b; Linaro Forge 25.1.1;

Falcon AMD Genoa 9654 / 2.4 GHZ

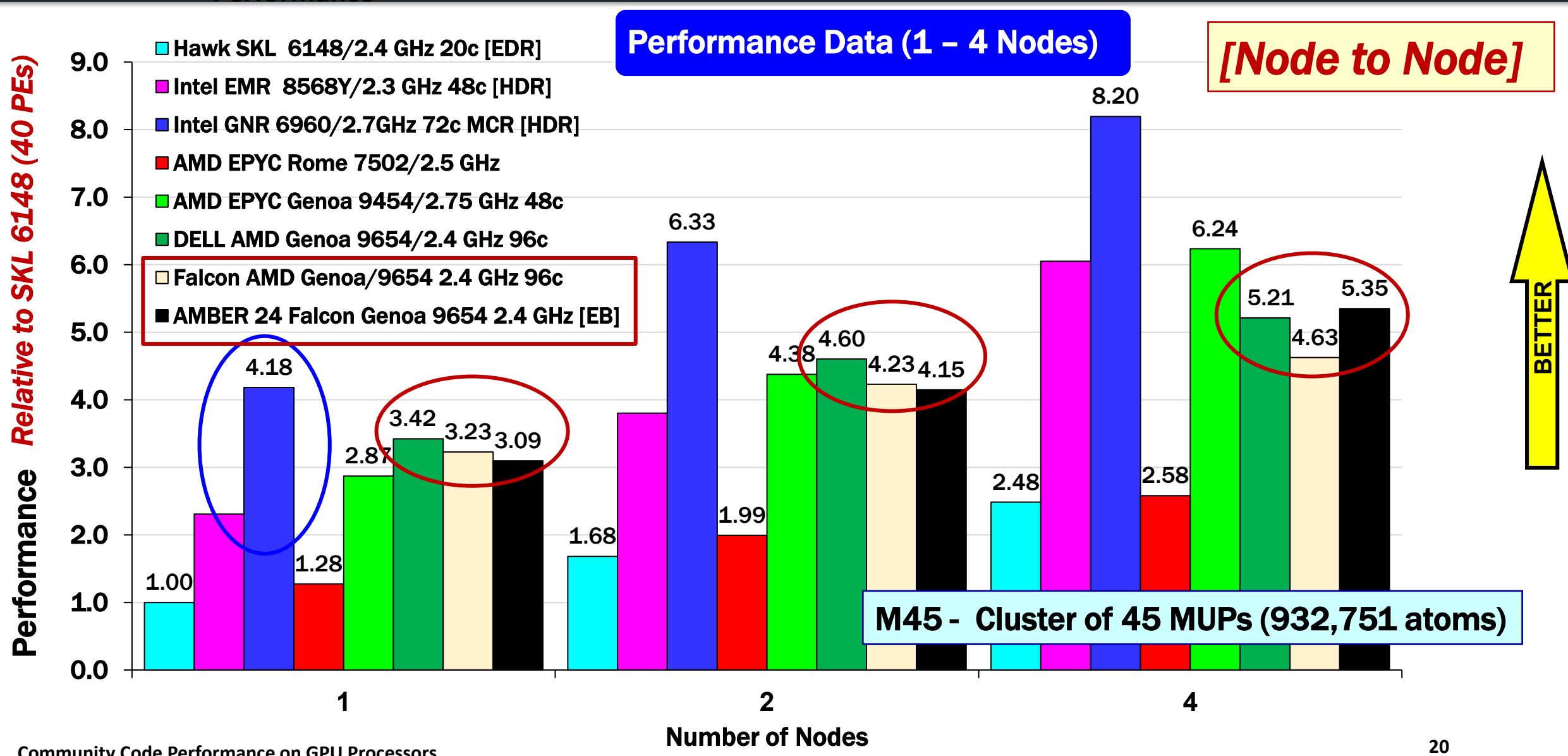


CPU Time Breakdown

Cluster of 45 Major Urinary Proteins (MUPs, 932,751 atoms)

Performance Data (128-512 PEs)

AMBER22 & AMBER24 – M45 Performance Analysis



AMBER 24 Background Architecture & Hardware Considerations

AMBER consists of a number of different S/W packages, with the **MD engine PMEMD** as the most compute-intensive and the target of extensive optimisations.

- Comprises single CPU (pmemd), multi-CPU (pmemd.MPI), single-GPU (pmemd.cuda), and multi-GPU (pmemd.cuda.MPI) versions. Traditionally, MD simulations would be executed on CPU, but AMBER MD simulations executing calculations on CUDA have made GPUs the most logical choice for speed and cost efficiency.
- Most AMBER simulations can fit on a single GPU running all within GPU memory; CPU performance, CPU memory, have little influence on simulation throughput performance, while storage speeds only impact the time it takes to load the model into GPU memory.
- AMBER also does not take significant advantage of multi-GPU acceleration on a single calculation. To fully maximize AMBER simulation throughput on multiple GPUs, run multiple independent AMBER simulations simultaneously on multiple GPUs i.e. test different molecular interactions simultaneously to answer questions faster.

AMBER 24 NVIDIA GPU Benchmarks

Breadth of classic MD problems, including PME and generalized born simulations. Simulations are ordered from largest to smallest to reflect relative performance at all model size levels. Results may vary based on storage speed, computing environment, and other factors.

STMV NPT 4fs: 1,067,095 atoms

Cellulose NVE 2fs = 408,609 atoms

Cellulose NPT 2fs = 408,609 atoms

FactorIX NVE 2fs = 90,906 atoms

FactorIX NPT 2fs = 90,906 atoms

DHFR (JAC Prod.) NVE 2fs = 23,558 atoms

DHFR (JAC Prod.) NPT 2fs = 23,558 atoms

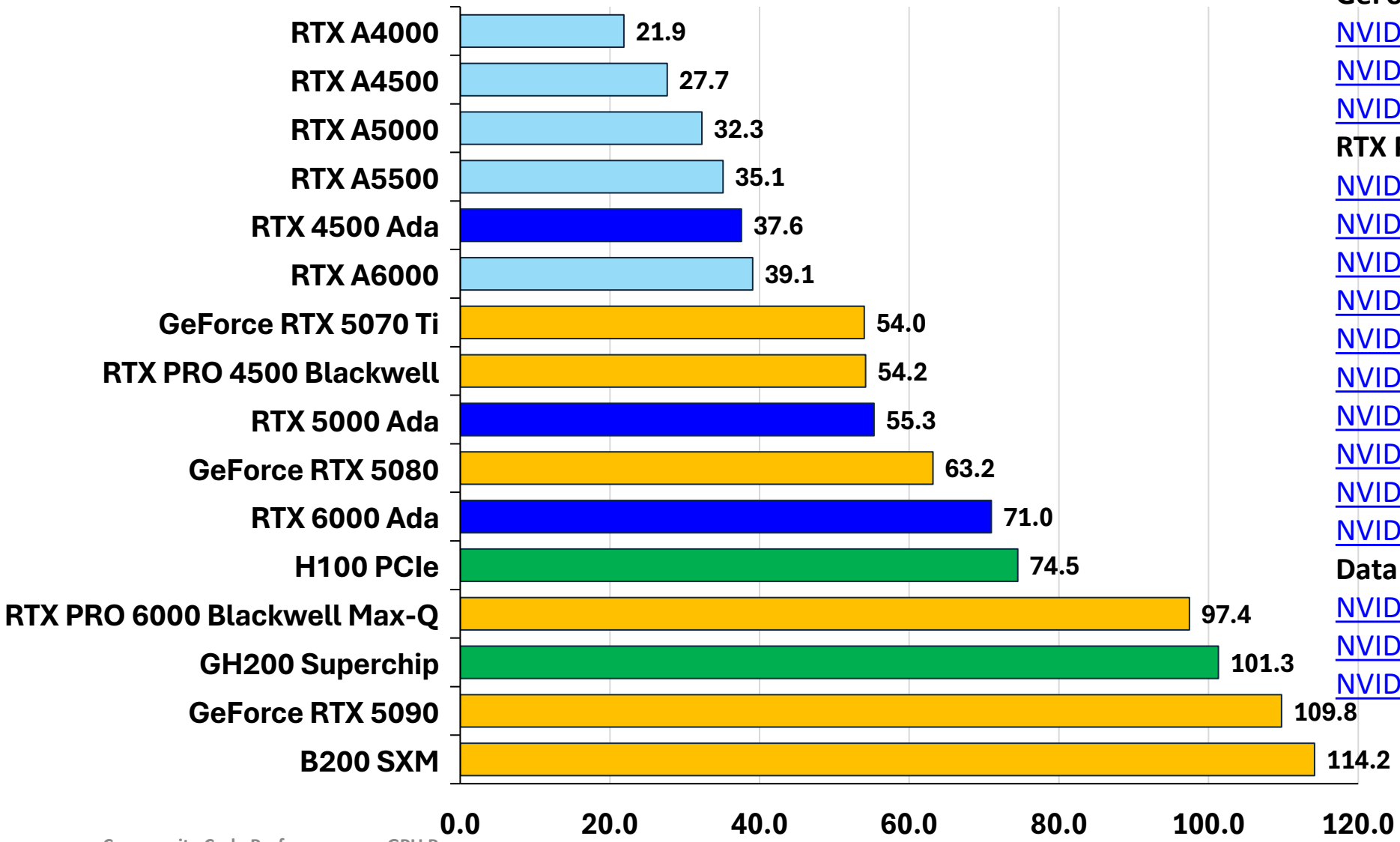
Nucleosome GB 2fs = 25,095 atoms

Myoglobin GB 2fs = 2,492 atoms

JAC_production_NVE_4fs	1672.00 ns/day ~24K atoms
JAC_production_NPT_4fs	1591.64 ns/day
FactorIX_production_NVE_2fs	451.00 ns/day ~91K atoms
FactorIX_production_NPT_2fs	418.44 ns/day
Cellulose_production_NVE_2fs	159.08 ns/day ~409K atoms
Cellulose_production_NPT_2fs	146 ns/day
STMV_production_NVE_4fs	131.41 ns/day ~1.1 M atoms
STMV_production_NPT_4fs	117.20 ns/day

September 2025 Blackwell (B200) results on a B200 cluster at the University of Florida (https://ambermd.org/GPUPerformance.php)

STMV Production NPT 4fs - 1,067,095 Atoms



GeForce

NVIDIA GeForce RTX 5090 (32GB)
NVIDIA GeForce RTX 5080 (16GB)
NVIDIA GeForce RTX 5070 Ti (16GB)

RTX Professional

NVIDIA RTX PRO 6000 Blackwell Max-Q (96G
NVIDIA RTX PRO 4500 Blackwell (32GB)
NVIDIA RTX 6000 Ada (48GB)
NVIDIA RTX 5000 Ada (32GB)
NVIDIA RTX 4500 Ada (24GB)
NVIDIA RTX A6000 (48GB)
NVIDIA RTX A5500 (24GB)
NVIDIA RTX A5000 (24GB)
NVIDIA RTX A4500 (20GB)
NVIDIA RTX A4000 (16GB)

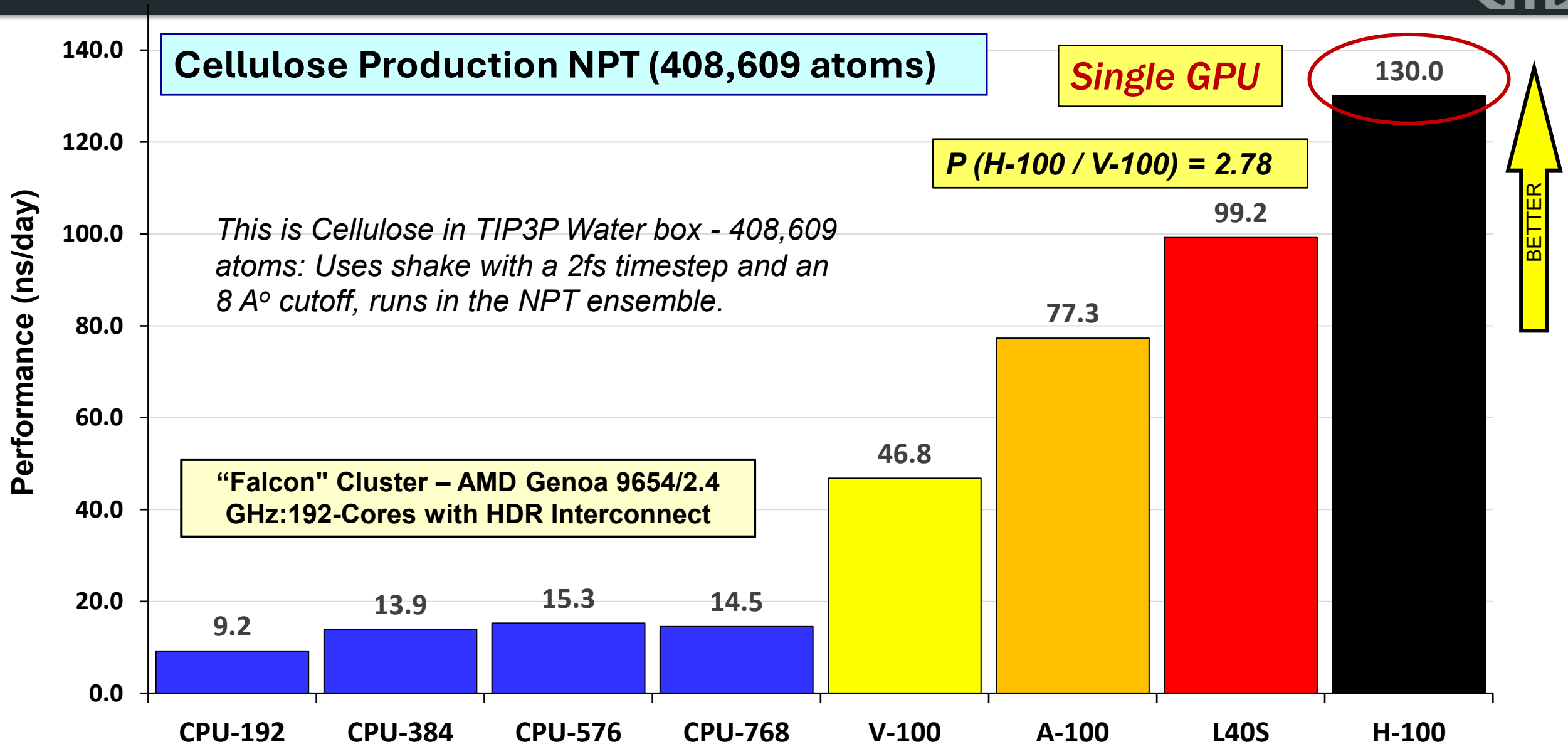
Data Centre

NVIDIA B200 SXM (192GB)
NVIDIA Grace Hopper GH200 (96GB)
NVIDIA H100 (PCIe) (80GB)

