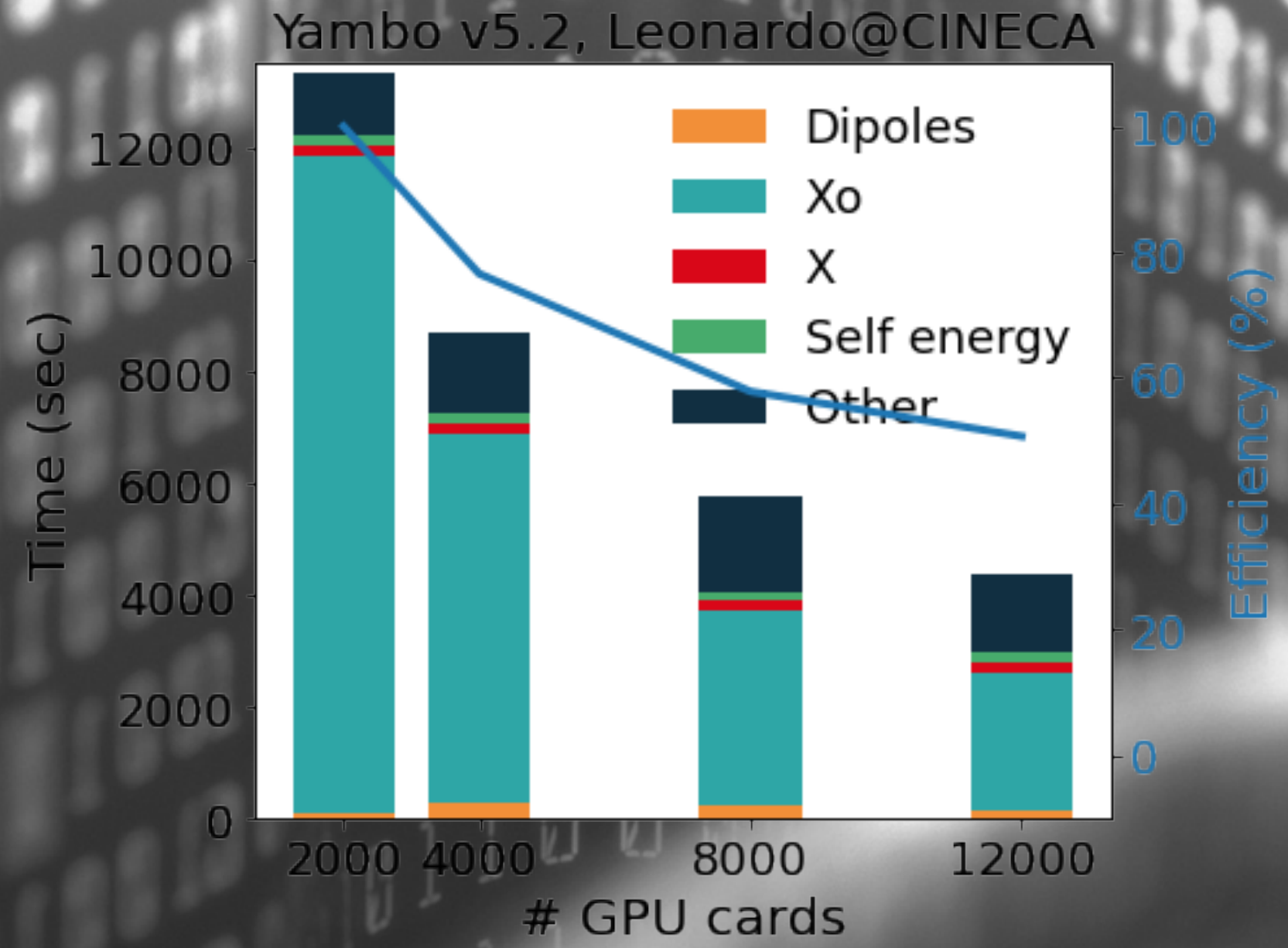


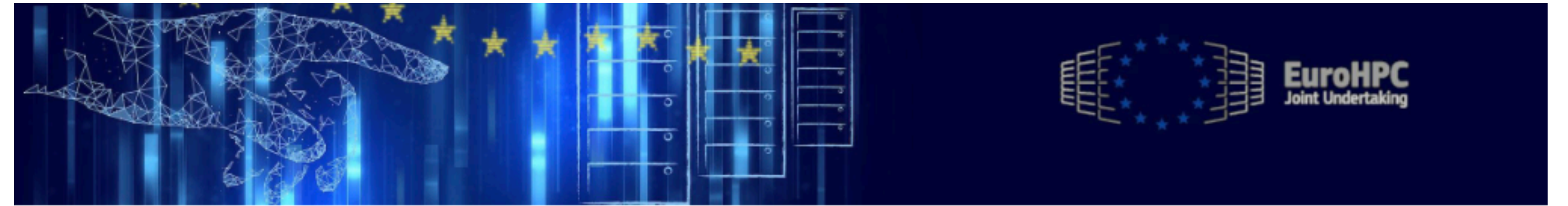
# Code Benchmarking and Performance Assessment from the Materials Science field

Andrea Ferretti  
[CNR-NANO, Modena, Italy]



# Applying for resources

---



## Project Scope and Plan – Regular Access

---

### 2.4 Software and Attributes (1 page)

---

*(Please see also **Examples of Performance Reporting** in Section 2.6.2.1). Describe the software that will be used including a discussion of the state of the art in the field. The description should mention:*

#### 2.4.1 Software

---

*Please describe all codes you are using in the proposal. Justify your choices and describe alternatives (if any).*

<Enter your text here>

#### 2.4.2 Particular libraries

---

*Describe particular libraries required by the production and analysis software, algorithms and numerical techniques employed (e.g. finite element, iterative solver), programming languages. Please specify requirements for compilation or build environment (build system (e.g. cmake, python version), version control system (e.g. git, subversion) etc.).*

<Enter your text here>

#### 2.4.3 Parallel programming

---

*Model(s) used (e.g., MPI, OpenMP/Pthreads, CUDA, OpenACC, etc.).*

<Enter your text here>

Relevant technical info required

# Applying for resources

---



## Project Scope and Plan – Regular Access

---

### 2.6 Performance of Software (Maximum 2 pages)

---

#### 2.6.1 Testing of your code on the requested machine

---

*It is strongly recommended that your production code is tested in the requested machine(s) (see also the text referring to Benchmark Access in the Important Notice at the top of page 2). Please specify the EuroHPC Benchmark Access project (if any, by providing their "EHPC-BEN-" ID) or other projects (previous PRACE calls, national calls, etc.) used to prepare the Regular Access proposal.*

*If the preparatory host machine is different from the target machine, please specify why you think the data is relevant. In this case, briefly report the conversion factor (in terms of ratio of time to solution, flops or requested node hours) between the machine where the preparatory tests were performed and the requested system. Moreover, your proposal must account for all technical constraints and requirements of the targeted machine(s) as documented in the separate Technical Guidelines for Applicants document. Failure to do so will significantly increase the risk of your project being technically rejected.*

<Enter your text here>

#### 2.6.2 Quantify the HPC performance of your project

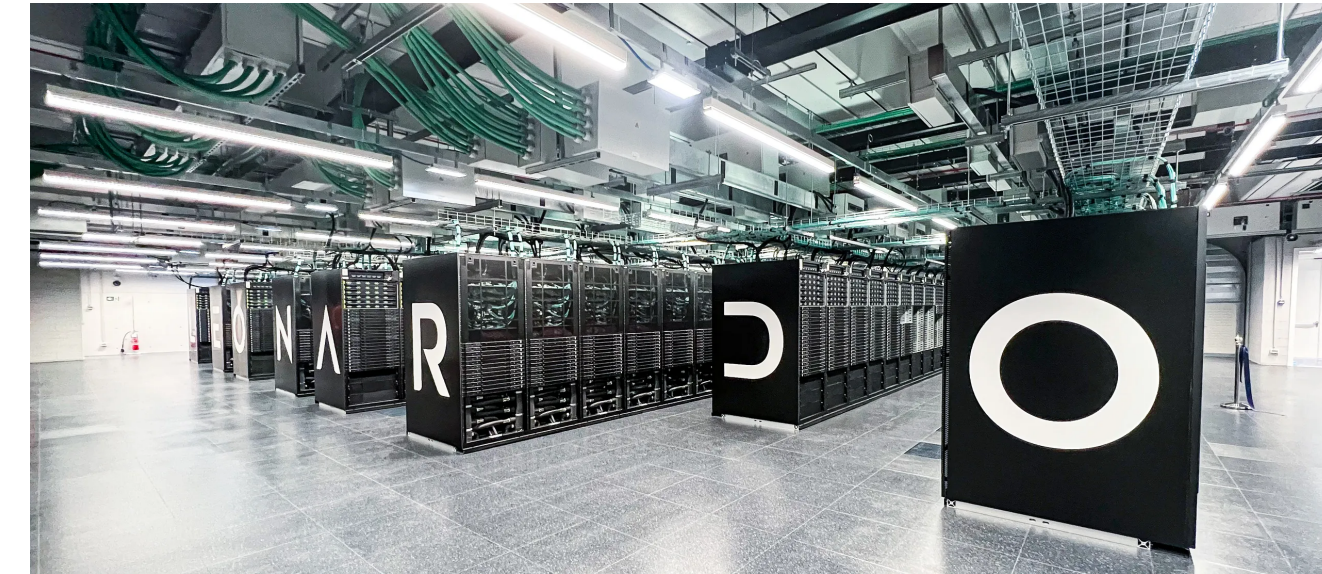
---

*The presented data must be representative of the entire workflow of the project proposed and refer to the main application code you intend for the production work. The software scalability data (see Examples of **Performance Reporting** below) must be used to choose the most efficient job size(s) for the simulations planned in production: the corresponding software performance must be clearly linked to the justification of the computing resources requested. The Domain Panels will not accept estimates based on related codes and/or data related to parts of your production. All data must refer to the targeted systems in your production runs or a system with comparable size, software stack and with the same architecture, and network (e.g. a project can be accepted on MeluXina GPU if it was benchmarked on another GPU machine with the same NVIDIA GPU). Please*

Quantify performance on the specific use cases to be studied

## Materials Science

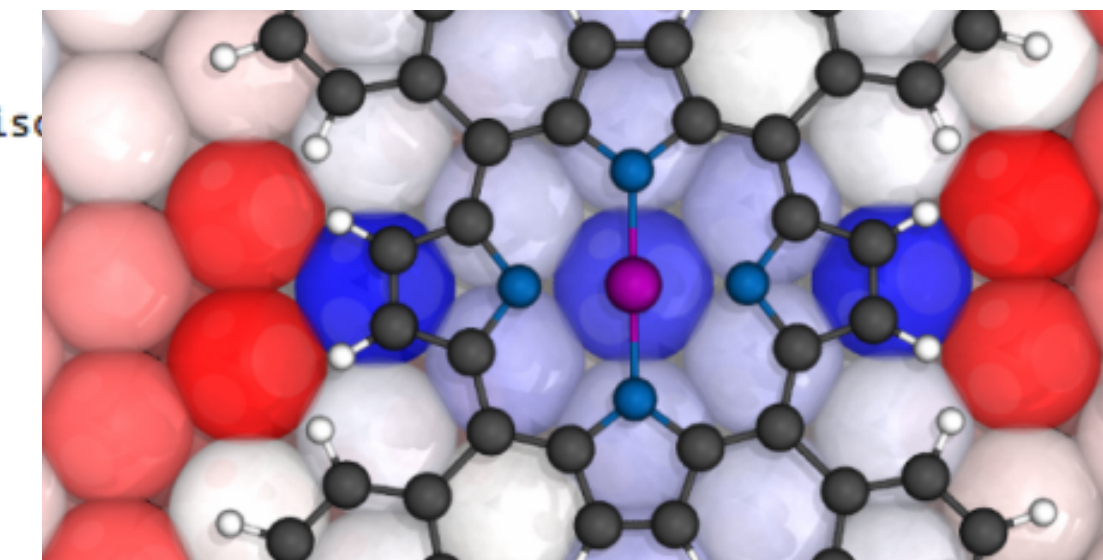
- Exascale
- HPC usage in MatSci
- The MaX CoE