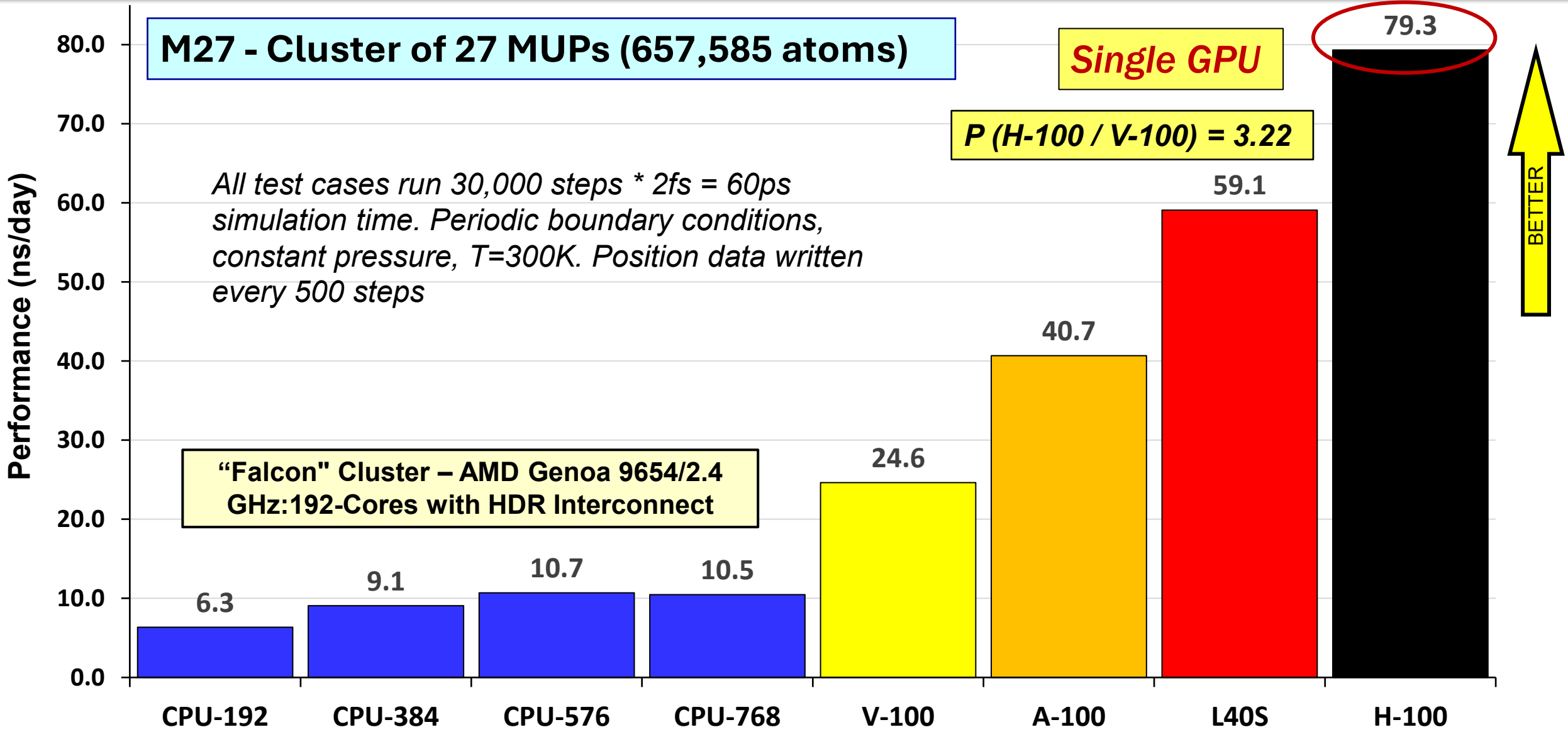
AMBER 24 Benchmark Cases - PMEMD & GPU accelerated PMEMD

1. **Cellulose Production NPT Benchmark**
   - Cellulose in TIP3P Water box - **408,609 atoms**: Uses shake with a 2fs timestep and an 8 Aº cutoff, runs in the NPT ensemble
2. **M27 Benchmark**
   - Cluster of 27 x Major Urinary Protein (MUP) + IBM ligand (21,736 atoms). Total number of atoms = **657,585 atoms**. 30,000 steps * 2fs = 60ps simulation time. Periodic boundary conditions, constant pressure, T=300K. Position data written every 500 steps.
3. **M45 Benchmark**
   - Cluster of 45 x Major Urinary Protein (MUP) + IBM ligand (21,736 atoms). Total number of atoms = **932,751 atoms**. Periodic boundary conditions, constant pressure, T=300K. Position data written every 500 steps
4. **STMV NPT 4fs Benchmark**
   - **1,067,095 atoms**. Uses shake with a 4fs timestep and an 8 Aº cutoff, runs in the NPT ensemble

AMBER 24 – GPU Performance: Cellulose Production Test Case



AMBER 24 – GPU Performance: M27 Test Case



AMBER 24 Performance Reports – GPU Utilisation Analysis I

Linaro Accelerator Analysis - Breakdown of how CUDA accelerators were used:

GPU utilization - % of time during which one or more kernels were executing on the GPU, averaged across available GPUs.

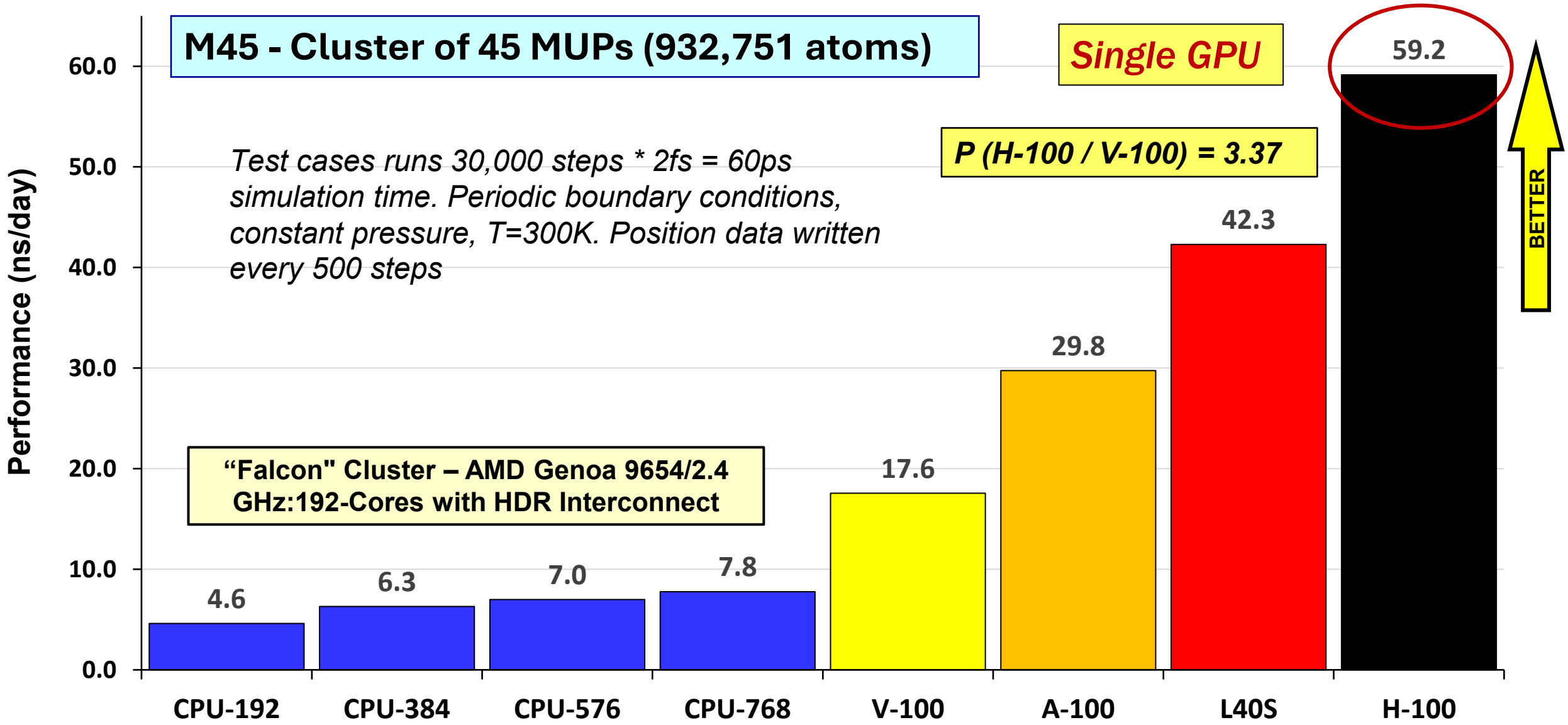
Cellulose Production NPT (408,609 atoms)

GPU Count	H100 Utilisation	L40S Utilisation
1	76.7%	78.7%
2	37.6%	35.6%
3	24.5%	25.1%
4	18.3%	17.9%

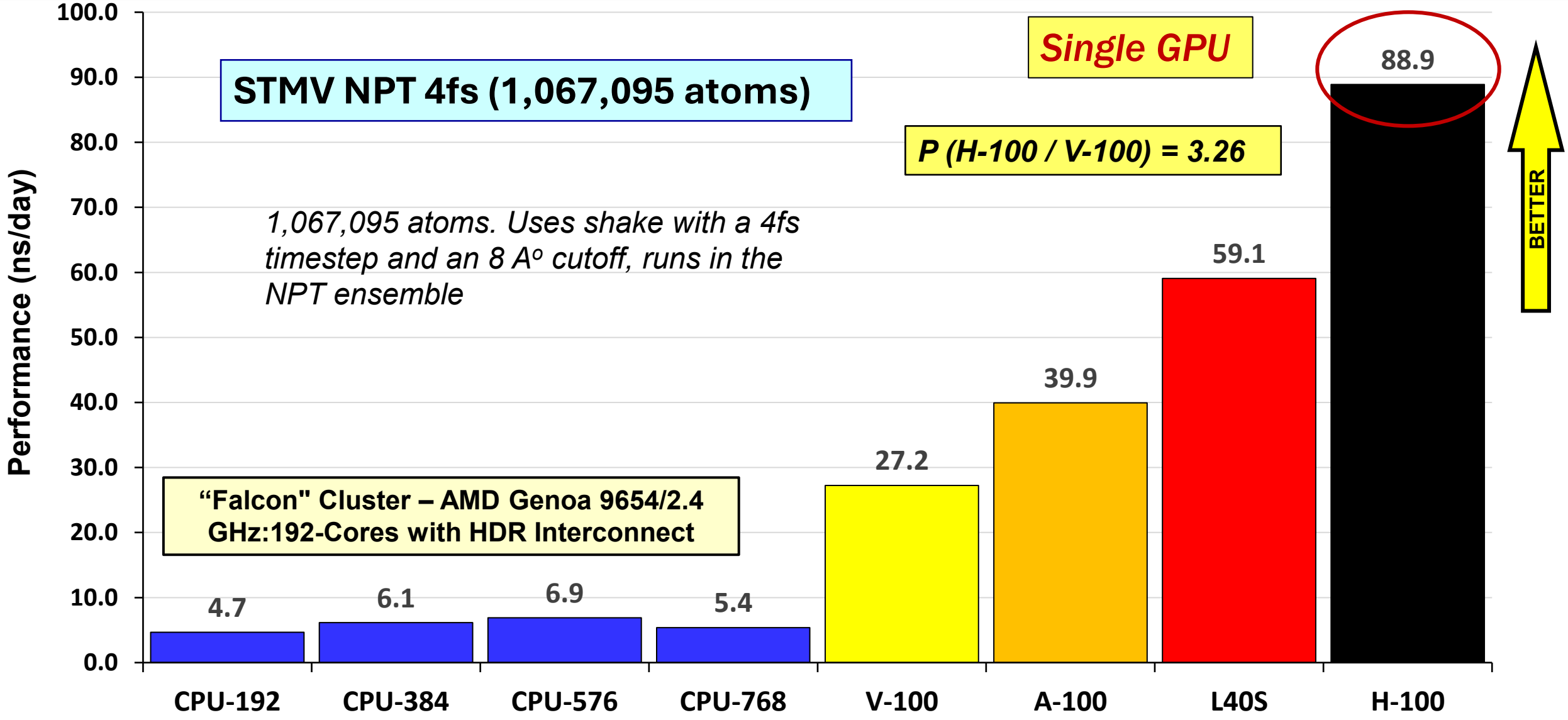
M27 - Cluster of 27 MUPs (657,585 atoms)

GPU Count	H100 Utilisation	L40S Utilisation
1	85.5%	89.2%
2	42.3%	44.1%
3	27.5%	27.8%
4	20.7%	20.3%

AMBER 24 – GPU Performance: M45 Test Case



AMBER 24 – GPU Performance: STMV NPT 4fs Test Case



AMBER 24 Performance Reports – GPU Utilisation Analysis II

Linaro Accelerator Analysis - Breakdown of how CUDA accelerators were used:

GPU utilization - % of time during which one or more kernels were executing on the GPU, averaged across available GPUs.