## Benchmarks

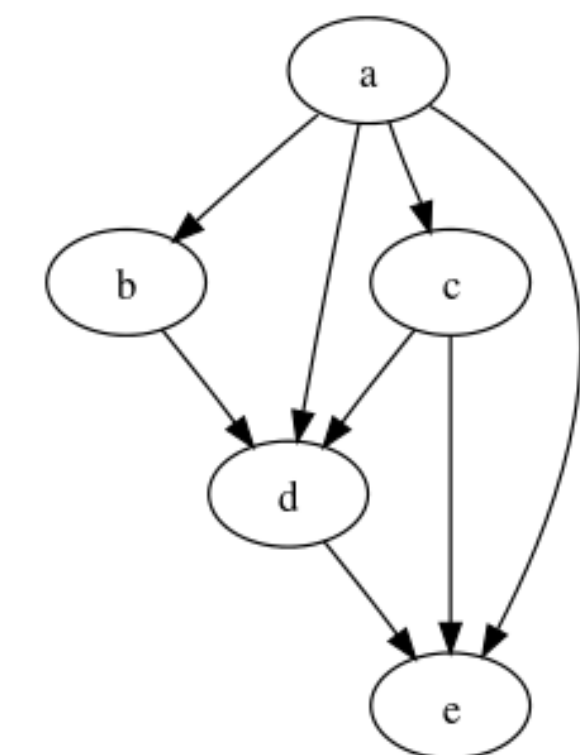
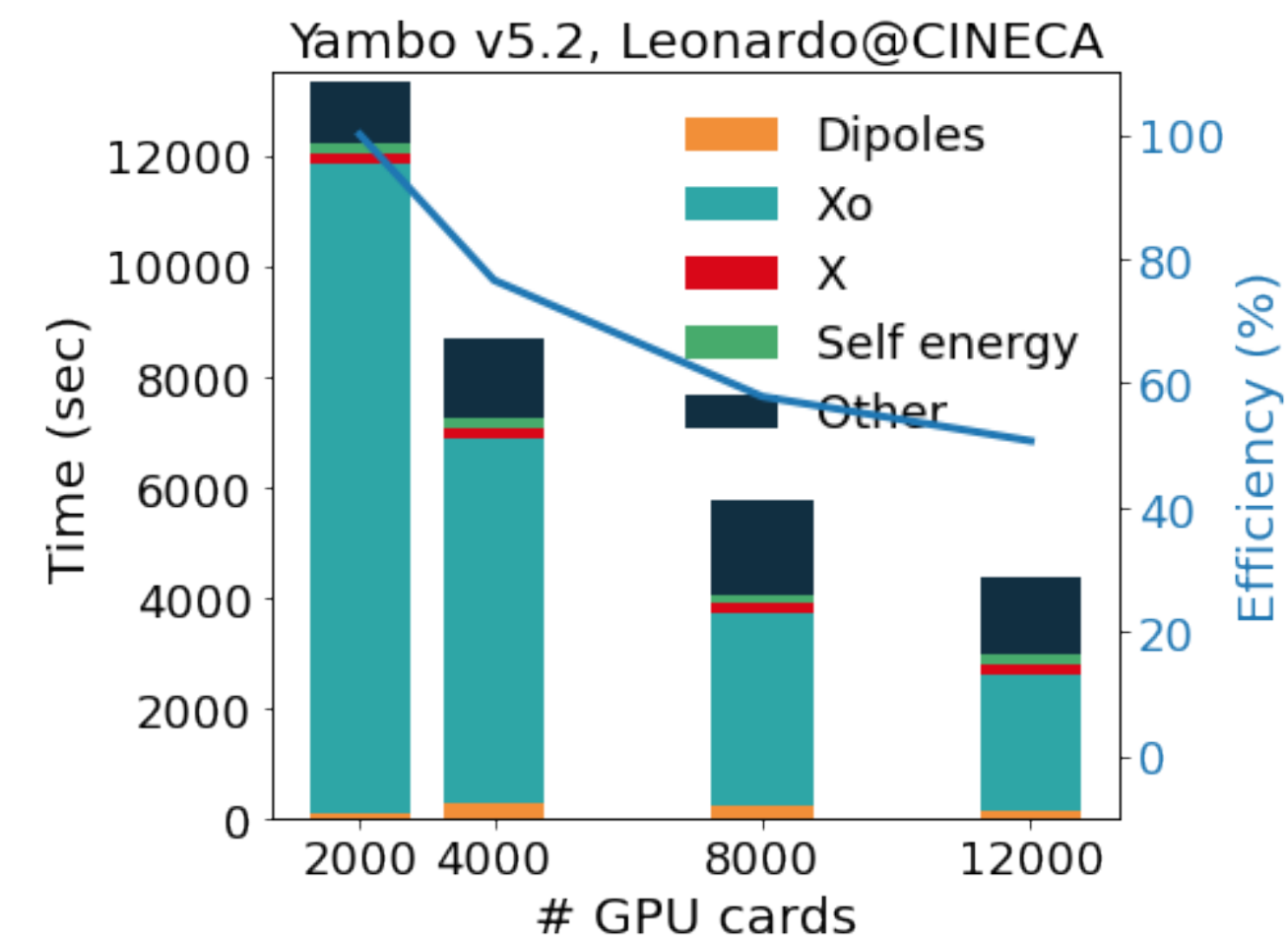
- Scope and structure
- Resources
- Multiple metrics

```
74 !$omp parallel default(shared), private(ir)
75 !$omp do
76 do ir = 1, fft_size
77   isc%rho_tw_rs(ir) = cplx(isc
78 enddo
79 !$omp end do
```



## Selected Results

- Tracking Performance
- Performance Portability
- Deployment



quantum mechanics based  
atomistic modelling of materials  
+  
interfacing with **multiscale** approaches

## Electronic Structure Methods

- highly accurate (predictive)
- computationally demanding
- **a case for HPC**

the **exascale** opportunity:



**Higher accuracy**

**High throughput  
screening**

**Improved modelling  
(complexity)**

quantum mechanics based  
atomistic modelling of materials  
+  
interfacing with **multiscale** approaches

## Electronic Structure Methods

- highly accurate (predictive)
- computationally demanding
- **a case for HPC**

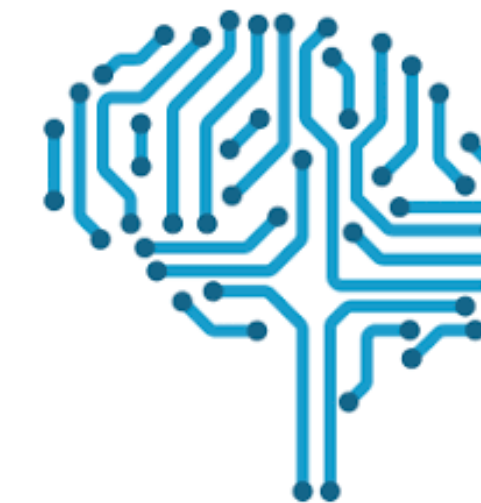
the **exascale** opportunity:



**High throughput  
screening**



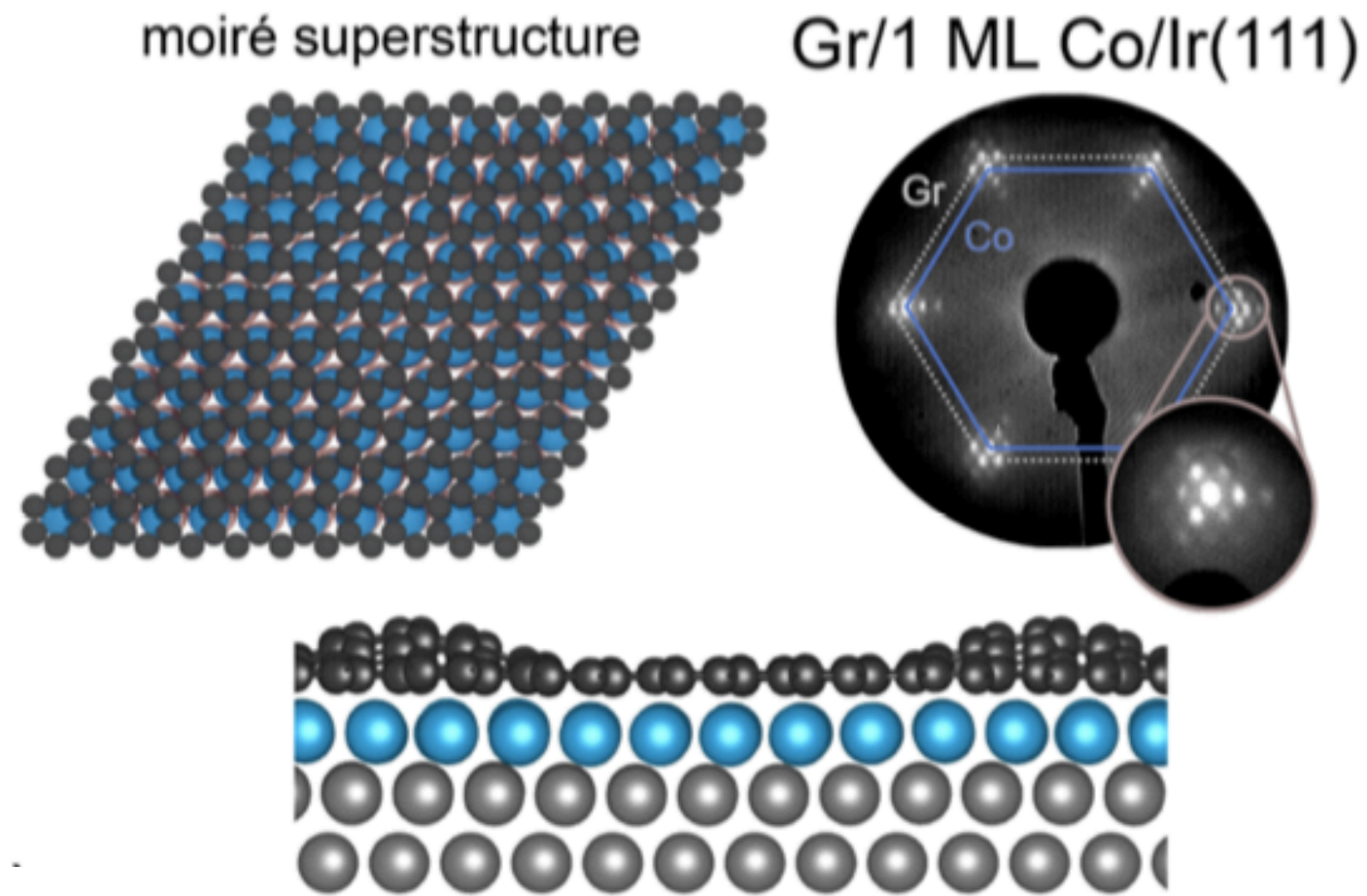
**Higher accuracy**



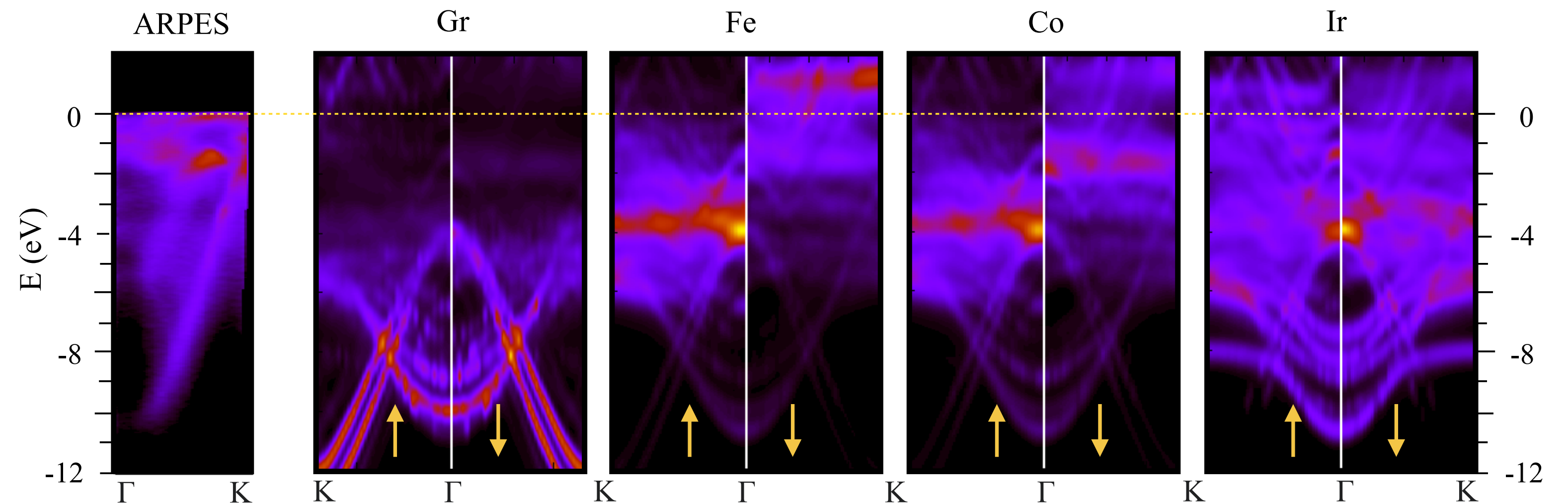
Machine Learning &  
Artificial Intelligence

**Improved modelling  
(complexity)**

# exascale opportunity: complexity

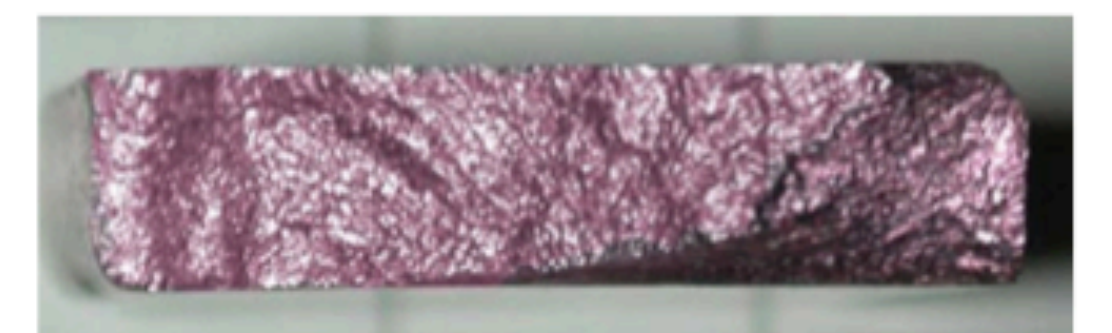
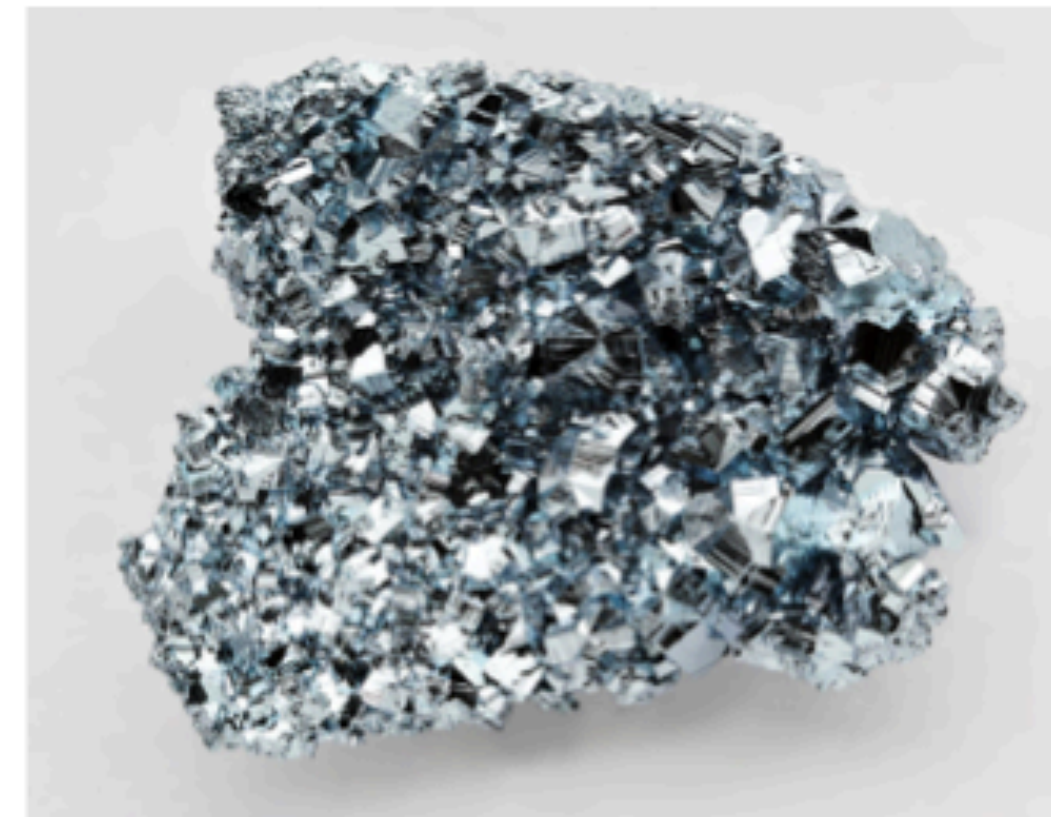
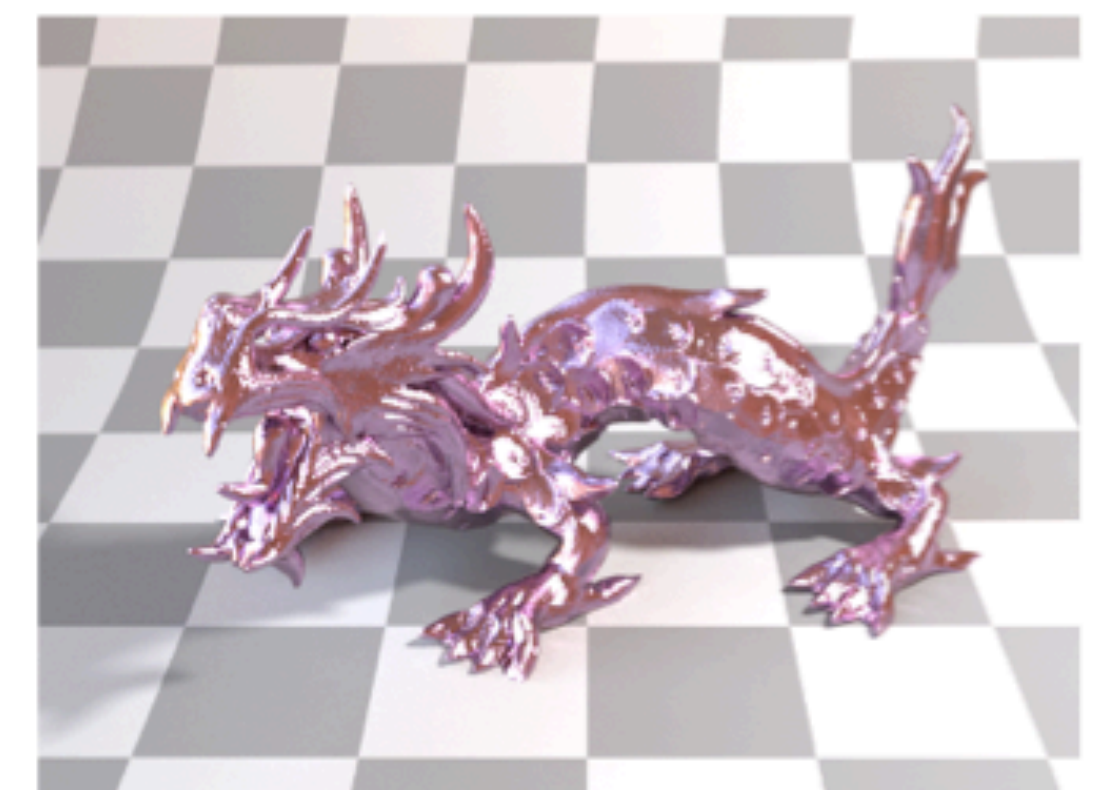
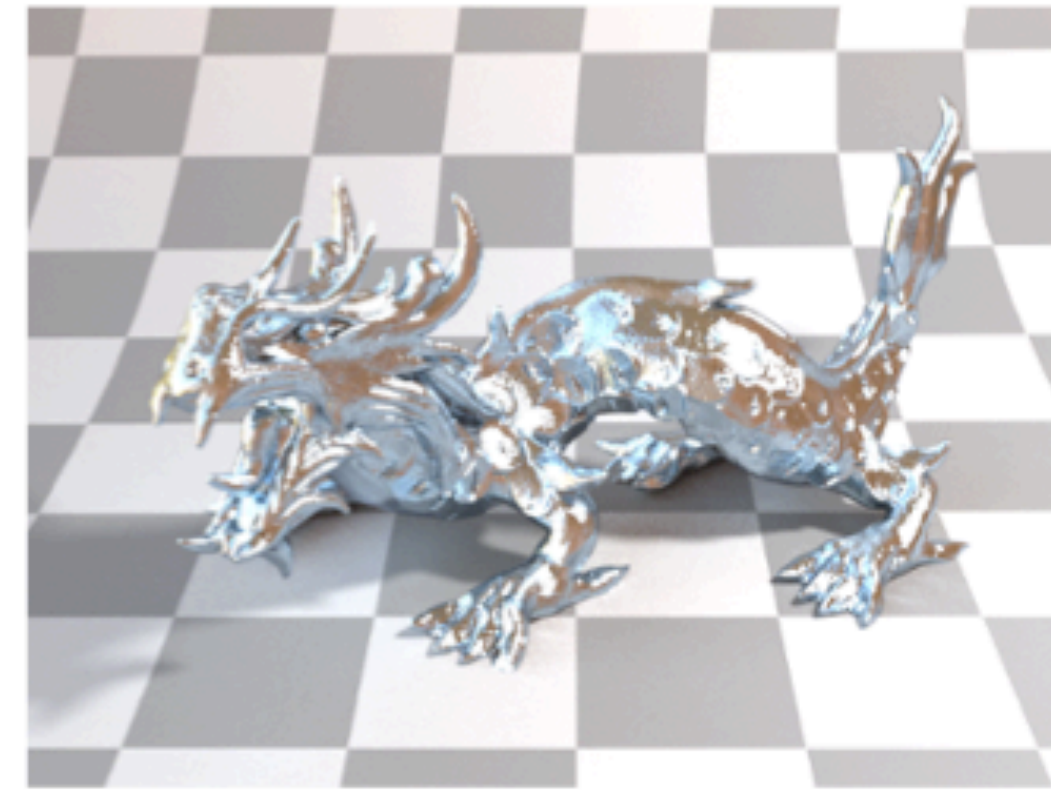
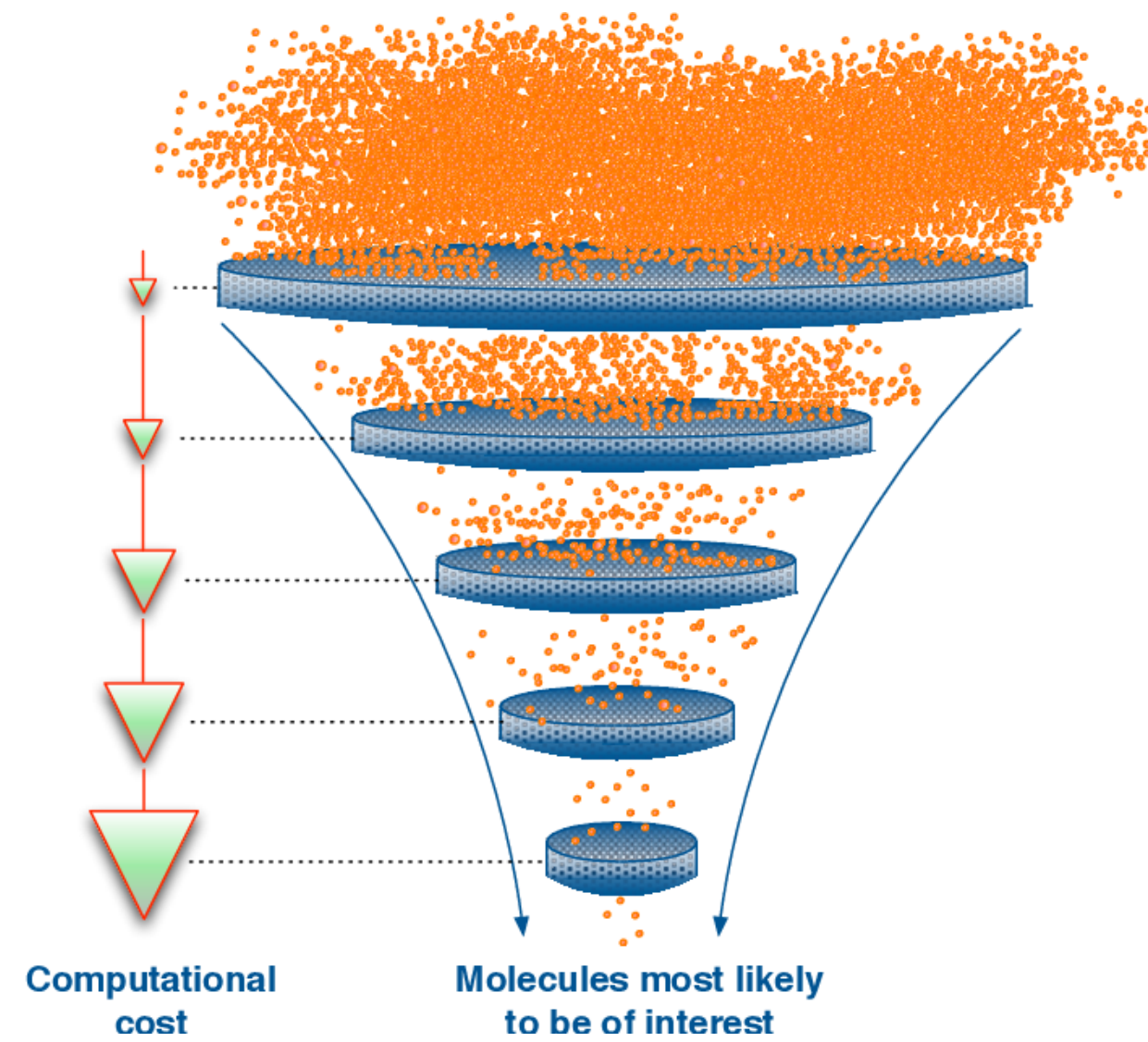


- Graphene / Transition Metal / Ir (111)
- **clear experimental evidence** for moiré pattern (**lattice mismatch**) and **Gr corrugation**
- 10x10 Graphene, 9x9 Iridium => 605 atoms / unit cell
- **Precise treatment of the structure** is important for modelling



- Avvisati et al, J Phys. Chem. C **121**, 1639 (2017)
- Avvisati et al, Nano Lett. **18**, 2268 (2018)
- Calloni et al, J. Chem. Phys. **153**, 214703 (2020)
- Cardoso et al, Phys. Rev. Mat. **5**, 014405 (2021)
- Pacile' et al, Appl. Phys. Lett. **118**, 121602 (2021)

# exascale opportunity: high throughput screening



G. Prandini, G.M. Rignanesse,  
N. Marzari, npj Comput.  
Materials **5**, 129 (2019)



## Electronic-structure methods for materials design

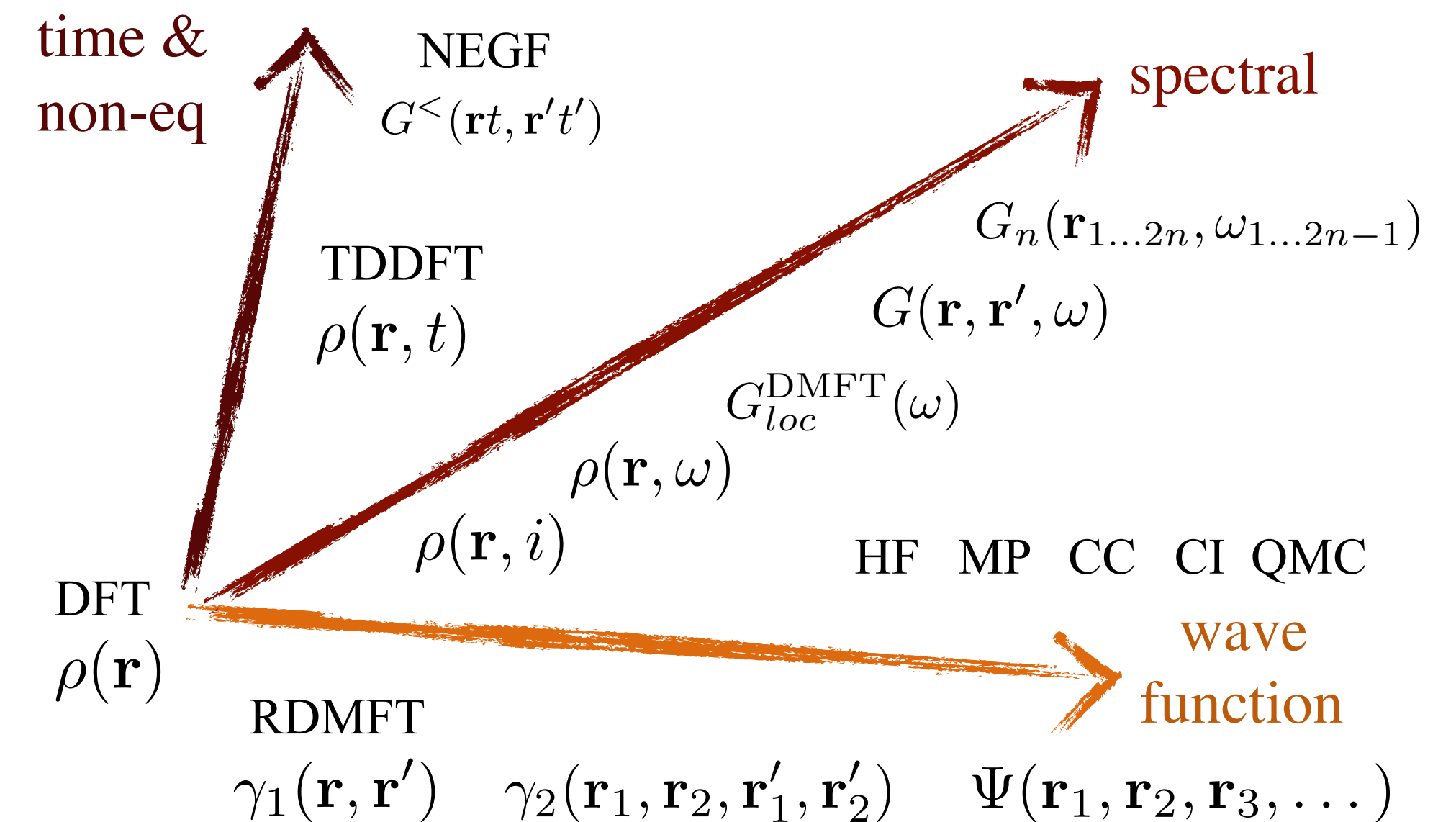
Nicola Marzari <sup>1</sup>✉, Andrea Ferretti <sup>2</sup> and Chris Wolverton <sup>3</sup>

### Density functional theory (DFT):

- applications ranging from **materials modelling, to quantum chemistry and drug design**
- compatible with **high performance computing and high-throughput screening**