M45 - Cluster of 45 MUPs (932,751 atoms)

GPU Count	H100 Utilisation	L40S Utilisation
1	85.1%	87.8%
2	42.2%	42.6%
3	27.9%	27.7%
4	20.3%	20.0%

STMV NPT 4fs (1,067,095 atoms)

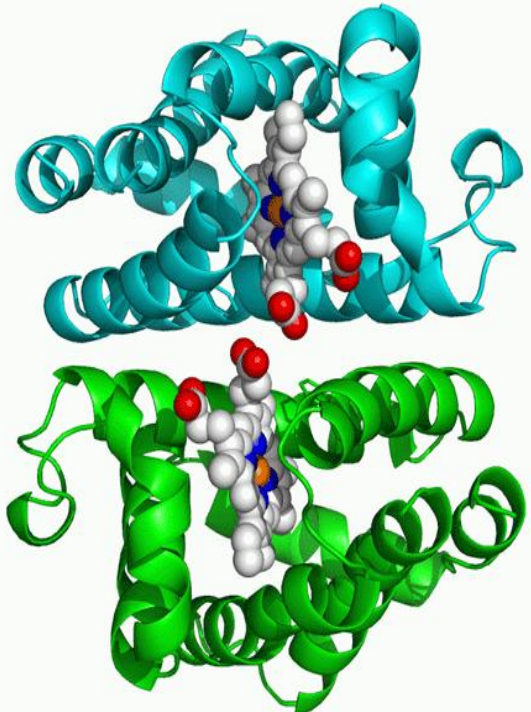
GPU Count	H100 Utilisation	L40S Utilisation
1	54.0%	56.4%
2	27.0%	26.8%
3	16.9%	18.7%
4	12.5%	13.0%

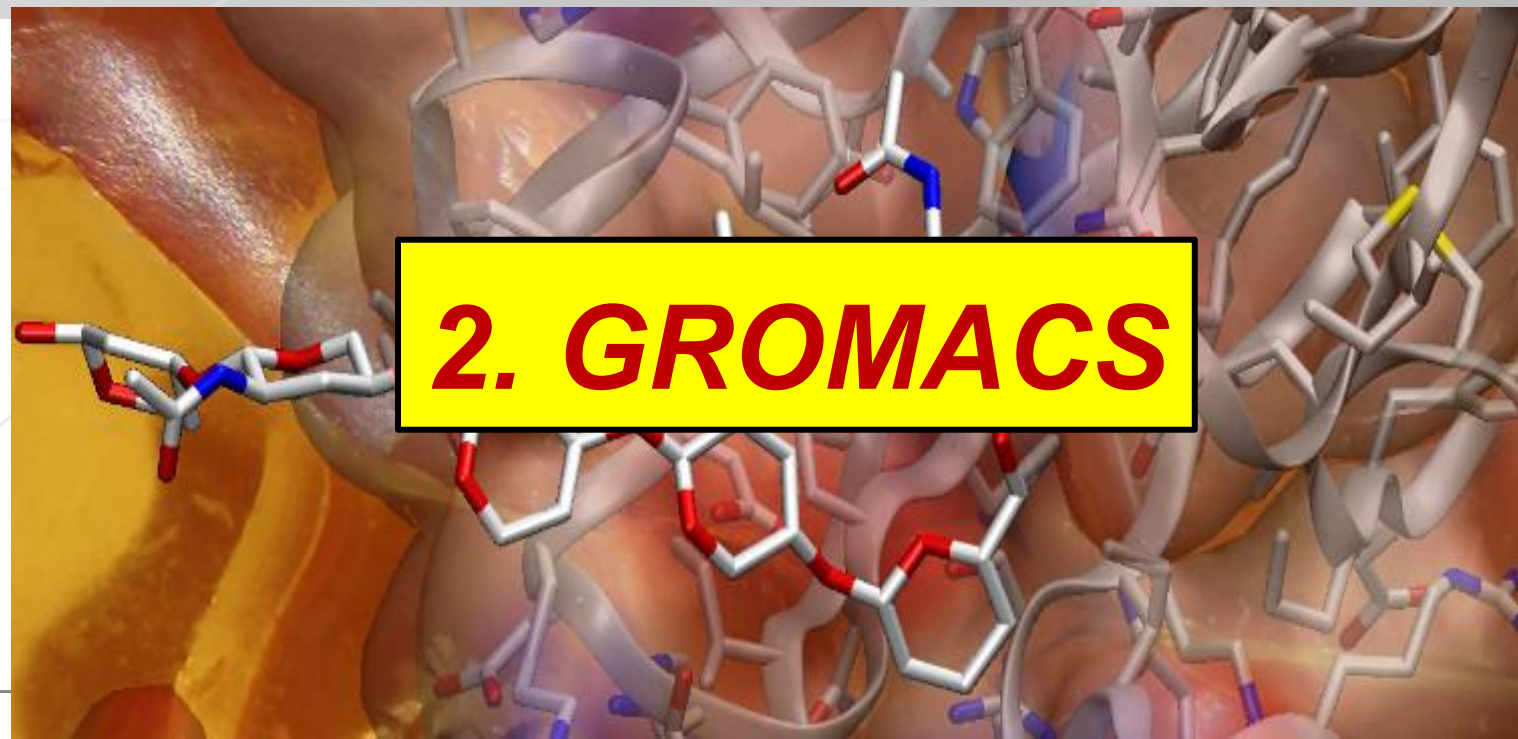
AMBER GPU Benchmark Takeaways (Exxact Corp.)

- In larger systems, the new NVIDIA Blackwell architecture GPUs outperform the previous generation GPUs.
  - NVIDIA GeForce **RTX 5090** offers the best performance for its cost, featuring high clock speeds and 32GB of memory. However, it lacks multi-GPU scalability due to its physical size, only suitable for **single-GPU workstations**.
  - The **NVIDIA RTX PRO 6000** Blackwell Max-Q Workstation Edition is impressive for large simulation sizes. In smaller simulations, its lower clock speed showcases poorer results since it can't effectively ramp up to fully take advantage of this GPU's performance compared to the consumer counterpart RTX 5090 and last generation NVIDIA RTX 6000 Ada.
- With **smaller simulation sizes (< 100,000K atoms),** the RTX PRO 4500 Blackwell is a great option, matching the popular NVIDIA RTX 5000 Ada, but at a cheaper price.
- The **B200 SXM** has impressive performance, but is **expensive for MD**. The B200 SXM, GH200, and H100 PCIe are all **geared towards AI workloads**; their high price tag makes them **not the best price-to-performance friendly option for "just" MD simulations.**

https://www.exxactcorp.com/blog/molecular-dynamics/amber-molecular-dynamics-nvidia-gpu-benchmarks

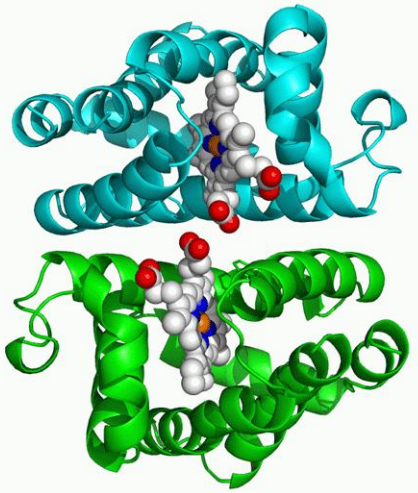
The Performance of Community GPU Codes in Life Sciences and Genomics





Molecular Simulation - GROMACS

GROMACS (GROningen MAchine for Chemical Simulations) is a molecular dynamics package designed for simulations of proteins, lipids and nucleic acids [University of Groningen] .



http://manual.gromacs.org/documentation/

Versions under Test:

Version 5.0.7 – 14 October 2015

Version 2016.3 – 14 March 2017

Version 2018.2 – 14 June 2018

Version 2019.6 – 28 February 2020

Version 2020.1 – 3 March 2020

Version 2023.1 – 21 April 2023

Version 2024.4 – 31 October 2024

Szilárd Páll, Artem Zhmurov; Paul Bauer, Mark Abraham, Magnus Lundborg, Alan Gray; Berk Hess, Erik Lindah, "Classical Molecular Dynamics (MD) Simulations: Codes, Algorithms, Force fields, and Applications", Chemical Physics Software Collection", *J. Chem. Phys.* 153, 134110 (2020)

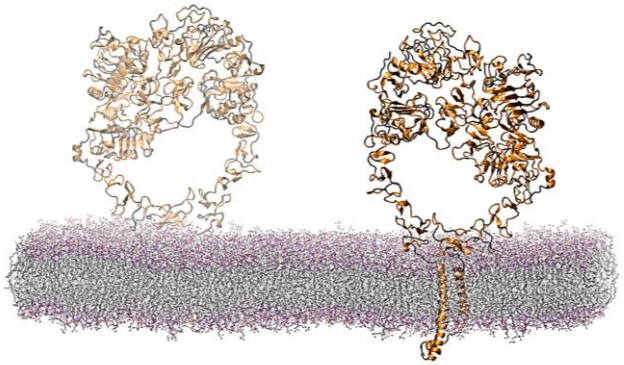
M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess, E. Lindahl, "GROMACS: High performance molecular simulations through multi-level, parallelism from laptops to supercomputers", SoftwareX (2015), DOI: 10.1016/j.softx.2015.06.001

Berk Hess et al. "*GROMACS 4: Algorithms for Highly Efficient, Load-Balanced, and Scalable Molecular Simulation*". *Journal of Chemical Theory and Computation* 4 (3): 435–447.

GROMACS Benchmark Cases

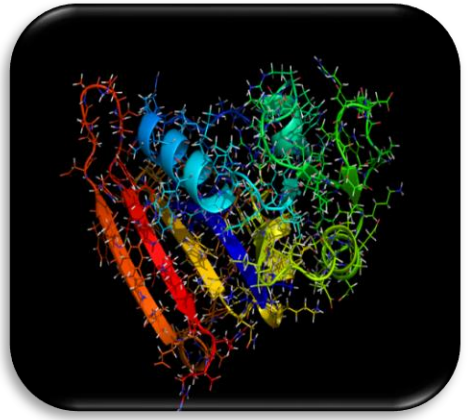
The HECBioSim Benchmarks

- PME simulation for **1.4M atom system** - Pair of Human Epidermal Growth Factor Receptor (hEGFR) Dimers of 1IVO and 1NQL, Total # atoms = **1,403,182** [Protein atoms, 43,498 ,Lipid atoms, 235,304 Water atoms, 1,123,392, Ions, 986



Lignocellulose

- Gromacs Test Case B - A model of cellulose and lignocellulosic biomass in an aqueous solution. This system of **3.3M atoms** is inhomogeneous and uses **reaction-field electrostatics** instead of PME and therefore should scale well.

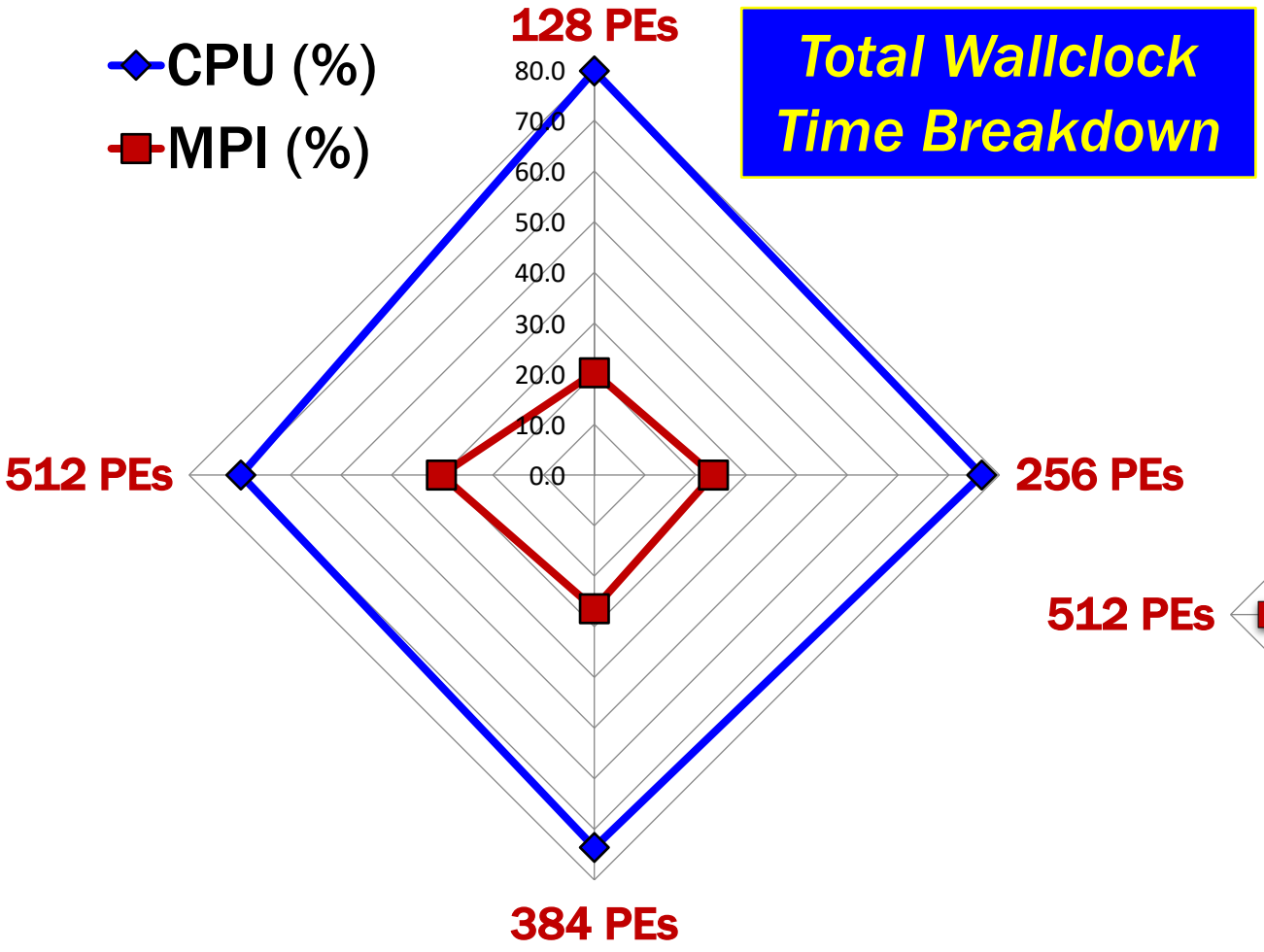


STMV

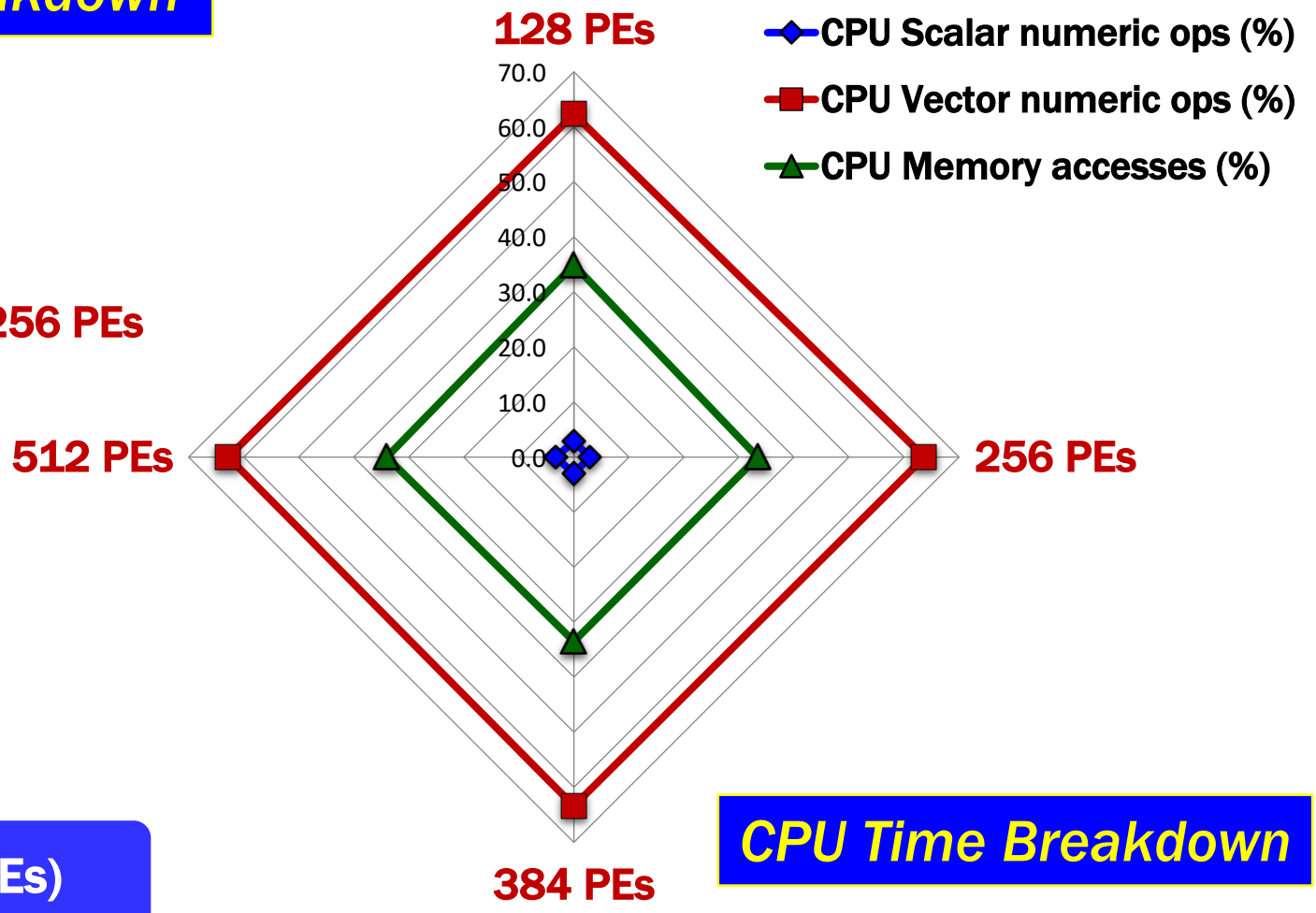
- **Satellite Tobacco Mosaic Virus** solvated in TIP3P using cubic box and the CHARMM27 force field. **1,066,628 atoms**, 2 fs time-step, 1.2 nm cut-offs, h-bond constraints, 0.15 nm PME grid spacing, NVT ensemble.

```
gmx mdrun -v -nsteps 20000 -resetstep 10000 -noconfout  -ntmpi $SLURM_NTASKS \
-ntomp $SLURM_CPUS_PER_TASK \  -nb gpu -bonded gpu -pme gpu -npme 1 \
 -nstlist 400  -g ${MYLOGS}/GROMACS.$GPU.$CASE.${SLURM_JOBID}.$irep
```

GROMACS 2024 – HECBioSim Performance Report



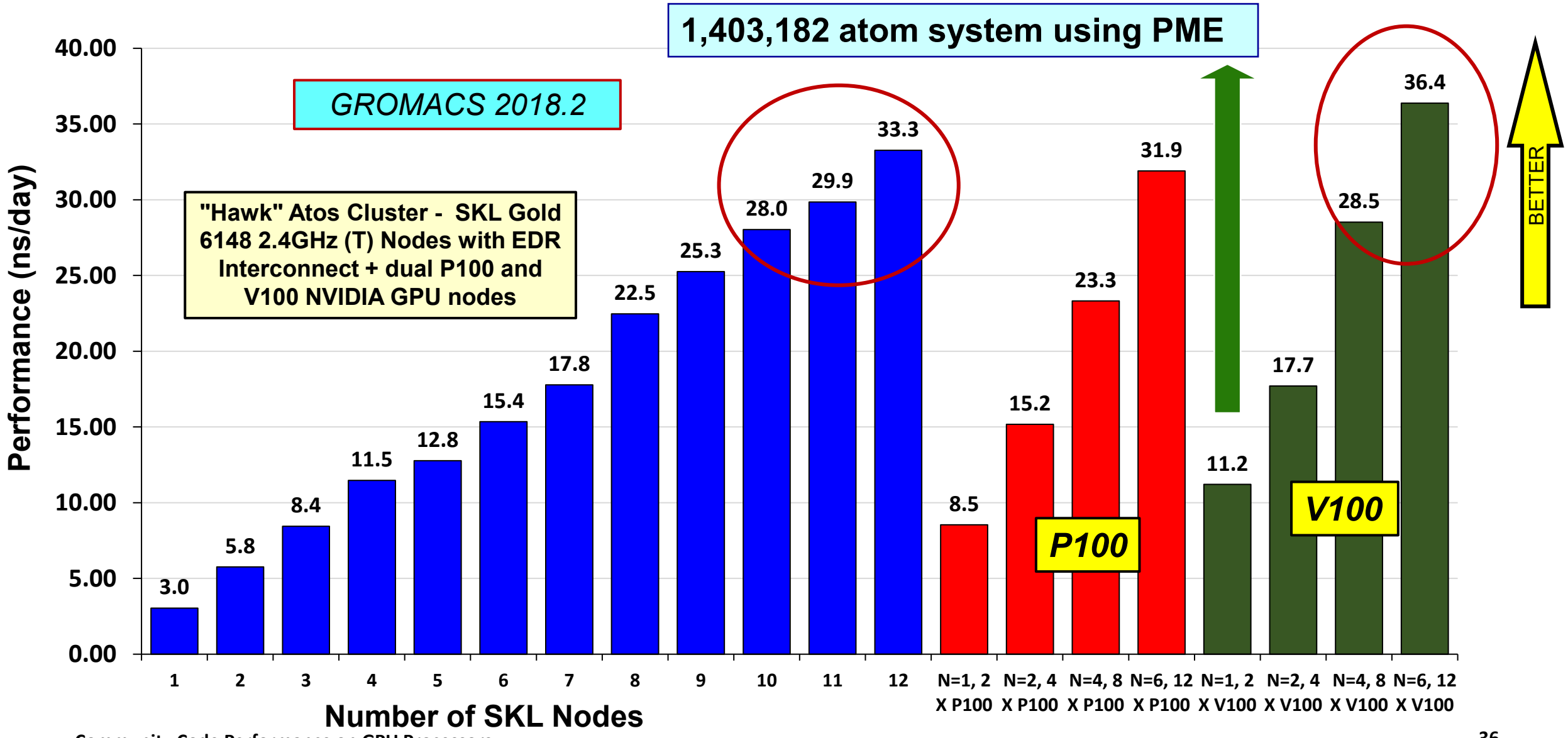
Total Wallclock Time Breakdown



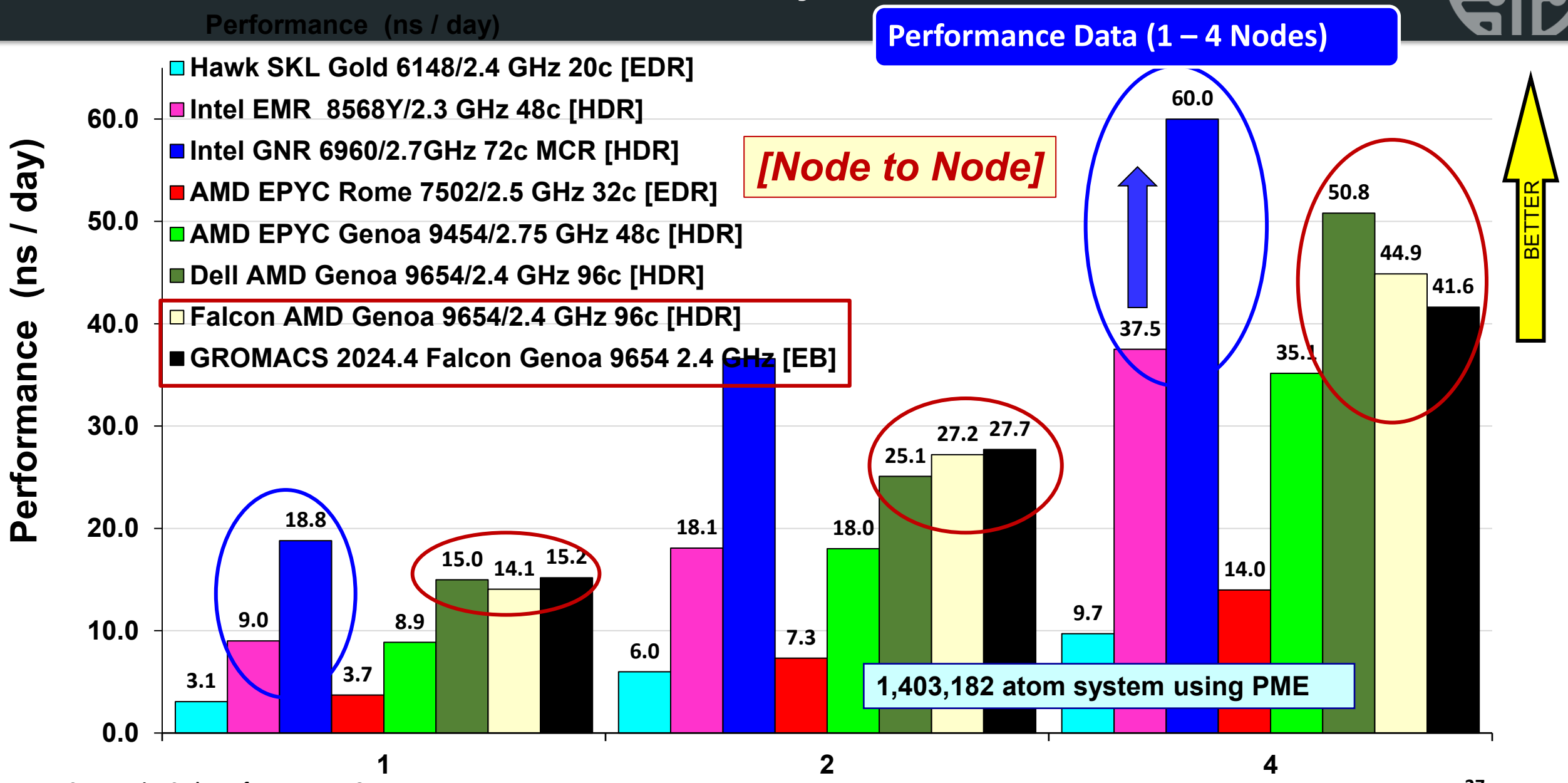
CPU Time Breakdown

Performance Data (128-512 PEs)

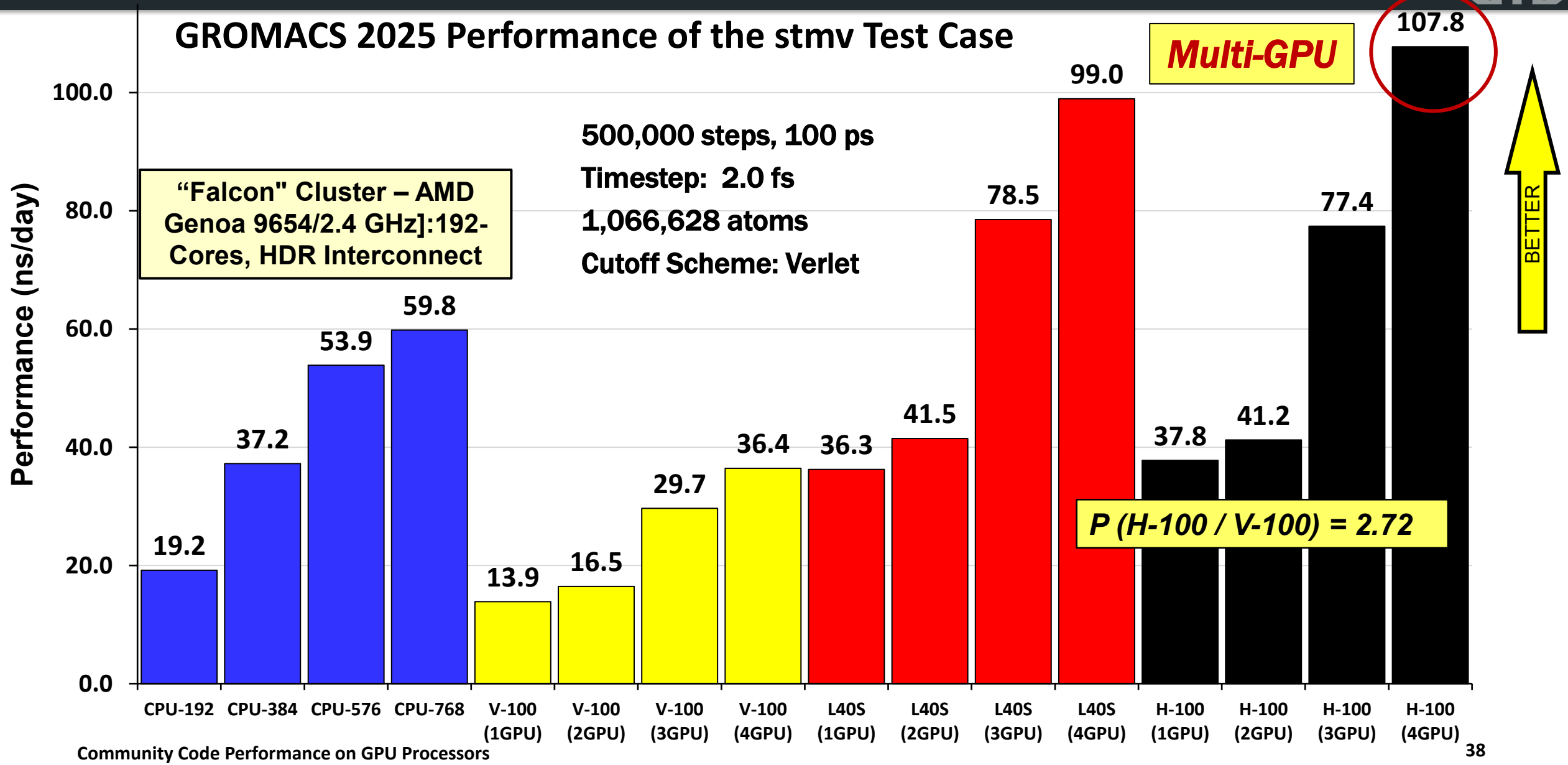
GROMACS 2018.2 – GPU Performance: HECBioSim Simulation