### beyond DFT:

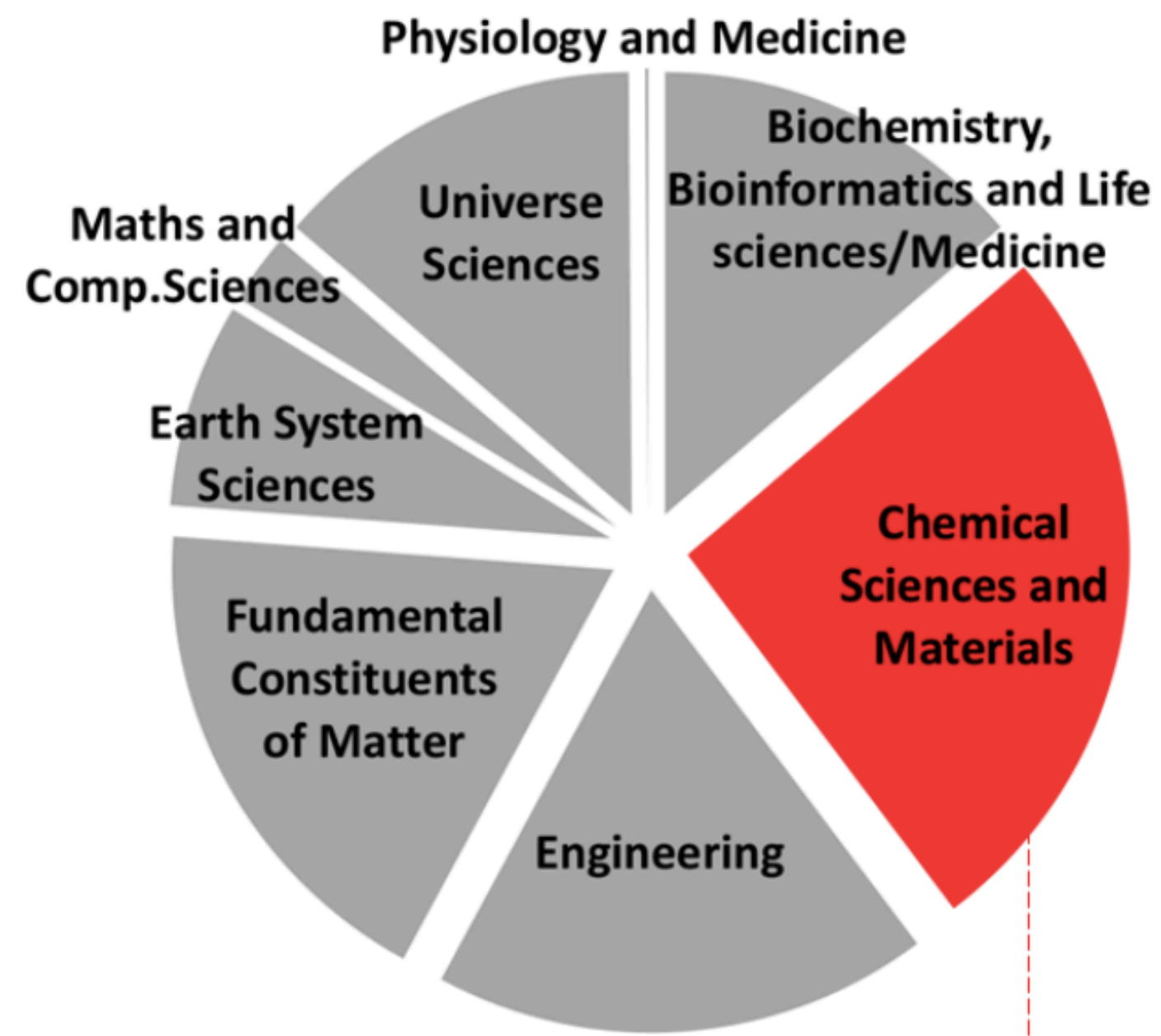
- **multiple hierarchies** can be climbed
  - ▶ wavefunction-based methods
  - ▶ **many-body perturbation theory and spectral methods**
  - ▶ time-dependent and non-equilibrium methods
  - ▶ **ensembles**





<http://www.max-centre.eu/>

- widely used **open source, community codes** in electronic structure



## SELECTED ACTIVITIES

- parallel optimization and performance portability** are key to keep exploiting HPC resources
- All MaX flagship codes released for **production with GPU support**

- hardware-software **codesign vehicles**
- energy-efficiency** of codes

- large effort on **education and training**: hands-on schools and hackathons

# Codes

**GOAL:** turn MaX flagship codes into exascale-enabled applications

- **large scale MPI** parallelism (order of 10,000 tasks)
- combined with **GPU awareness**

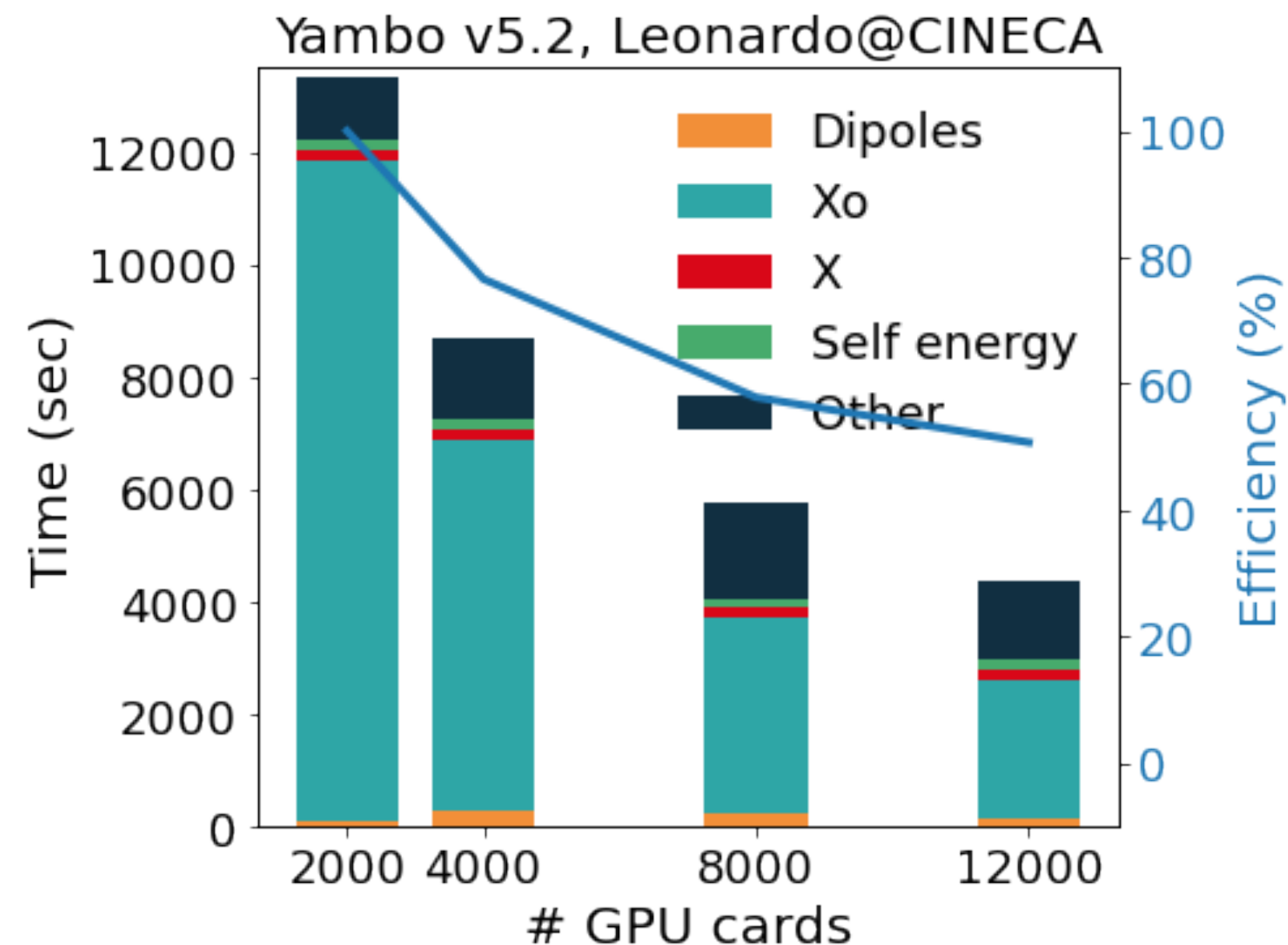


**EuroHPC**  
Joint Undertaking

**Jupiter (@FZJ-JSC):** > 1 ExaFlops  
~ 6000 nodes with 4 Nvidia GH200/node  
~ 24,000 GPUs in total

**Leonardo @ CINECA:**  
4 Nvidia A100 next / node

**runs:** up to 3000 nodes about  
87% of the whole machine (3456  
nodes)



**... more about benchmarking**

# Benchmarking and profiling

<https://gitlab.com/max-centre/JUBE4MaX>

<https://gitlab.com/max-centre/benchmarks-max3/>

max-centre / JUBE4MaX / Repository

yambo-jube ▾ JUBE4MaX



**updated leonardo qos: boost\_qos\_bprod**  
Nicola Spallanzani authored 1 week ago

benchmark-yambo-single-node.xml

benchmark-yambo.xml

profile-yambo-grco.xml

workflow-yambo.xml



- benchmarks are collected in a **dedicated repo**
- **work coordinated by CINECA**
- Using the **JUBE benchmarking environment**, to
  - create benchmark sets
  - run benchmarks on different computer systems
  - evaluate and display the results

- **JUBE** adopted as a **standard** for benchmarks production
- Objectives: **standardization**, **reproducibility**, **accessibility**
- **Benchmark repository** available incl **data inspection** interface

# Benchmark repository

<https://gitlab.com/max-centre/JUBE4MaX>



## PLATFORM

platform.xml

CINECA/**Leonardo**(Booster, DCGP)  
CINECA/**G100**  
**LUMI** (C/G)  
**MareNostrum/4**  
**MareNostrum/5** (ACC,GPP)  
IT4I/**Barbora**  
IT4I/**Karolina**(CPU/GPU)  
IT4I/**Power10**  
**Meluxina**(CPU/GPU)  
**Deucalion**(CPU/GPU)  
**Discoverer**  
Sling/**Vega** (CPU/GPU)  
Sling/**Arnes**(CPU/GPU)

## APPLICATIONS

applications.xml

QuantumESPRESSO  
Yambo  
Siesta  
Fleur

## WORKLOAD

workload.xml

QE (PW/PH)  
Yambo  
Siesta  
Fleur


## SCRIPT

script.xml

HDEEM  
MERIC  
Nsys  
Nvidia-smi  
Scorep  
Tau

# Benchmark portal

MaX3 benchmarks    Quantum ESPRESSO    Yambo    SIESTA    BigDFT    FLEUR



**About Max**

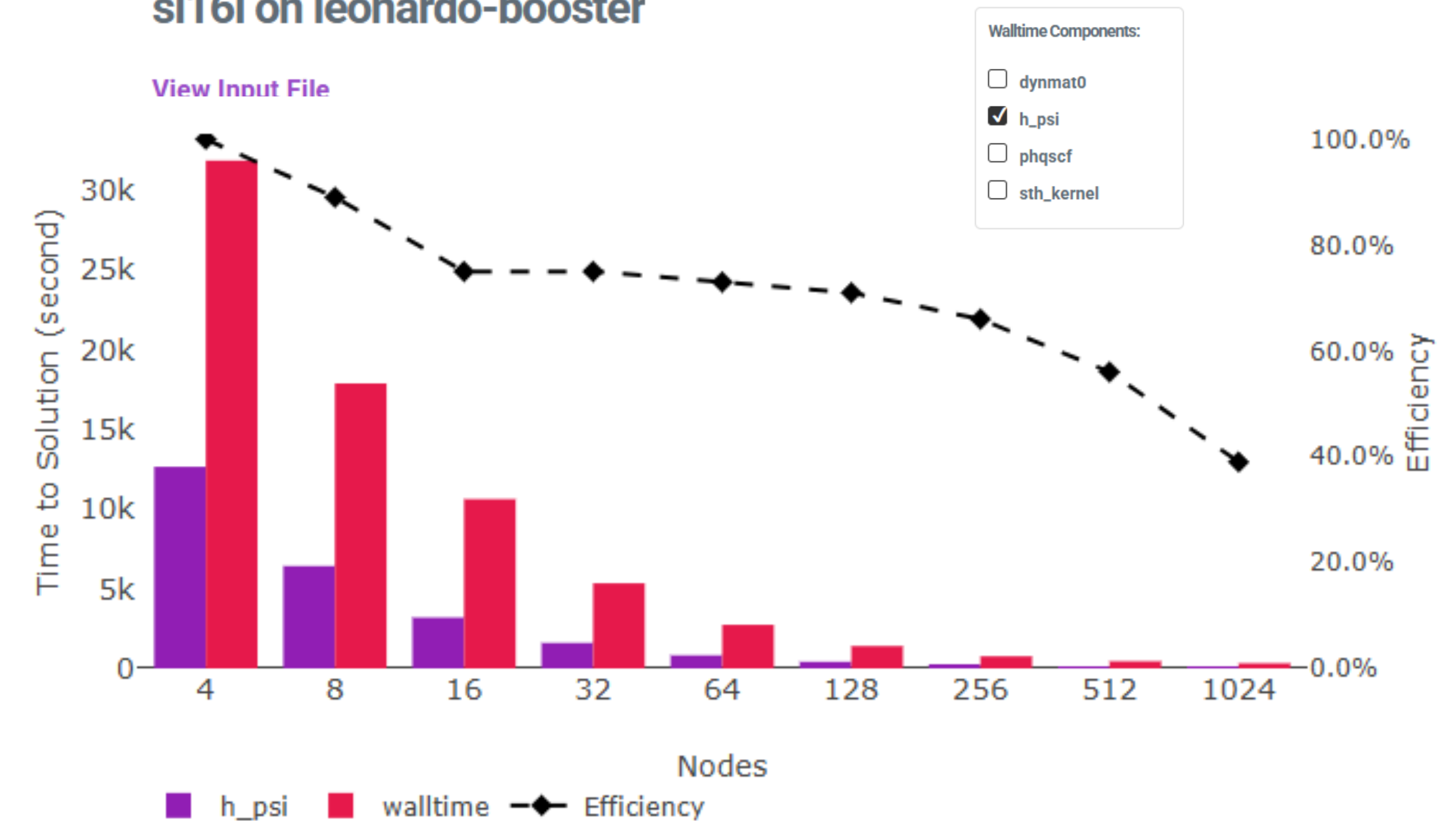
MAX (MAterials design at the eXascale) is a European Centre of Excellence which enables materials modelling, simulations, discovery and design at the frontiers of the current and future High Performance Computing (HPC), High Throughput Computing (HTC) and data analytics technologies.

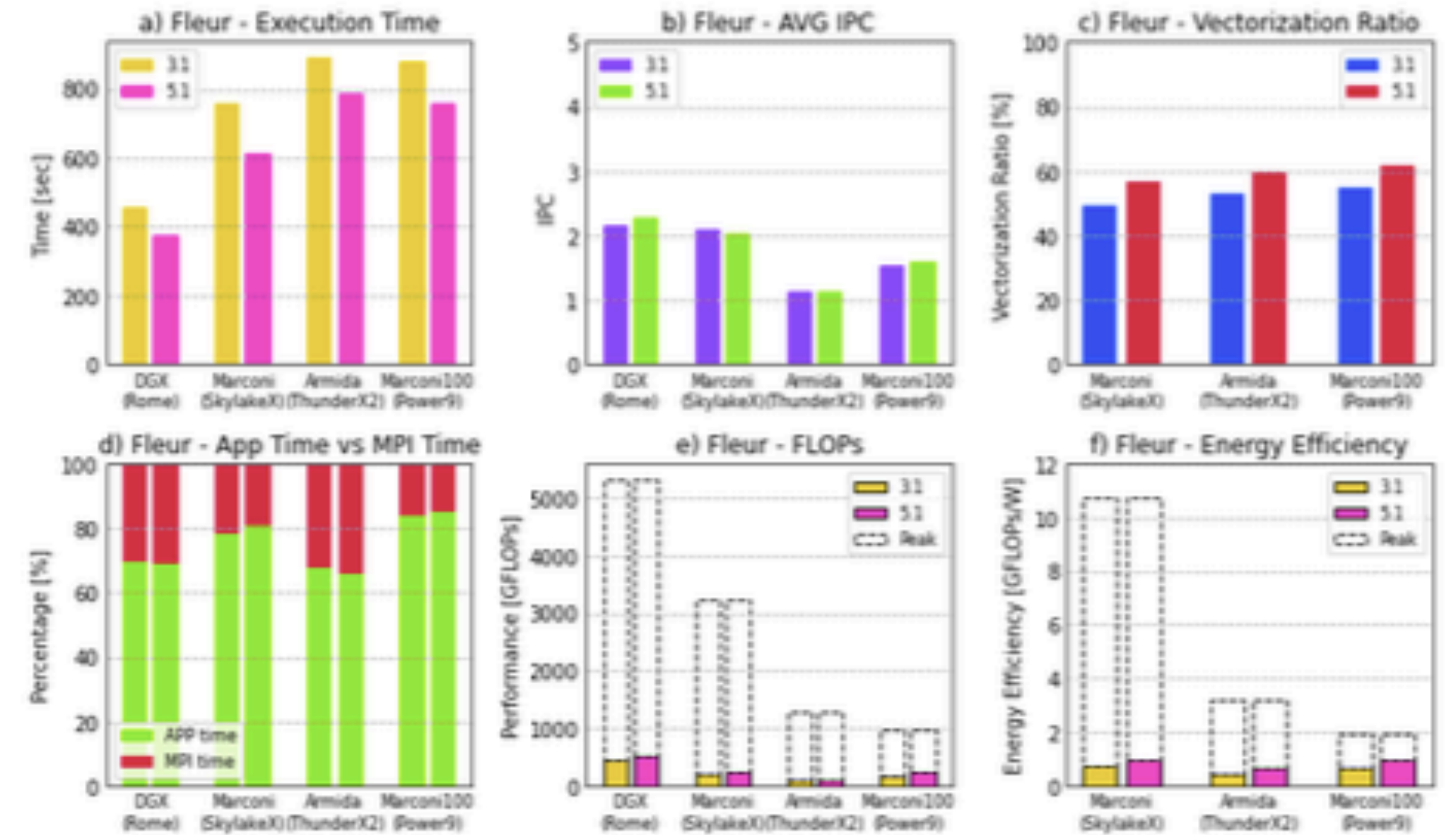
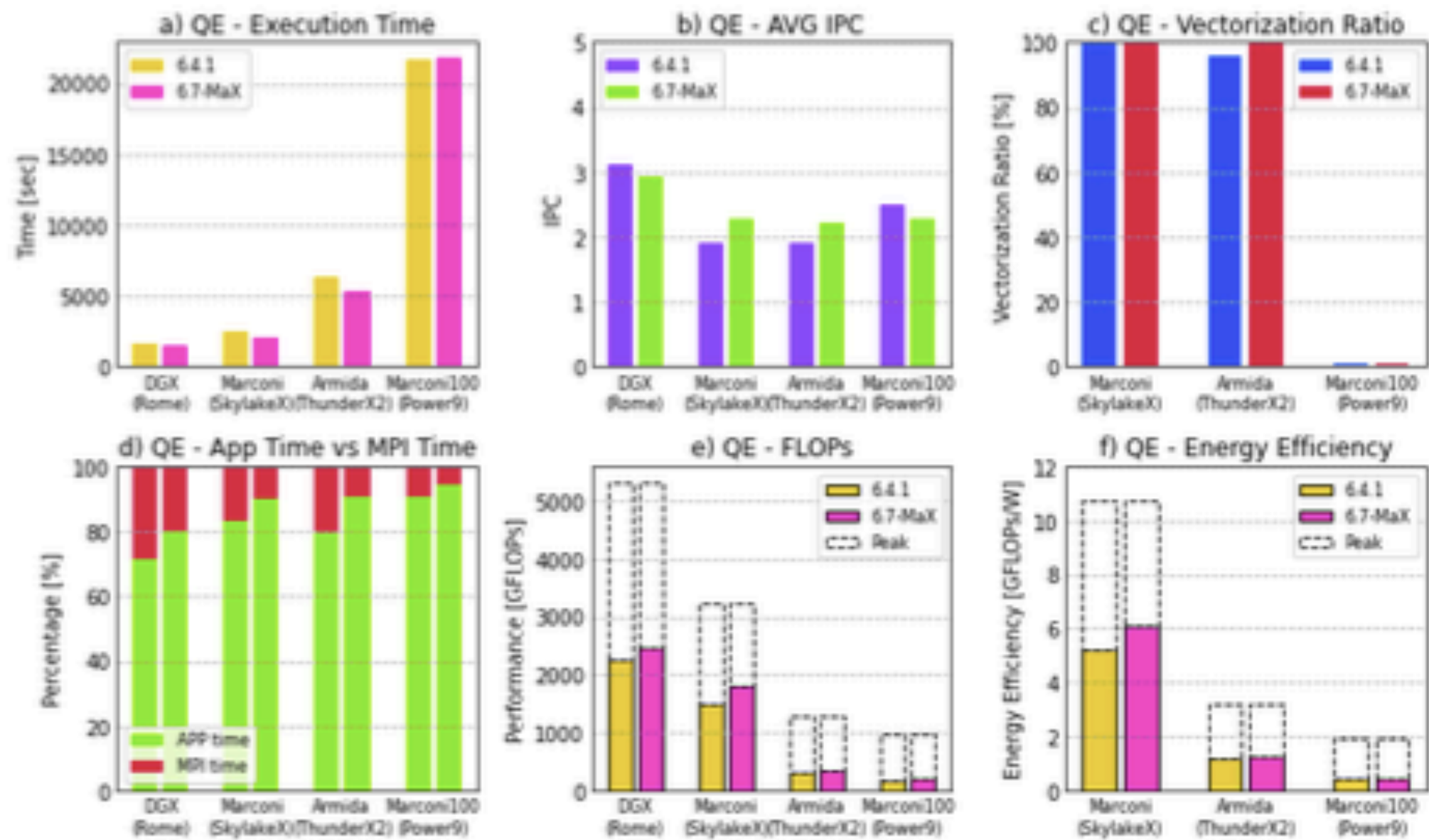
Workload	Info	Platform	Version	Date	Link
waters	-	leonardo-booster	v7.3.1	2024-02	<a href="#">waters_0f4059</a>
waters	weakscaling	leonardo-booster	v7.3.1	2024-02	<a href="#">waters_1ff052</a>
grir443	-	leonardo-booster	v7.3.1	2024-10	<a href="#">grir443_e8cdd8</a>
si16l	-	leonardo-booster	v7.2	2023-10	<a href="#">si16l_37aff3</a>
ausurf112_2scf	-	mn5acc	7.3.1	2024-07-fullnode	<a href="#">ausurf112_2scf_99f13b</a>
ausurf112_fullscf	-	davinci1	dev7.4	2024-11	<a href="#">ausurf112_fullscf_6f6f7a</a>
si16l-mps	-	leonardo-booster	7.4.1	2025-03	<a href="#">si16l-mps_02e268</a>
quartz9-mps	-	leonardo-booster	7.3.1	2025-03	<a href="#">quartz9-mps_70acc6</a>
cnt10por8_19	-	leonardo-booster	7.3.1	2024-10	<a href="#">cnt10por8_19_3b273f</a>
cnt10por8_19	-	leonardo-booster	7.4.1	2025-03	<a href="#">cnt10por8_19_c4911a</a>