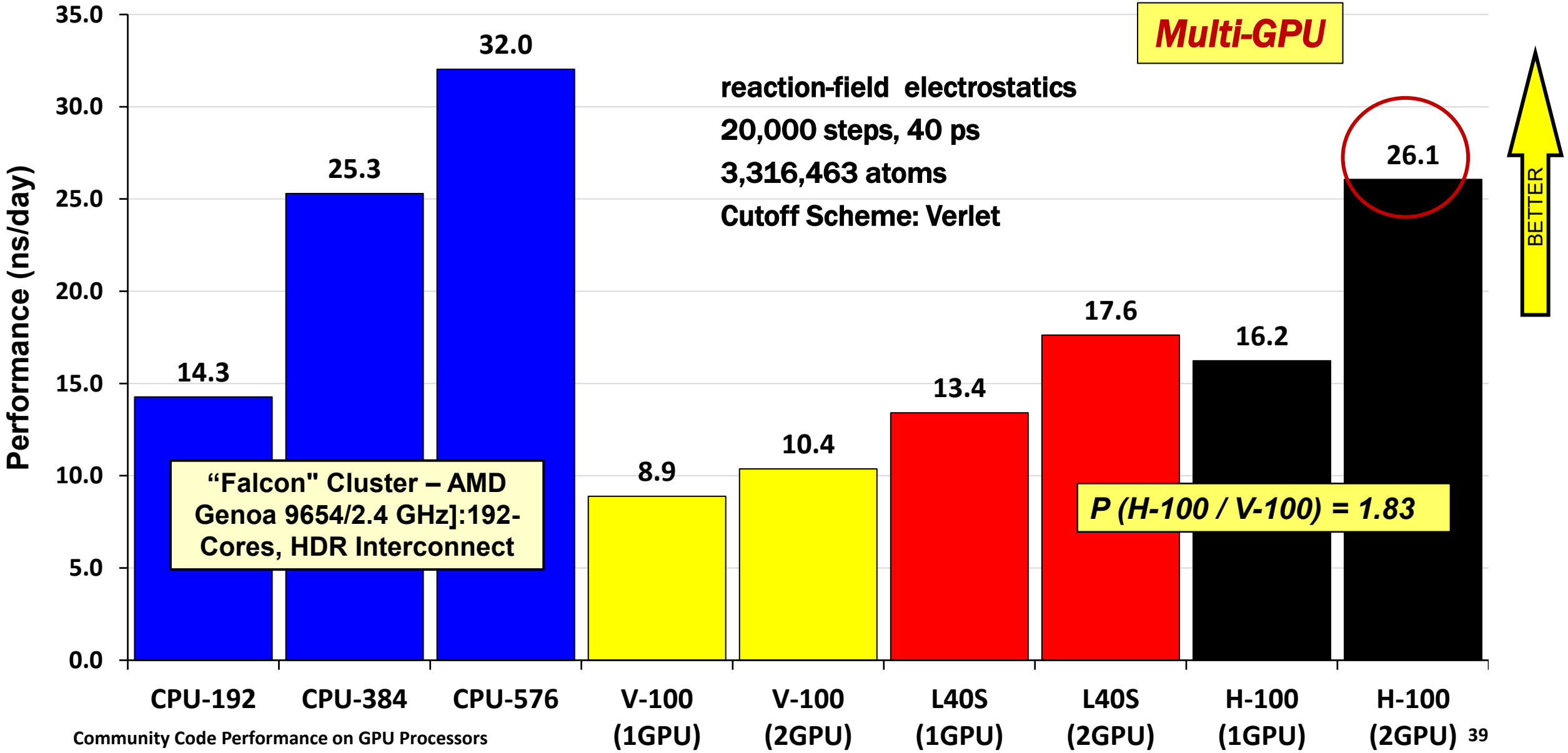
GROMACS – HECBioSim 1.4M Atom System



GROMACS 2025 Performance: stmv



GROMACS 2025 Performance: lignocellulose



GROMACS 2025 Performance Reports – GPU Utilisation Analysis

Linaro Accelerator Analysis - Breakdown of how CUDA accelerators were used:

GPU utilization - % of time during which one or more kernels were executing on the GPU, averaged across available GPUs.

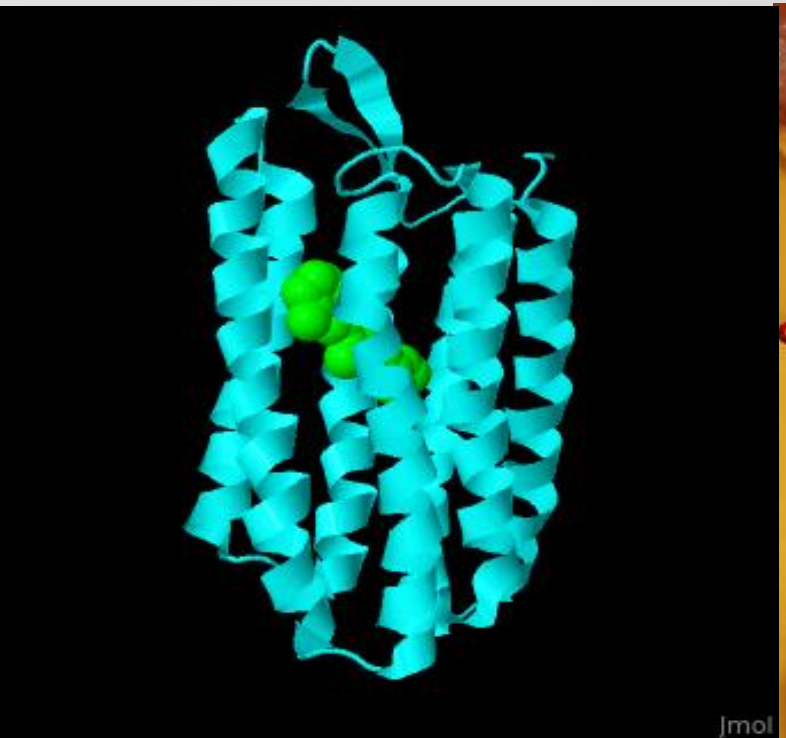
STMV - 500,000 steps, 100 ps: 1,066,628 atoms Timestep: 2.0 fs; Cutoff Scheme: Verlet

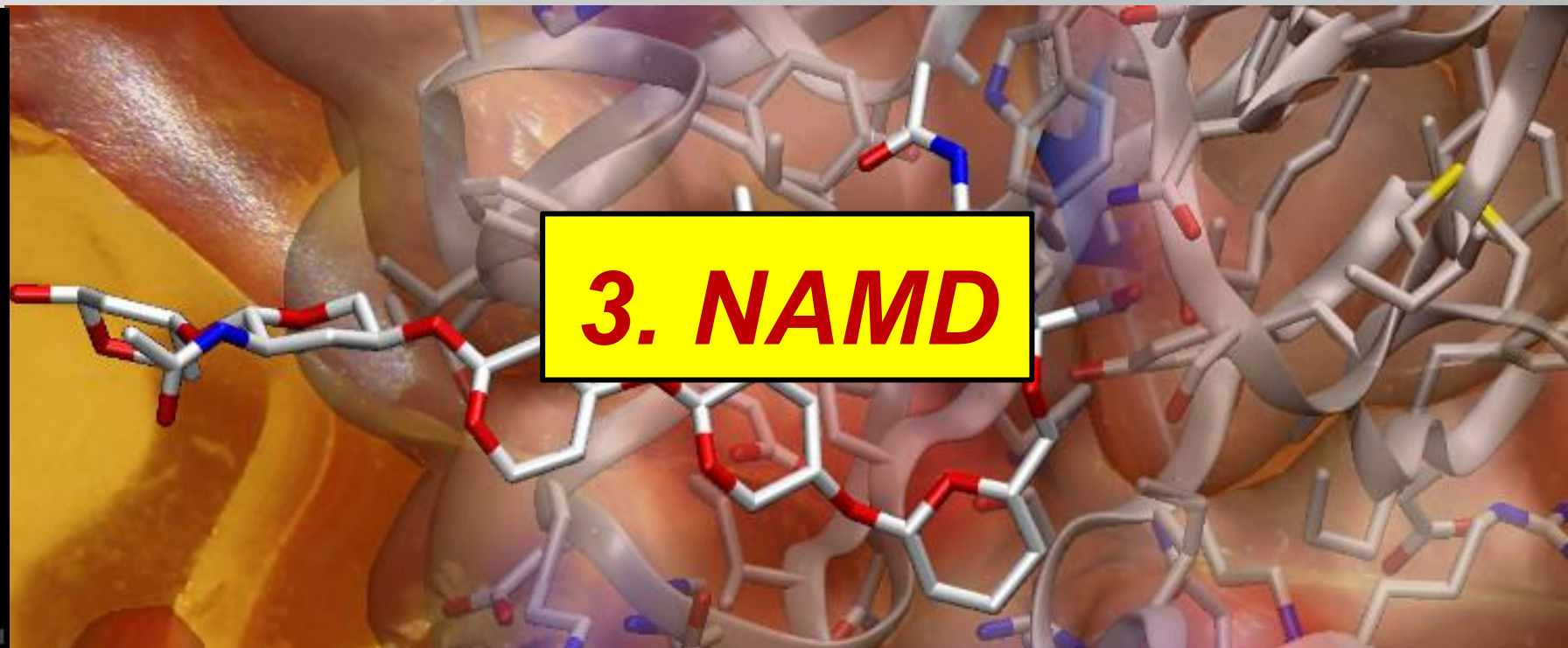
GPU Count	H100 Utilisation	L40S Utilisation
1	91.6%	88.8%
2	50.3%	53.3%
3	60.7%	63.0%
4	64.0%	65.0%

LIGNOCELLULOSE - reaction-field electrostatics; 20,000 steps, 40 ps; 3,316,463 atoms; Cutoff Scheme: Verlet

GPU Count	H100 Utilisation	L40S Utilisation
1	87.9%	76.5%
2	70.1%	51.6%
3	59.5%	43.7%
4	55.7%	40.0%

The Performance of Community GPU Codes in Life Sciences and Genomics





NAMD

NAMD, recipient of the 2020 Gordon Bell Prize, is a parallel molecular dynamics code designed for high-performance simulation of large biomolecular systems. Based on Charm++ parallel objects, NAMD scales to hundreds of cores for typical simulations and beyond 500,000 cores for the largest simulations. NAMD uses the popular molecular graphics program VMD for simulation setup and trajectory analysis, but is also file-compatible with AMBER, CHARMM, and X-PLOR. tutorials show how to use NAMD and VMD for biomolecular modelling.

https://www.ks.uiuc.edu/Research/namd/

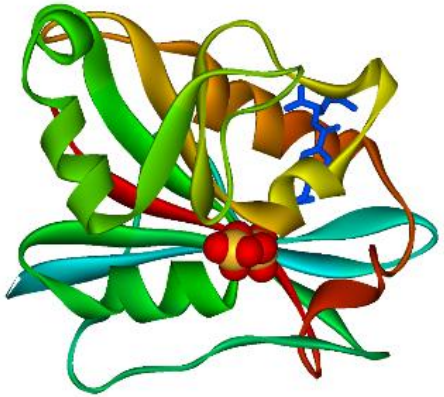
NAMD 3.0 beta 6 GPU-resident benchmarking results

NAMD 3.0 has a new **GPU-resident simulation mode** available, for multicore and netlrts builds, that **maximizes performance of small- to medium-sized molecular dynamics simulations, enabling more than 2x performance gain over the earlier GPU-offload mode**.

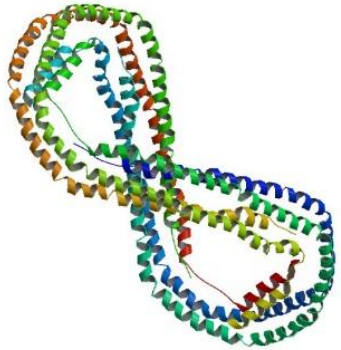
In addition to performing force calculations on GPU, GPU-resident mode performs **the numerical integration and rigid bond constraint calculations on GPU. Simulation data is maintained on the device between time steps, eliminating per-step host-device communication latencies**. With little work remaining on the CPU cores, the new GPU-resident mode no longer needs a large number of CPU cores to "keep up" with each GPU, reflected in the greater aggregate throughput performance now possible on GPU-dense architectures.