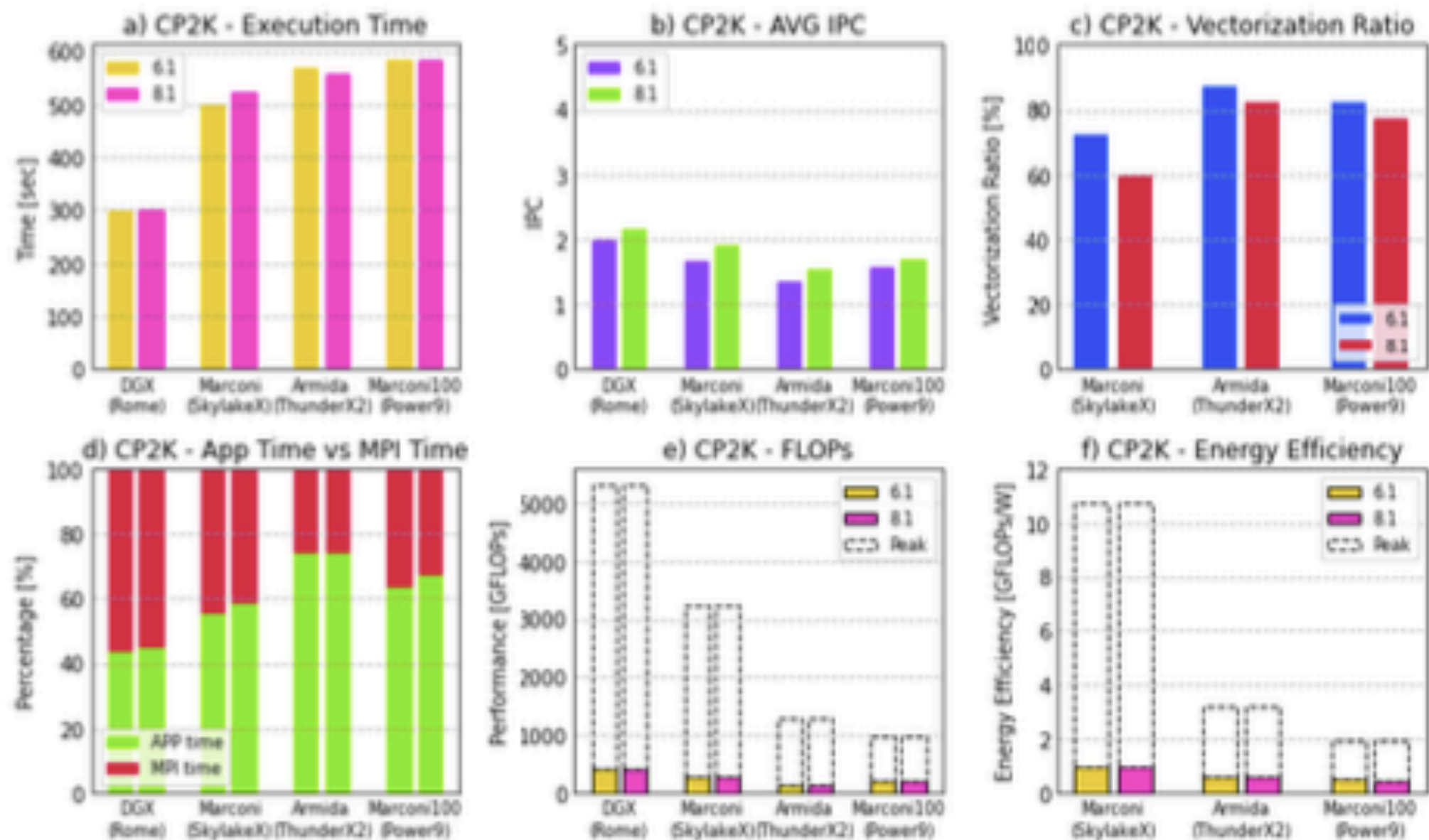
<https://tinyurl.com/max3-benchmarks>

## si16l on leonardo-booster





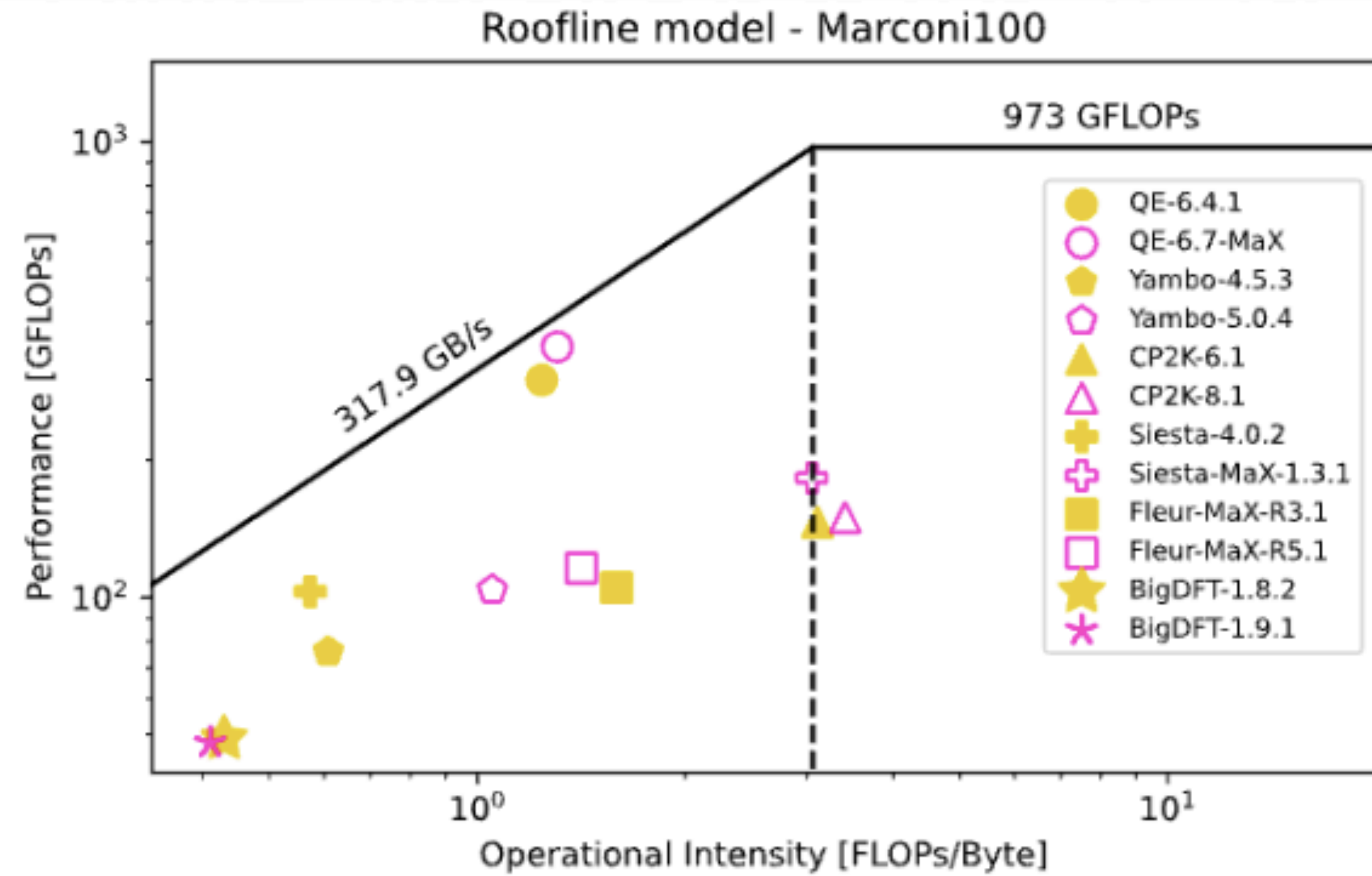
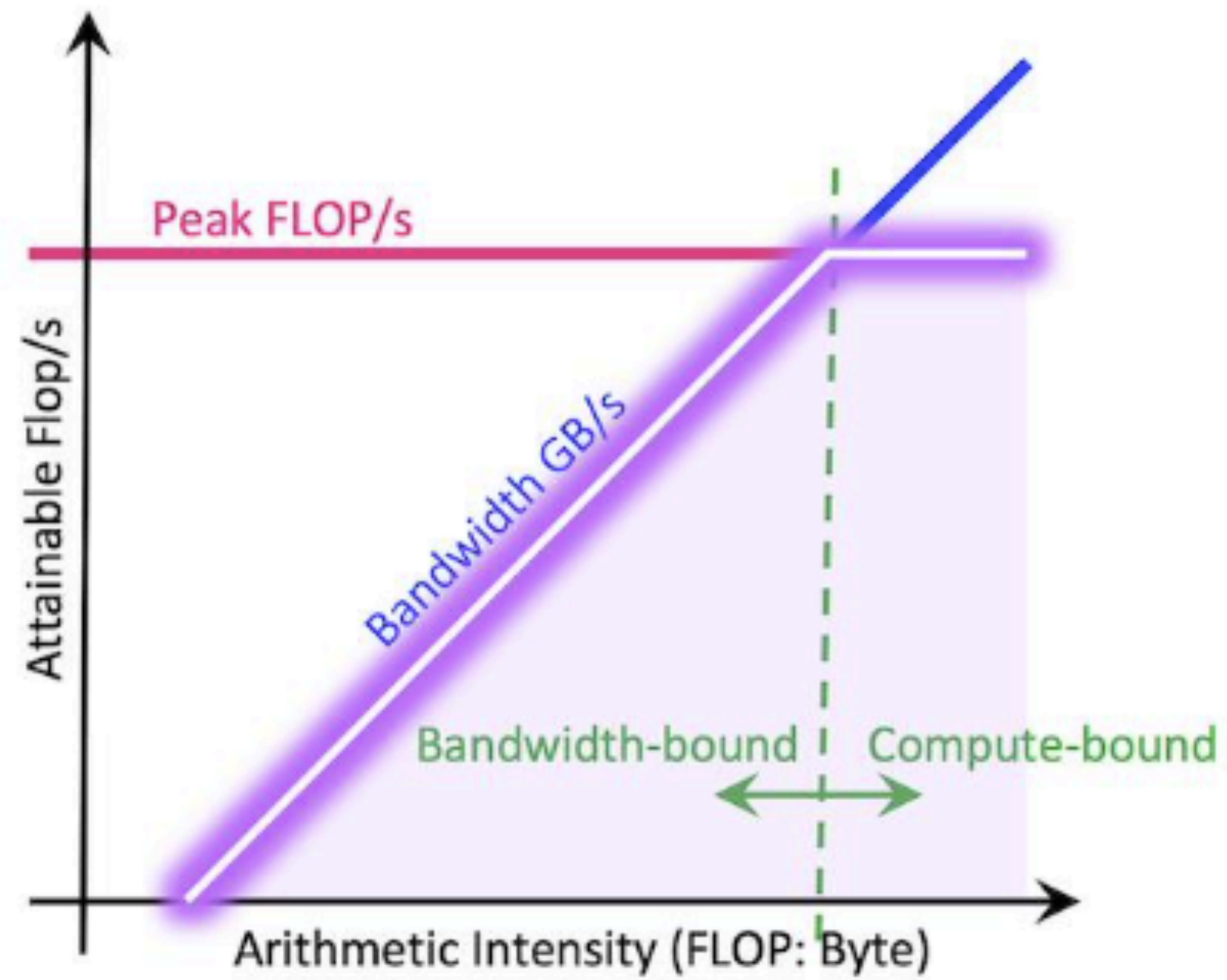
# Multiple Metrics



For each code:

- execution time
- instruction per cycle
- vectorization ratio
- app time vs MPI time
- FLOP/s
- energy efficiency

# Roofline model

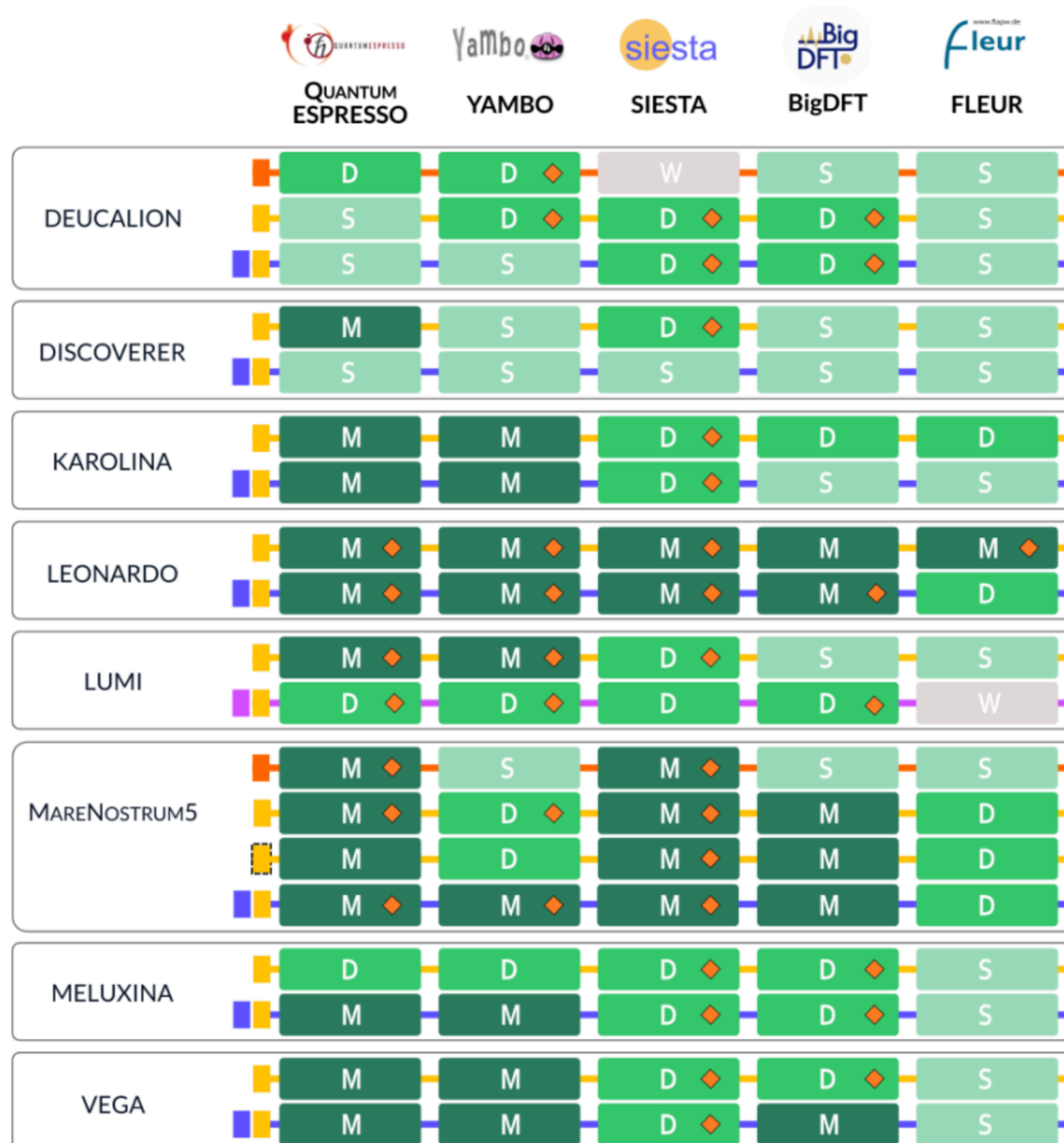


# EasyBuild and Spack support



MaX Codes	Spack support	Spack Repository type	EasyBuild support	EasyBuild Repository type
<b>QuantumEspresso</b>	CPU GPU	Spack dev <a href="https://github.com/spack/spack">https://github.com/spack/spack</a>	CPU GPU	EB Repo <a href="https://github.com/easybuilders/easybuild-easyconfigs">https://github.com/easybuilders/easybuild-easyconfigs</a>
<b>Yambo</b>	CPU GPU	Local repo <a href="https://github.com/nicspalla/my-repo">https://github.com/nicspalla/my-repo</a>	CPU GPU wip	Local Repo <a href="https://github.com/nicspalla/my-repo">https://github.com/nicspalla/my-repo</a>
<b>Siesta</b>	CPU GPU	Local repo <a href="https://gitlab.com/siesta-project/ecosystem/build-tools">https://gitlab.com/siesta-project/ecosystem/build-tools</a>	CPU GPU	EB Repo and Local Repo <a href="https://github.com/AnthoniAlcaraz/easybuild-easyconfigs/tree/develop/easybuild/easyconfigs/s/Siesta">https://github.com/AnthoniAlcaraz/easybuild-easyconfigs/tree/develop/easybuild/easyconfigs/s/Siesta</a>
<b>BigDFT</b>	CPU GPU wip	Spack dev <a href="https://github.com/spack/spack-packages">https://github.com/spack/spack-packages</a>	CPU GPU wip	EB Repo <a href="https://github.com/easybuilders/easybuild-easyconfigs">https://github.com/easybuilders/easybuild-easyconfigs</a>
<b>Fleur</b>	CPU GPU	Local repo <a href="https://iffgit.fz-juelich.de/fleur/fleur/-/tree/develop/packaging/spack?ref_type=heads">https://iffgit.fz-juelich.de/fleur/fleur/-/tree/develop/packaging/spack?ref_type=heads</a>	CPU	Local Repo <a href="https://repo.sling.si/aprah/easyconfigs">https://repo.sling.si/aprah/easyconfigs</a>