NAMD Benchmark Cases

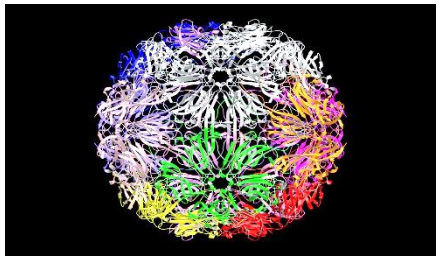
1. **dhfr** - Dihydrofolate reductase, **23,558** atoms - NVE (constant energy), multiple time stepping, 500,000 steps, Timestep: 2.0 fs



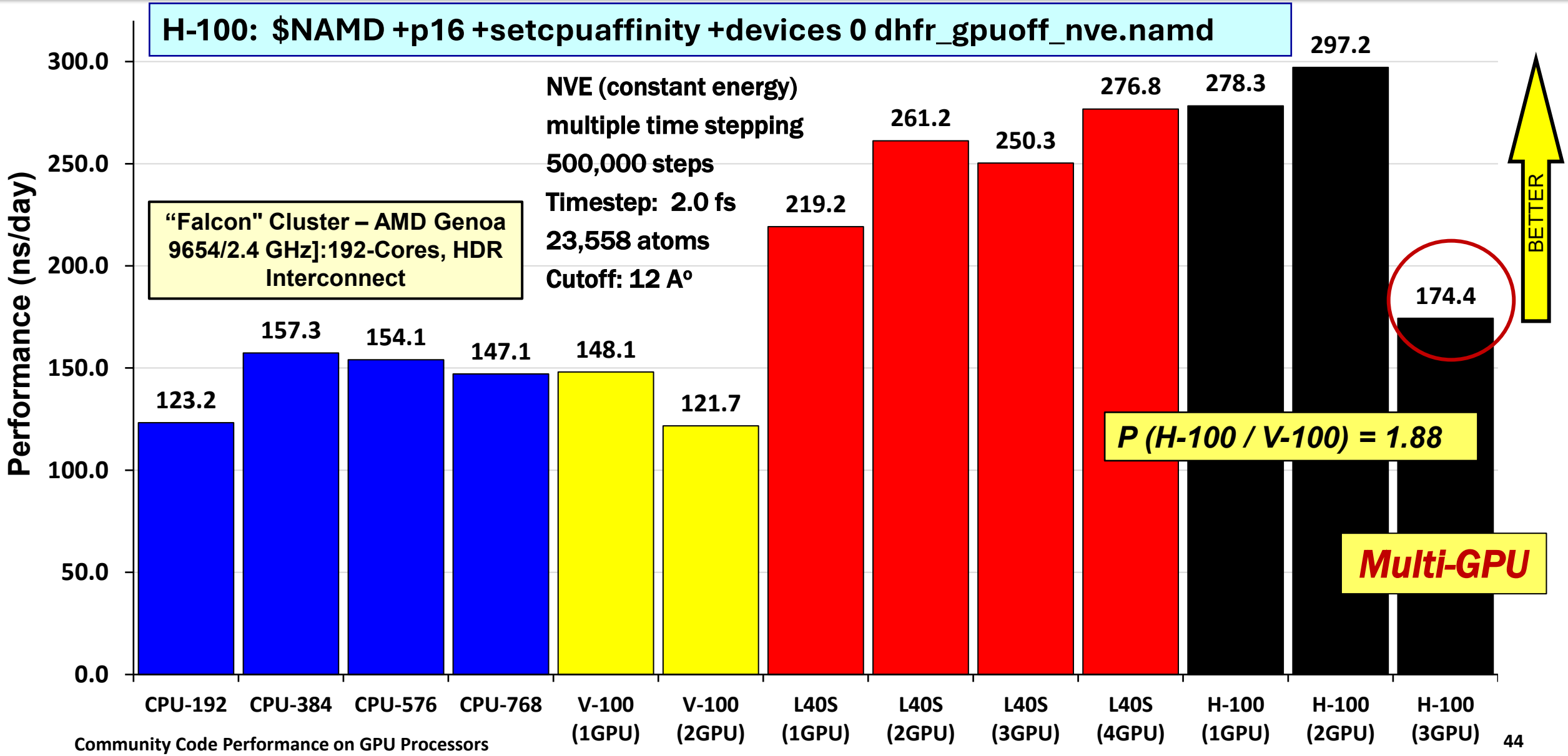
2. **apoa1** - Apolipoprotein A1, **92,224** atoms - NVE (constant energy), multiple time stepping, 50,000 steps, Timestep: 2.0 fs



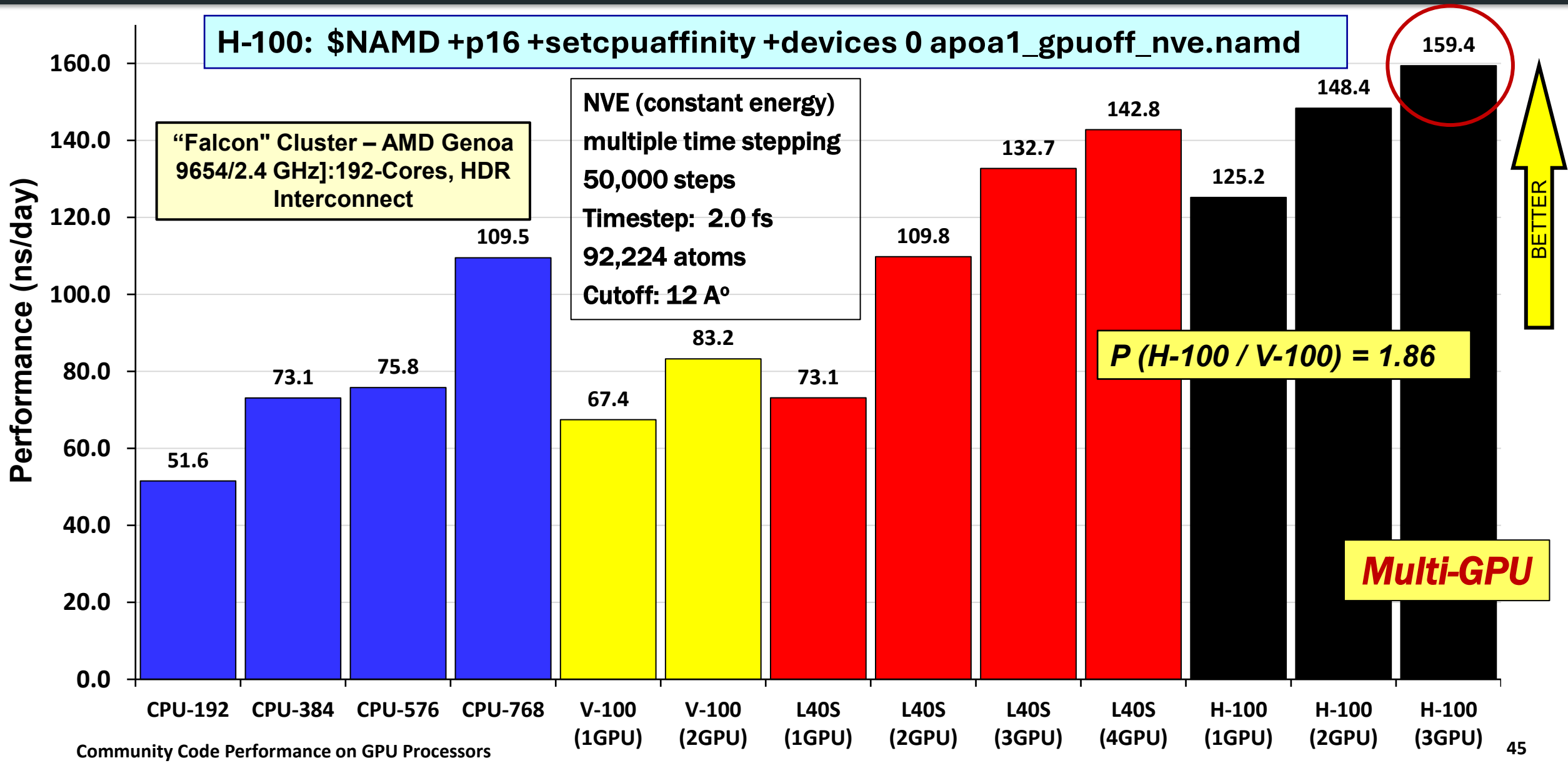
3. **stmv** - Satellite Tobacco Mosaic Virus benchmark (**1,066,628** atoms, periodic, PME), NVE (constant energy), 25,000 time steps, Timestep: 2.0 fs



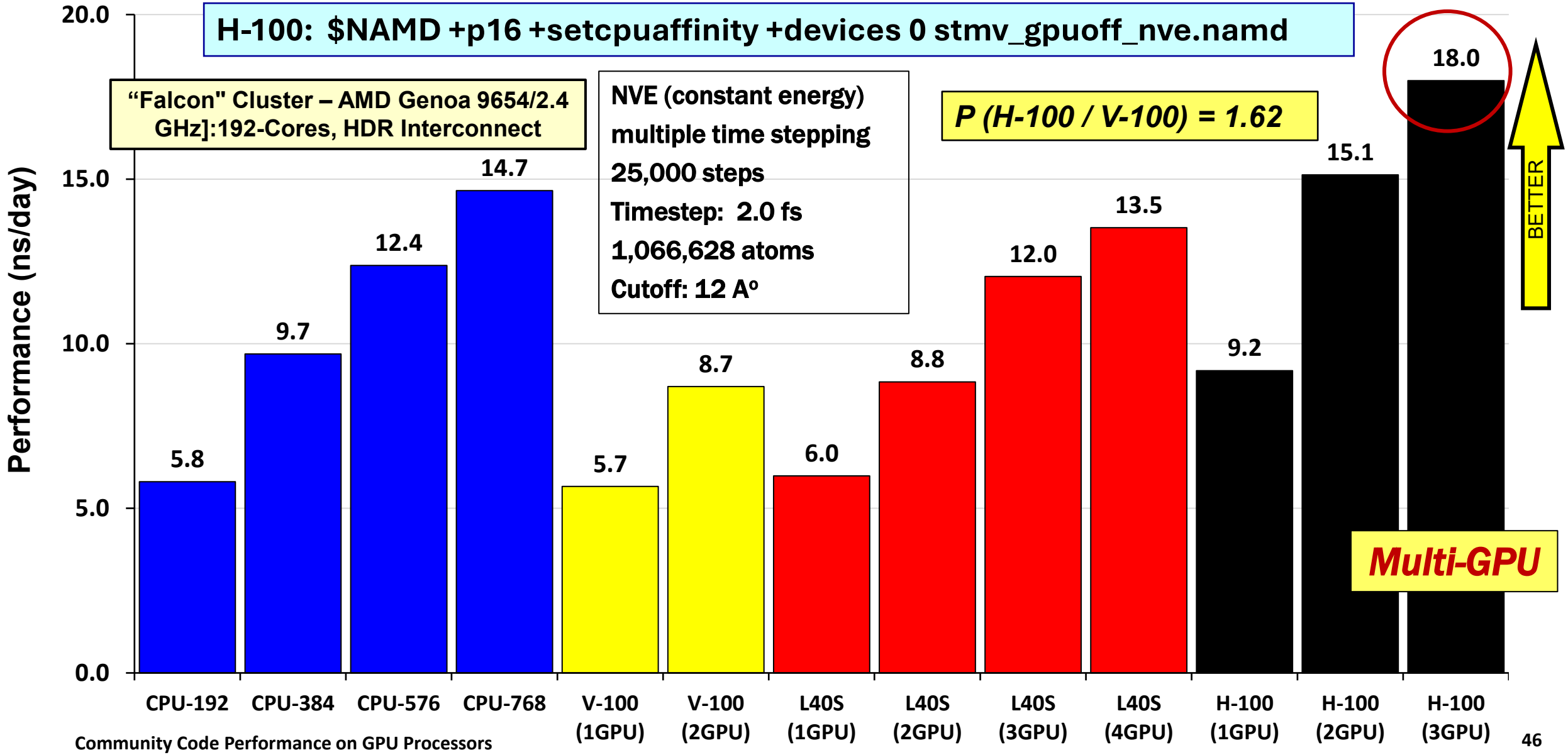
NAMD 3.0.2 – GPU-offload Performance: dhfr benchmark



NAMD 3.0.2 – GPU-offload Performance: apoa1 benchmark



NAMD 3.0.2 – GPU-offload Performance: stmv benchmark



NAMD 3.0.2 Performance Reports – GPU Utilisation Analysis

Linaro Accelerator Analysis - Breakdown of how CUDA accelerators were used:

GPU utilization - % of time during which one or more kernels were executing on the GPU, averaged across available GPUs.

DHFR NVE (constant energy) MTS: 23,558 atoms; 500,000 steps, Timestep: 2.0 fs, Cutoff: 12 A°

GPU Count	H100 Utilisation	L40S Utilisation
1	63.3%	54.2%
2	41.6%	49.7%
3	22.0%	45.4%
4	8.3%	33.8%

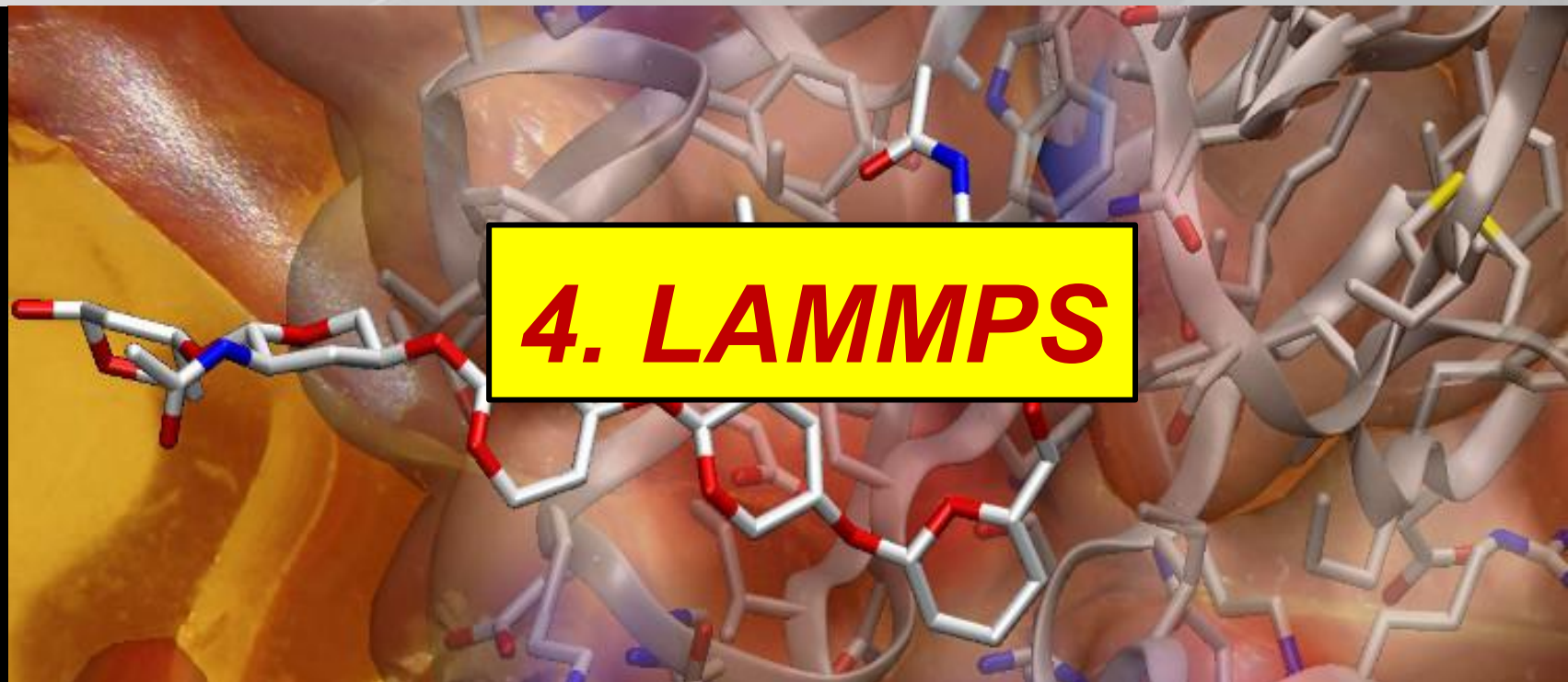
Apoa1 NVE (constant energy) MTS: 92,224 atoms; 50,000 steps, Timestep: 2.0 fs, Cutoff: 12 A°