# Deployment on EuroHPC machines



July 2025



**M**

Deployed module available

**D**

Demonstrated developers installation

**S**

Supported architectures tested on similar architectures

**W**

Work in progress



Benchmarks configured and executed

CPU ARM

CPU x86

--- + HBM

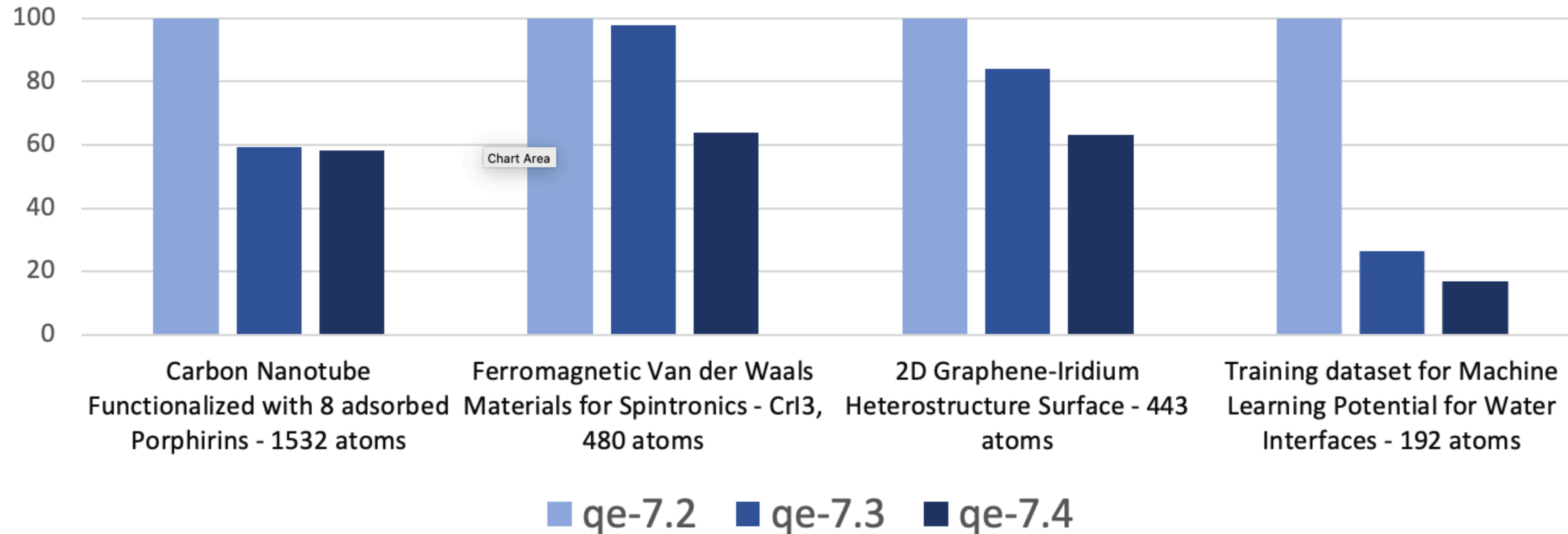
GPU AMD

GPU NVIDIA

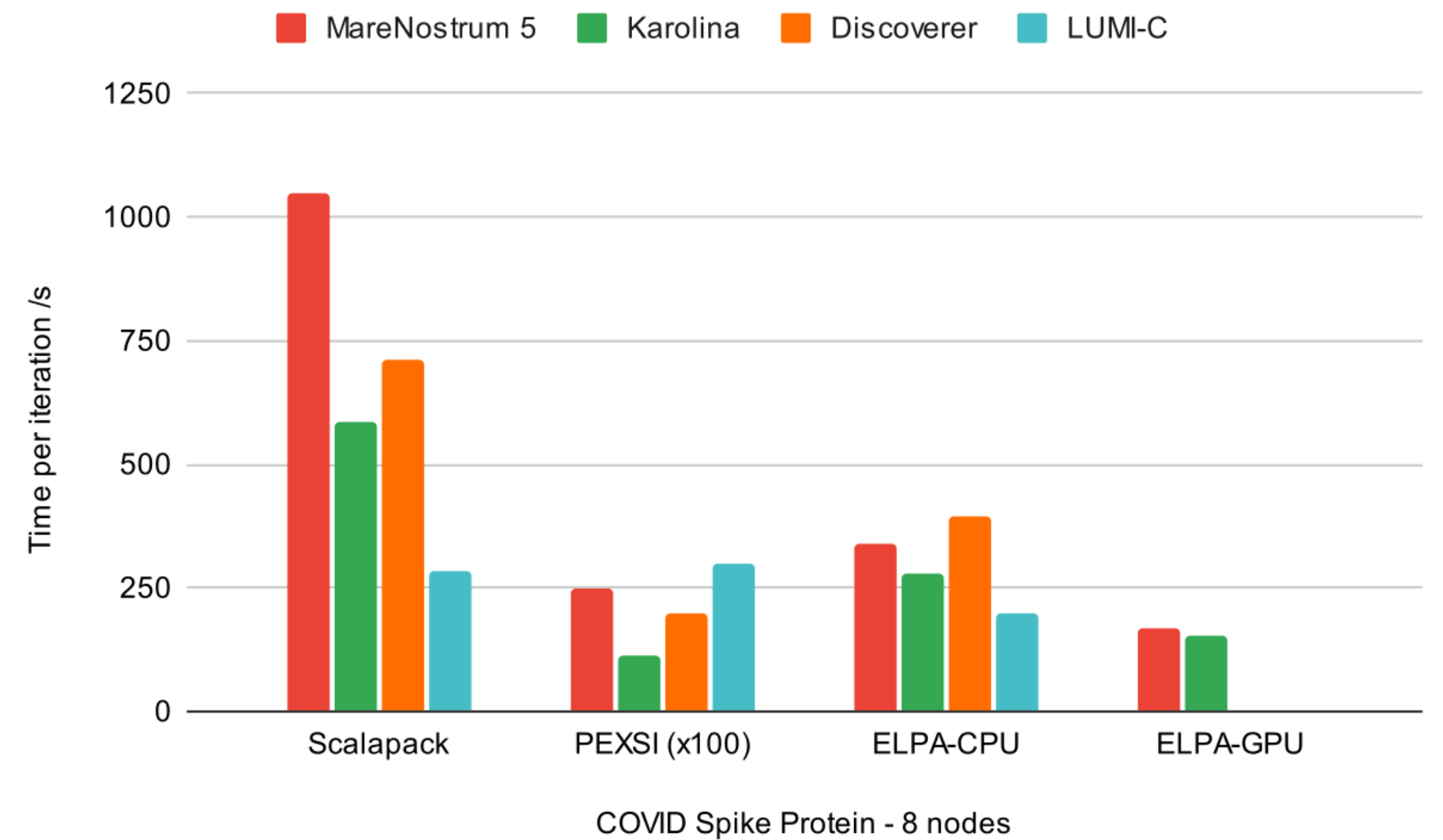
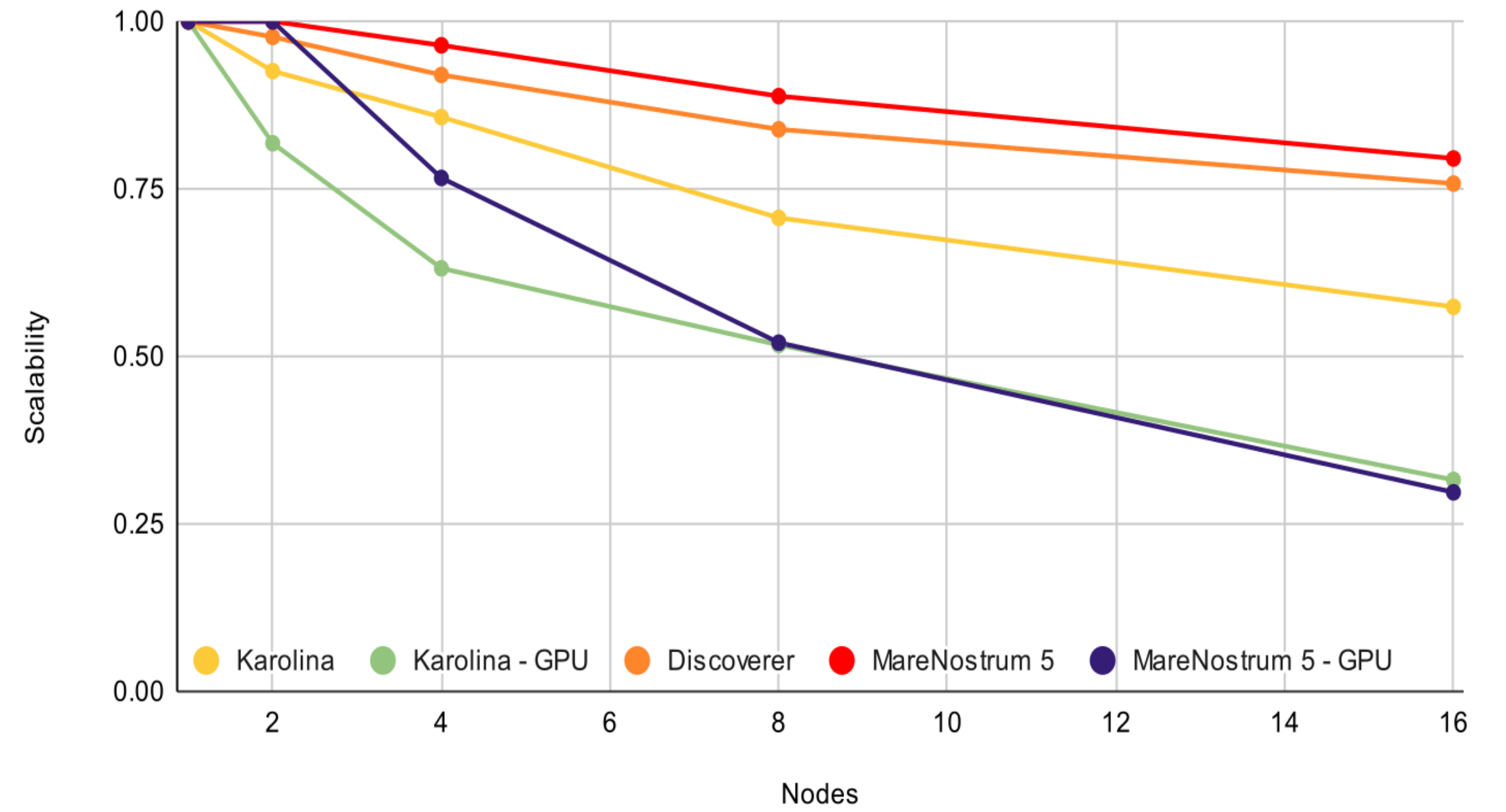
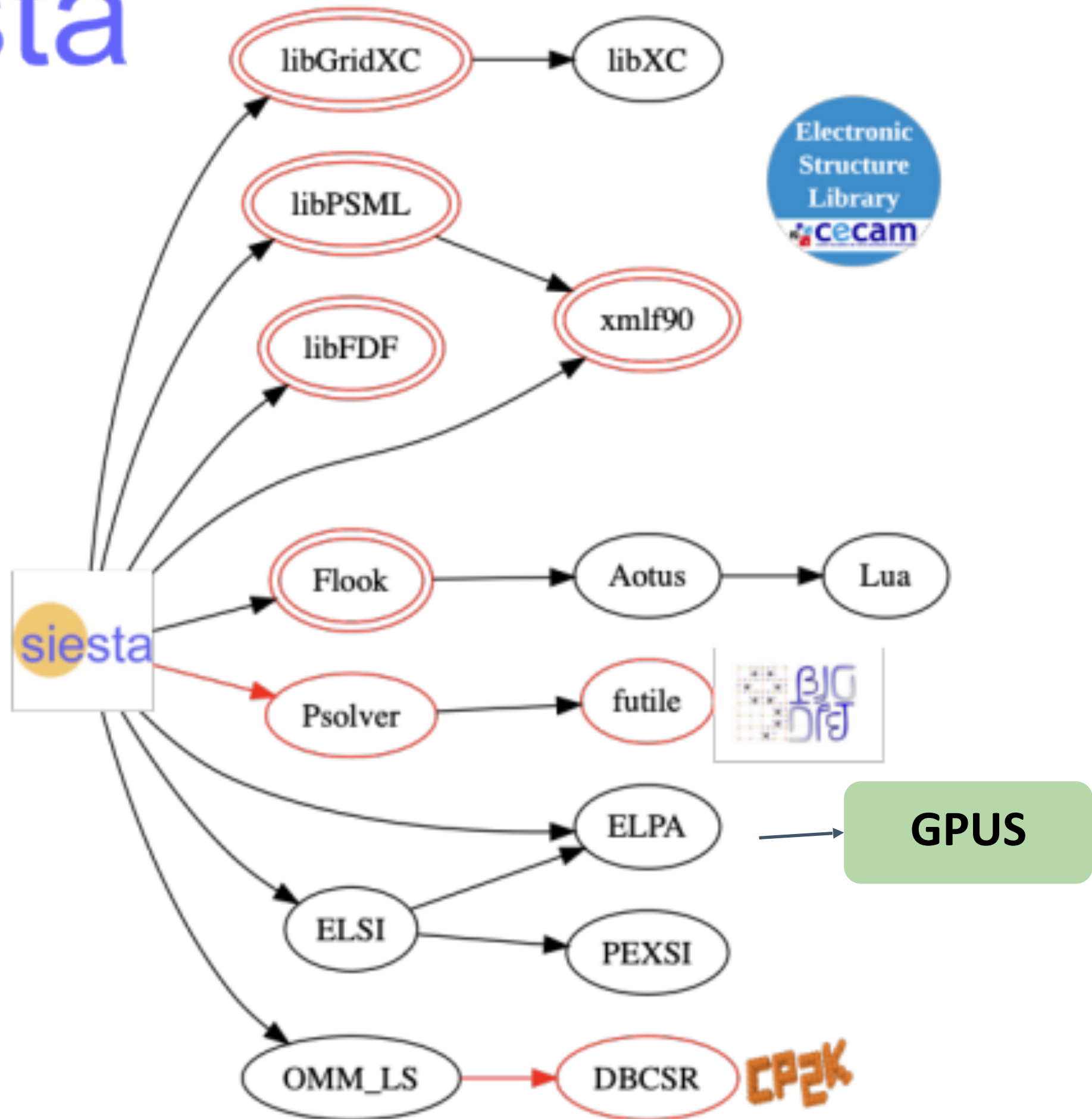


## Selected results

## Performance improvement of Quantum ESPRESSO Releases on Leonardo (GPU) on selected classes of systems

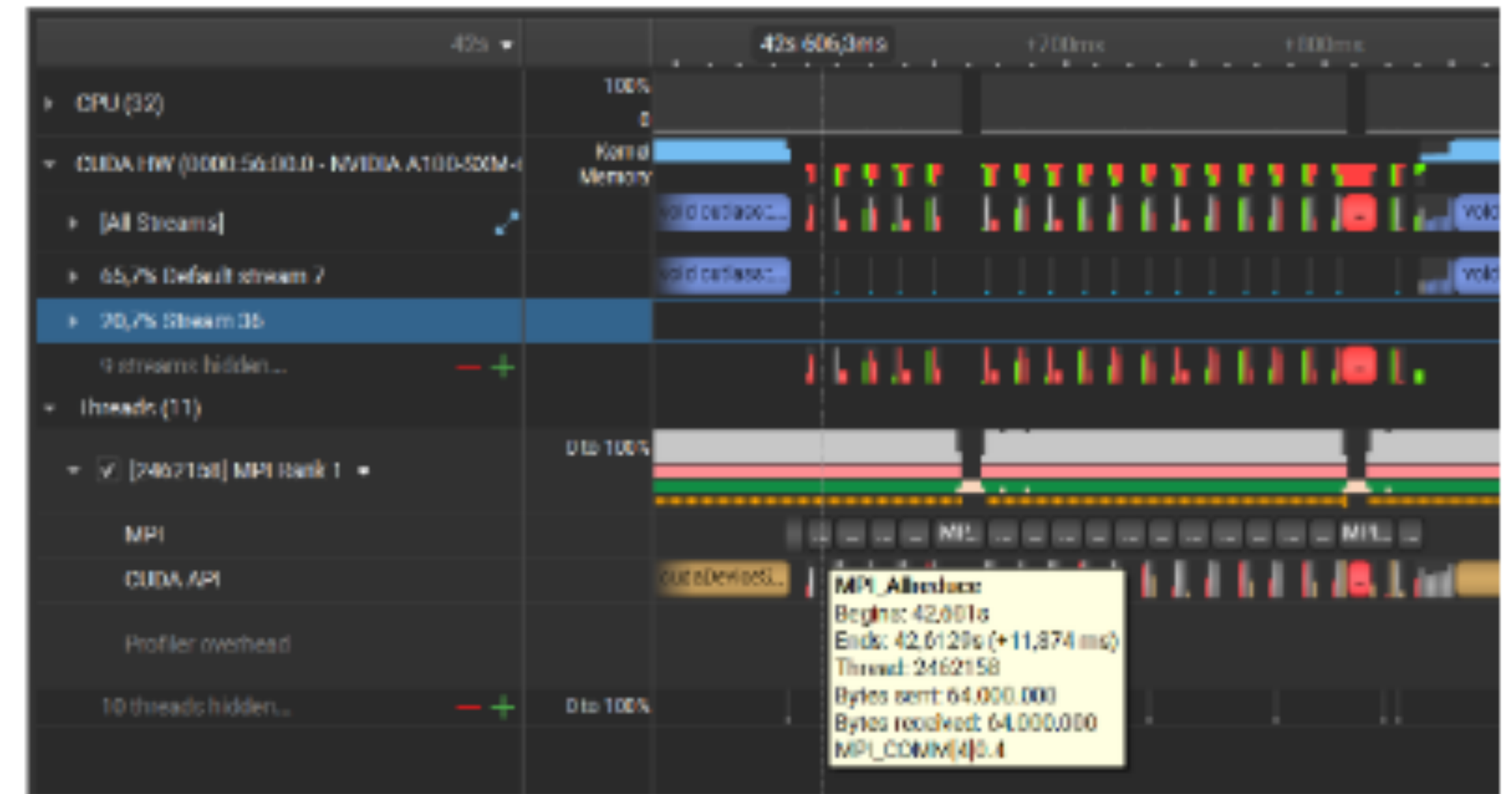
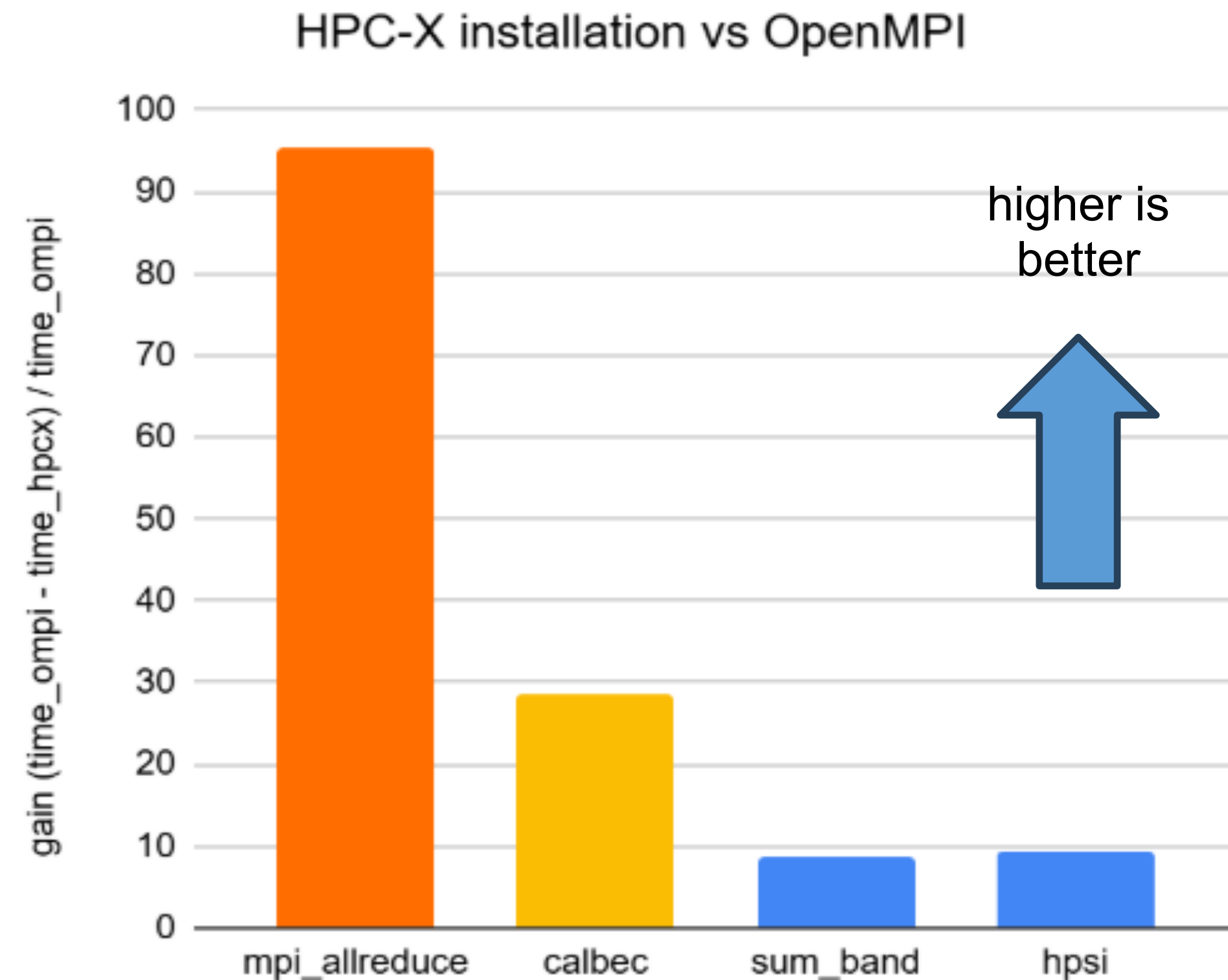