GPU Count	H100 Utilisation	L40S Utilisation
1	50.2%	38.5%
2	40.0%	34.8%
3	35.5%	33.4%
4	17.4%	30.5%

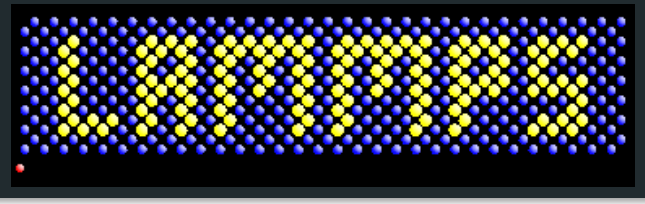
stmv NVE (constant energy) MTS: 1,066,628 atoms; 25,000 steps, Timestep: 2.0 fs, Cutoff: 12 A°

GPU Count	H100 Utilisation	L40S Utilisation
1	37.3%	32.3%
2	31.1%	25.3%
3	26.2%	22.9%
4	19.7%	20.6%

The Performance of Community GPU Codes in Life Sciences and Genomics



LAMMPS



- **LAMMPS** is one of the leading MD simulation codes, in part because of the flexibility and collection of capabilities that have grown through over 30 years of development and user contributions.
- The **GPU *package***, released as part of LAMMPS in 2010, took the common approach of simply offloading the force calculation, typically the most computationally expensive step. Nearly all other kernels run on the host CPU, requiring frequent data copies between host and device in every timestep. This method has clear drawbacks given the limited transfer speed and high latency between the separate memories of the CPU and the GPU.
- The **KOKKOS *package*** in LAMMPS uses the Kokkos *library* to achieve performance portability. The USER-CUDA package attempted to be "**GPU resident**", i.e. as many kernels as possible run on the GPU. However, data transfer is still necessary to maintain compatibility when using functionality not yet ported to GPUs. The **KOKKOS package** brought the additional promise of keeping LAMMPS vendor-agnostic (using GPUs from AMD, Intel, and NVIDIA).

S. Plimpton, *Fast Parallel Algorithms for Short-Range Molecular Dynamics*, J Comp Phys, 117, 1-19 (1995).

Richard Lawrence, James X. Mao, Honggao Liu *Comparison of GPU Performance Scaling for Molecular Dynamics*, PEARC '25: https://doi.org/10.1145/3708035.373608

Anders Johansson, Evan Weinberg, Christian R. Trott, Megan J. McCarthy, Stan G. Moore, *LAMMPS-KOKKOS: Performance Portable Molecular Dynamics Across Exascale Architectures*, arXiv:2508.13523

http://lammps.sandia.gov/index.html

LAMMPS Benchmark cases

Attention focused on the CPU and GPU version **LAMMPS/29Aug2024, with** three standard LAMMPS Benchmark cases used, with a focus on both **Weak and Strong Scaling:**

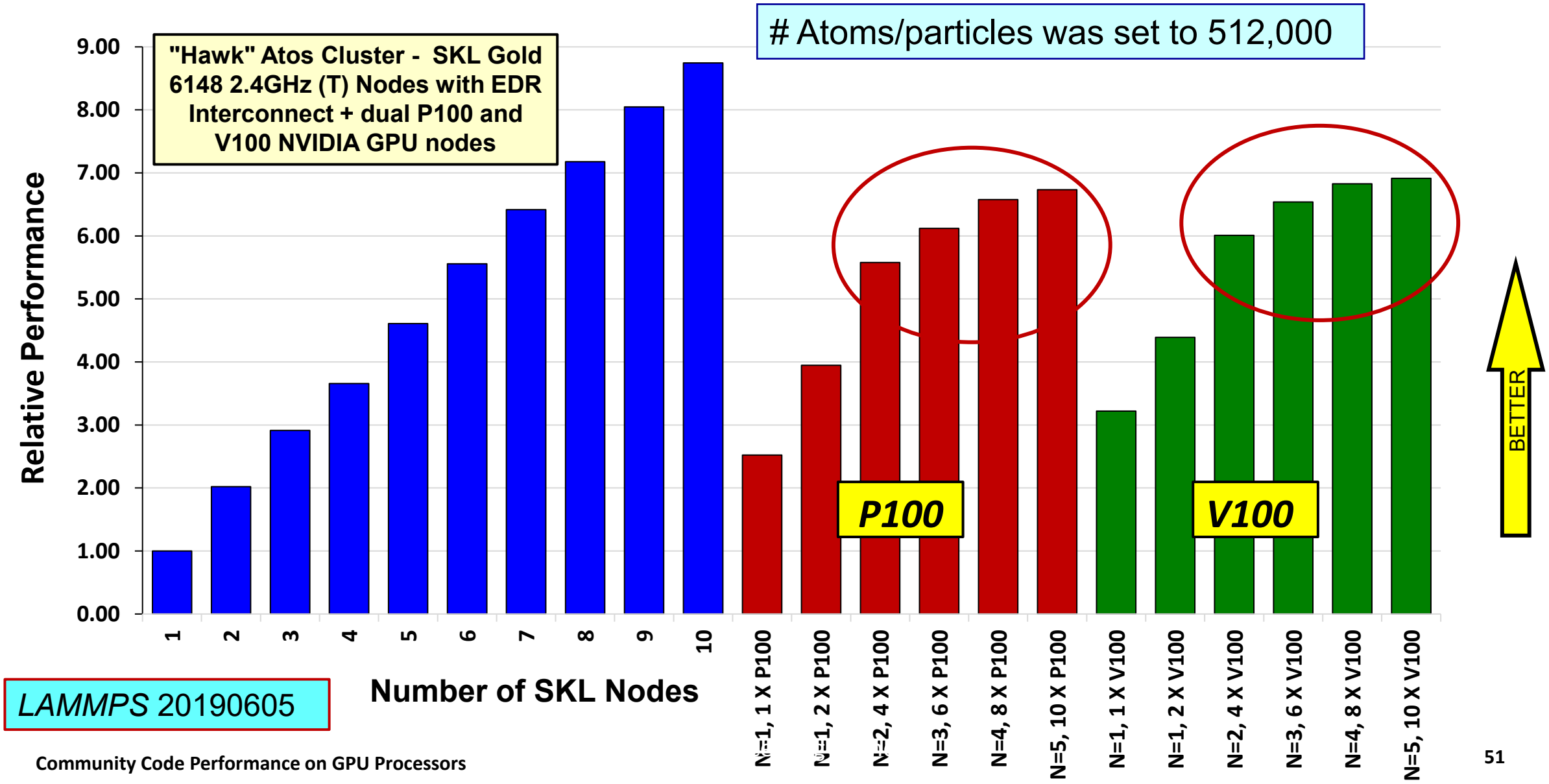
1. **Standard short-range Lennard-Jones (LJ) potential**;
2. **Full protein, Rhodopsin:** long-range forces evaluation using the LAMMPS default implementation of a particle mesh approach. Standard LAMMPS input files unaltered other than increasing # time steps from 100 to 10,000. The number of atoms/particles was set to 512,000 via:

   ./lmp_exe -suffix gpu -pk gpu $NGPUS -in in. rhodo.scaled -v x 4 -v y 2 -v z 2
3. **Reactive potential ReaxFF:** Models the reaction of crystalline Hexanitrostilbene (HNS)energetic material at the atomic scale. ReaxFF is an empirical model designed to support physical phenomena such as bond formation & breaking, polarization, & charge transfer effects.

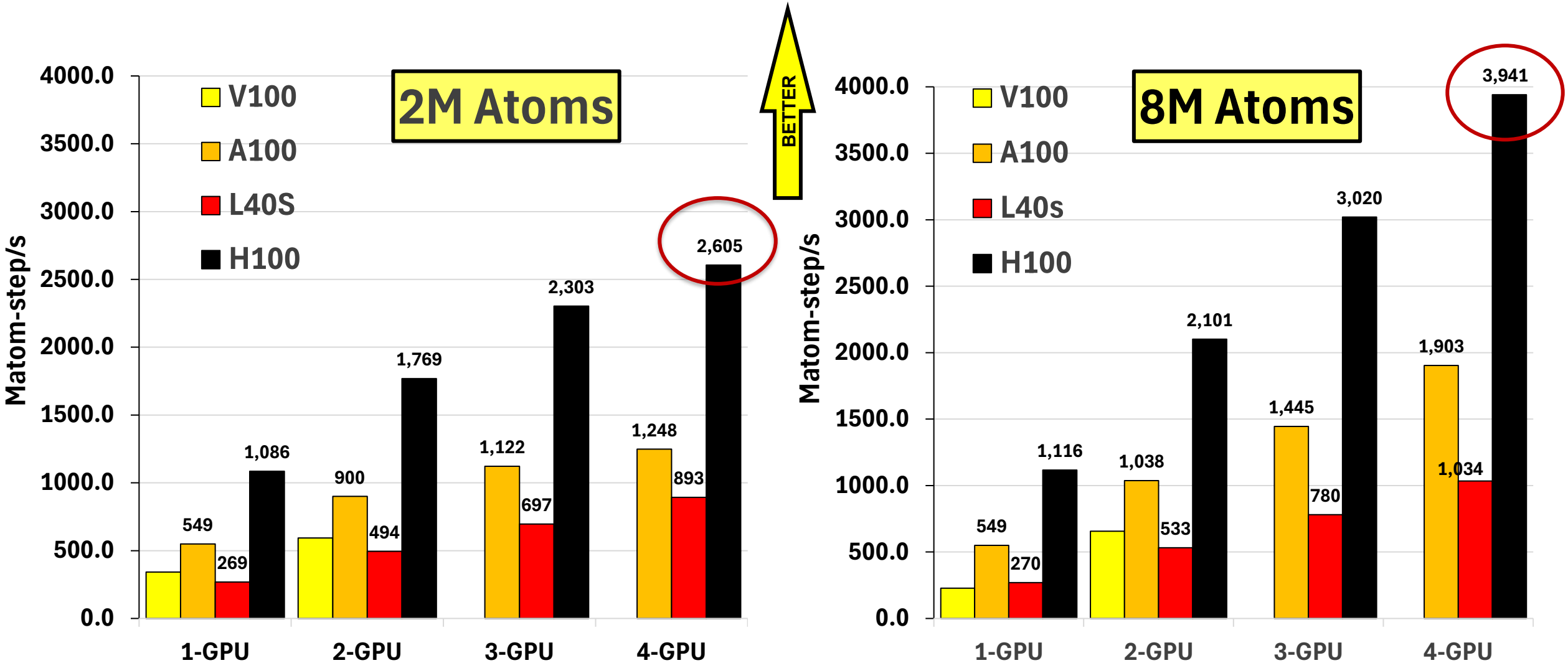
Comparison of GPU Performance Scaling for Molecular Dynamics, Richard Lawrence, James X. Mao, Honggao Liu, PEARC '25: Practice and Experience in Advanced Research Computing 2025: The Power of Collaboration, 30, pp.1 – 4, https://doi.org/10.1145/3708035.373608.

The units "**atom timesteps per second**" and "**atom timesteps per second per GPU**" are the primary metrics used to describe the performance of hardware acceleration for MD

LAMMPS – GPU Performance in Rhodopsin Simulation

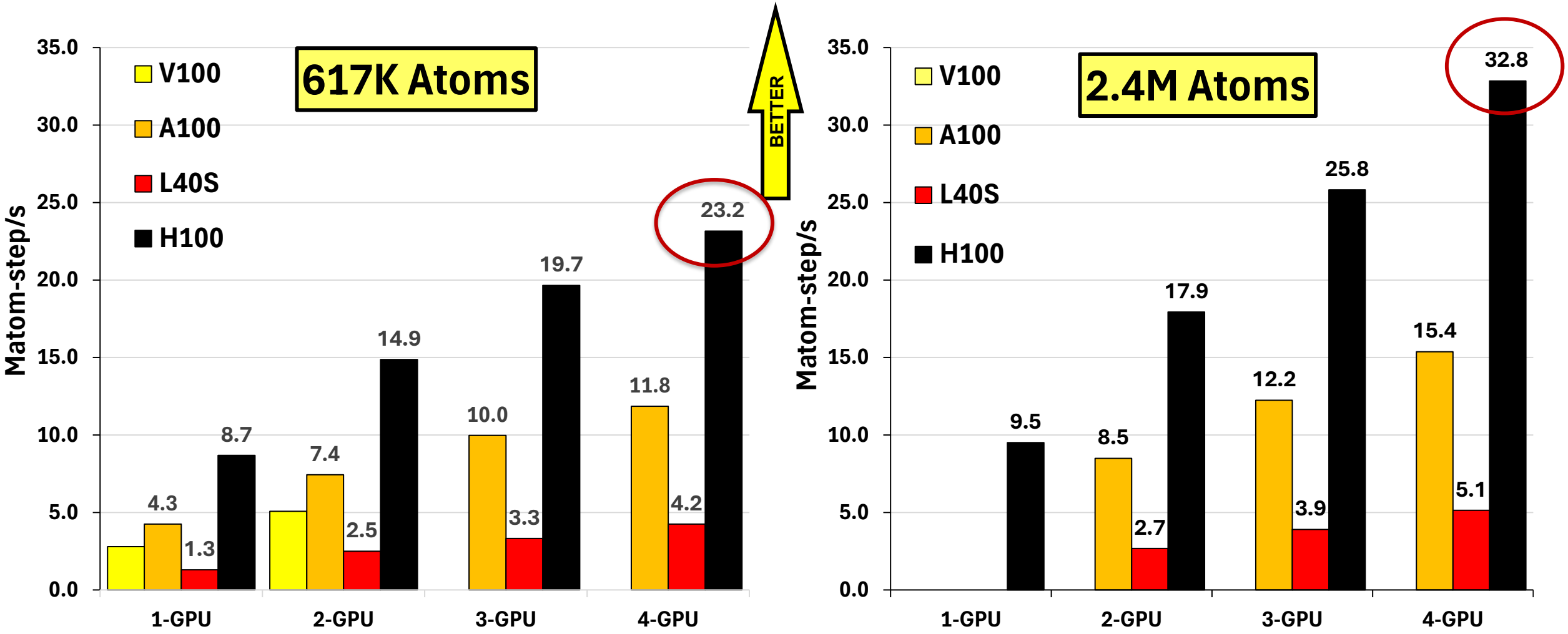


LAMMPS – Standard short-range Lennard-Jones (LJ) potential



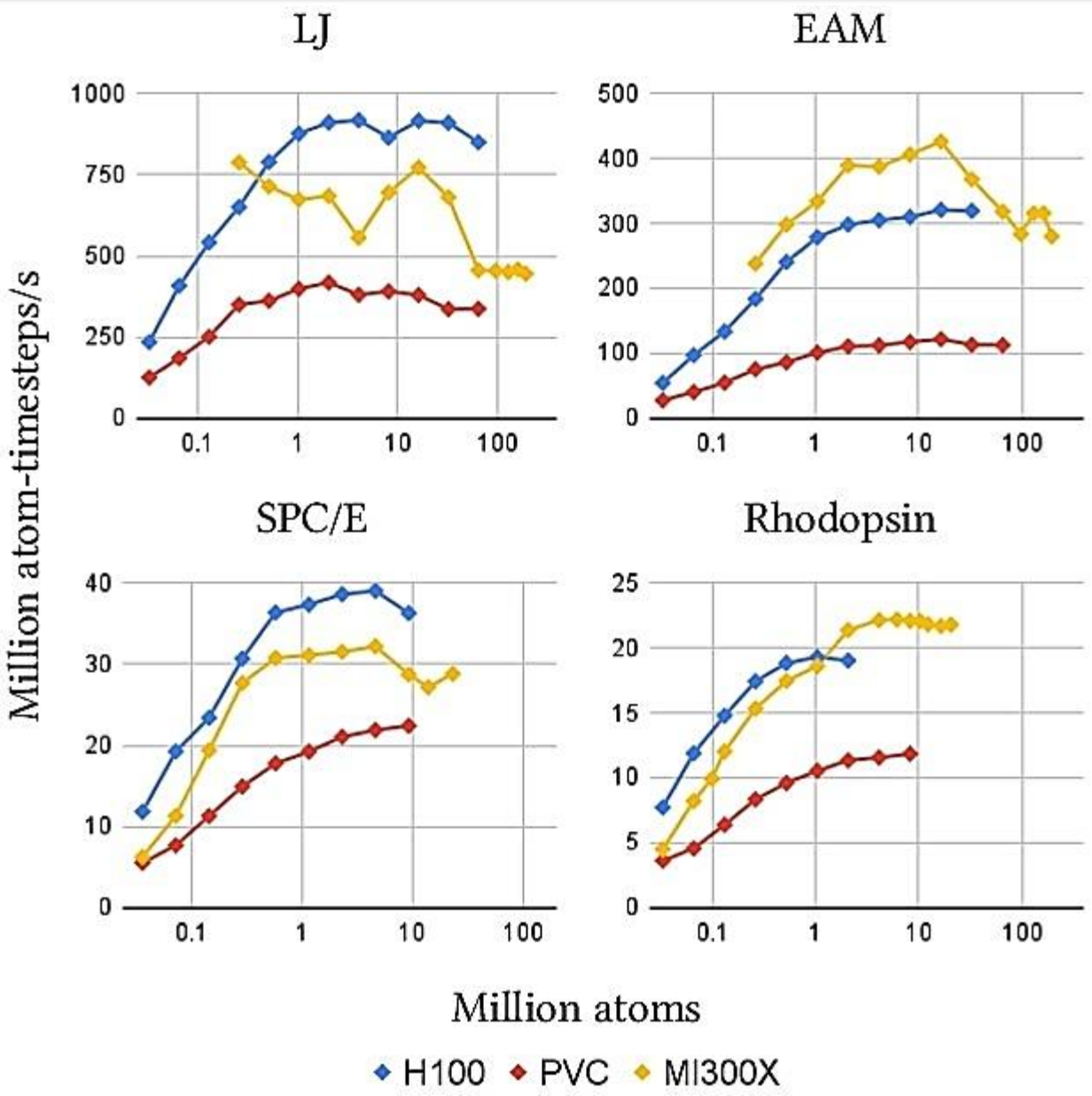
Poor performance of the L40S – absence of fp64 tensor cores ?

LAMMPS – Reactive potential ReaxFF



Poor performance of the L40S – absence of fp64 tensor cores ?

LAMMPS Benchmark cases – Strong Scaling

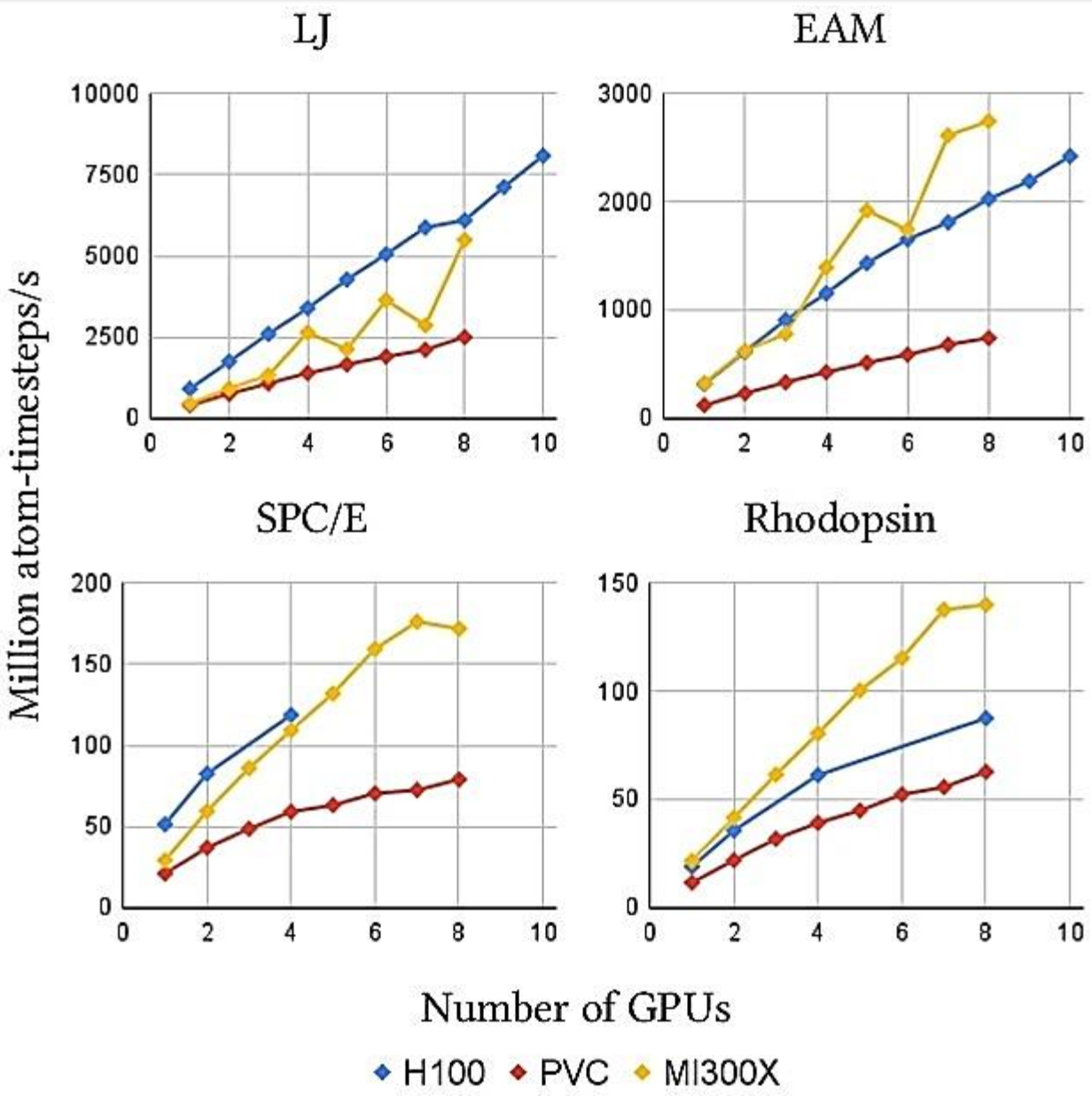


Comparison of GPU Performance Scaling for Molecular Dynamics, Richard Lawrence, James X. Mao, Honggao Liu, PEARC '25: Practice and Experience in Advanced Research Computing 2025: The Power of Collaboration, 30, pp.1 – 4, https://doi.org/10.1145/3708035.373608

Figure 1:

Four plots of strong scaling for LJ, EAM, Rhodopsin, and SPCE including performance on a single GPU of each of the three accelerators. (*higher is better*)

LAMMPS Benchmark cases – Strong Scaling

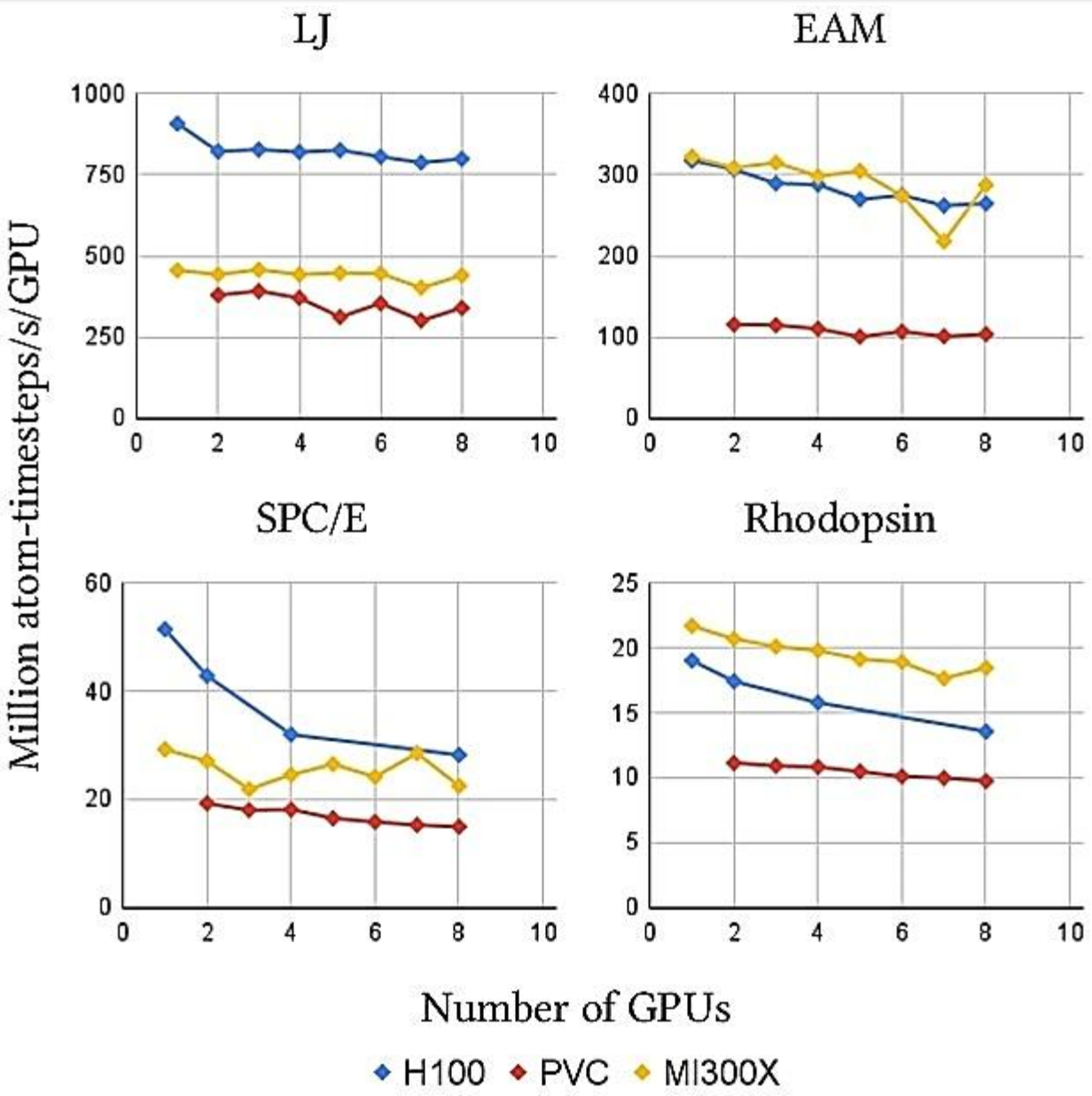


Comparison of GPU Performance Scaling for Molecular Dynamics, Richard Lawrence, James X. Mao, Honggao Liu, PEARC '25: Practice and Experience in Advanced Research Computing 2025: The Power of Collaboration, 30, pp.1 – 4, https://doi.org/10.1145/3708035.373608

Figure 2:

Four plots of strong scaling for LJ, EAM, Rhodopsin, and SPCE including performance on each of the three accelerators. (*higher is better*)

LAMMPS Benchmark cases – Weak Scaling



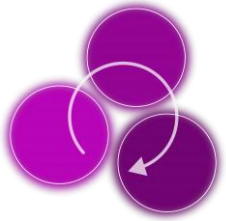
Comparison of GPU Performance Scaling for Molecular Dynamics, Richard Lawrence, James X. Mao, Honggao Liu, PEARC '25: Practice and Experience in Advanced Research Computing 2025: The Power of Collaboration, 30, pp.1 – 4, https://doi.org/10.1145/3708035.373608

Figure 3:

Four plots of weak scaling for LJ, EAM, Rhodopsin, and SPCE including performance on each of the three accelerators. (*higher is better*)

Summary and Conclusions

- Considered the performance of a number of widely used **GPU-enabled Molecular Dynamics (MD) community codes** important in understanding biological systems and drug interactions. The featured codes here – **AMBER, GROMACS, LAMMPS** and **NAMD** – were early and significant adopter of GPU acceleration and are widely used today on both local and regional UK clusters.
- **In Progress: Demonstrated the performance improvements of the selected codes on NVIDIA's V100, A100, L40S and H100 GPUs currently** available on the **Supercomputing Wales** systems, compared, for example, to those seen on **CPU systems (AMD Genoa 9654).** Additional insight provided from use of **Linaro's Performance Reports.**
- **Impressive 1-GPU performance of AMBER … issues with GROMACS & NAMD.**
- Note again that this work was in part undertaken under the auspices of the UKRI funded **SHAREing project** ("*Skills Hub for Accelerated Research Environments Inspiring the Next Generation*").



https://developer.nvidia.com/hpc-application-performance