# Performance tracking



# HPCX-MPI

**HPCX-MPI** showed to outperform OpenMPI on NVLink networks, thanks to the support to GPUdirect communications.



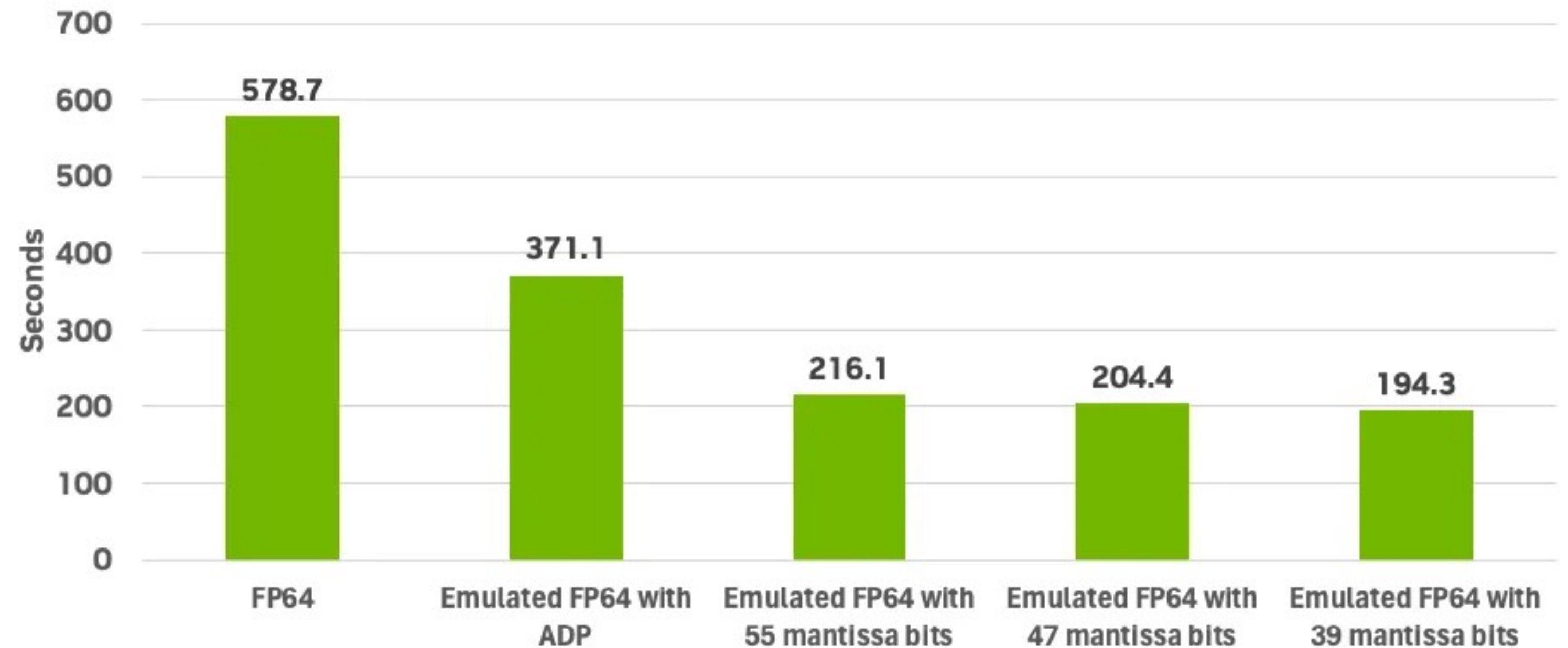
# Precision emulation

## FP64 emulation of GEMMs

From the Nvidia website



Ausurf Performance on RTX PRO 6000 Blackwell SE



<https://developer.nvidia.com/blog/unlocking-tensor-core-performance-with-floating-point-emulation-in-cublas/>

# Yambo-cuSOLVERMp interface

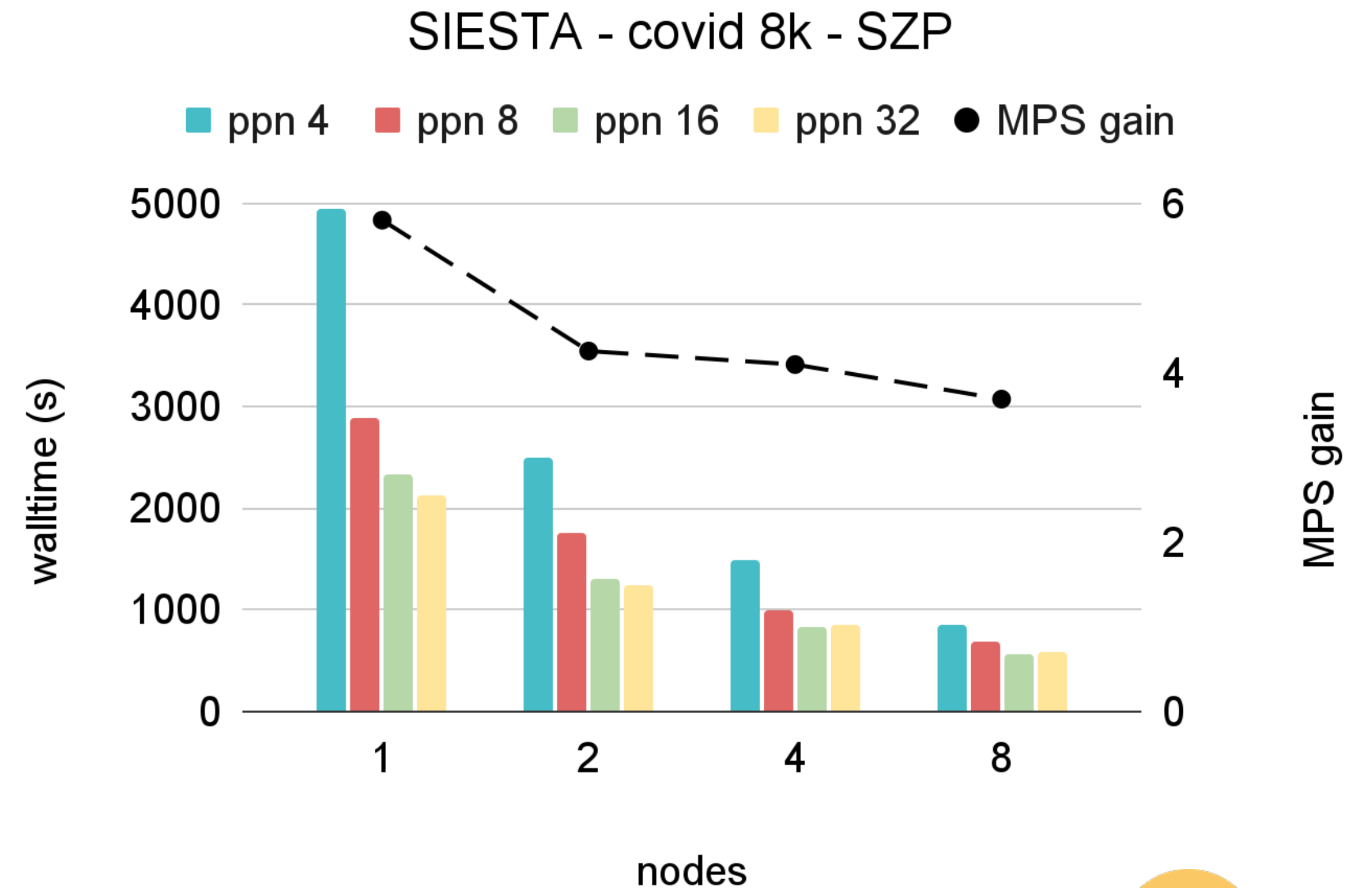
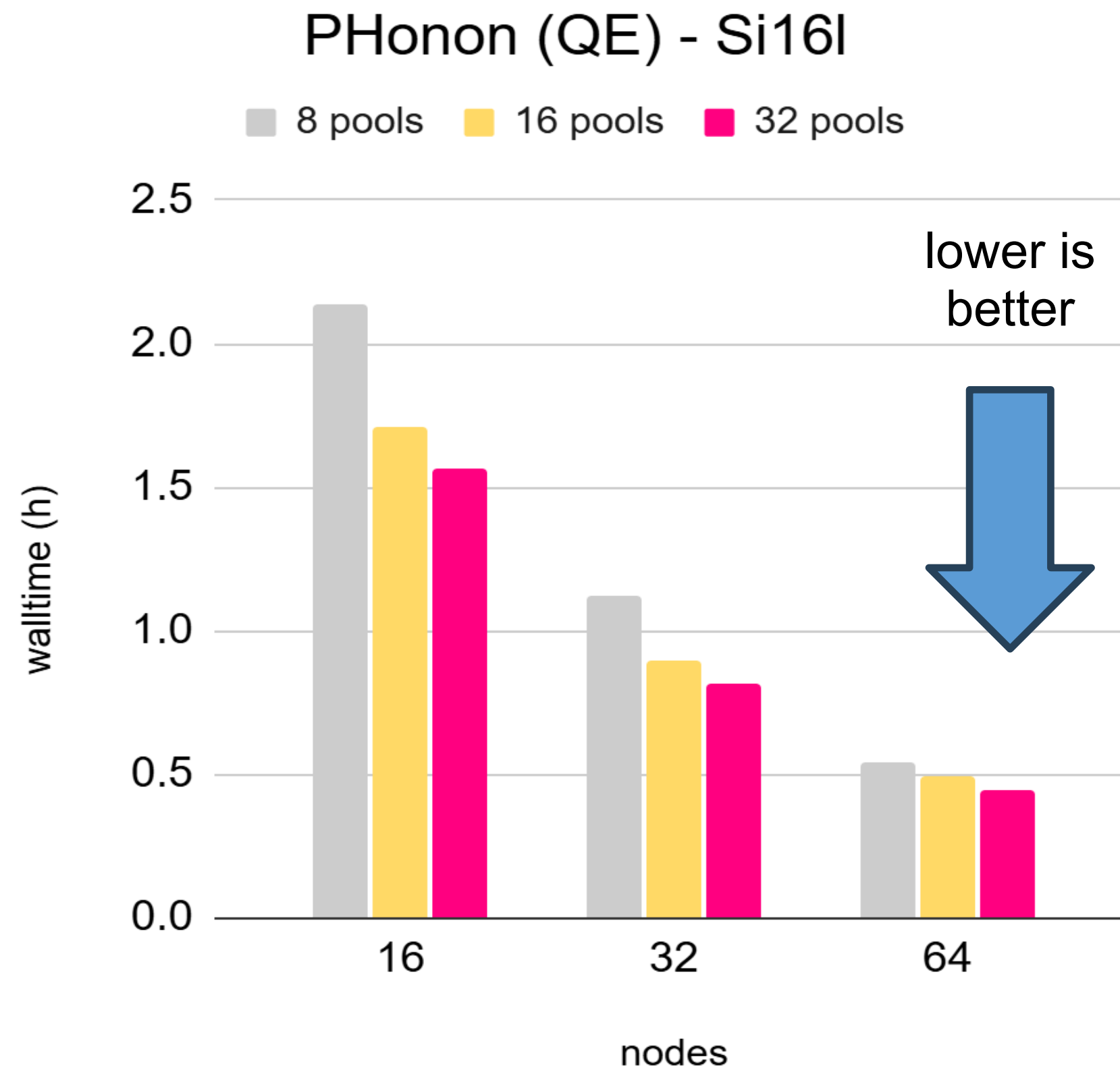
- **Nvidia cuSOLVERMp** interfaced and validated  
(Courtesy of S. Orlandini and M. Montagna — CINECA)

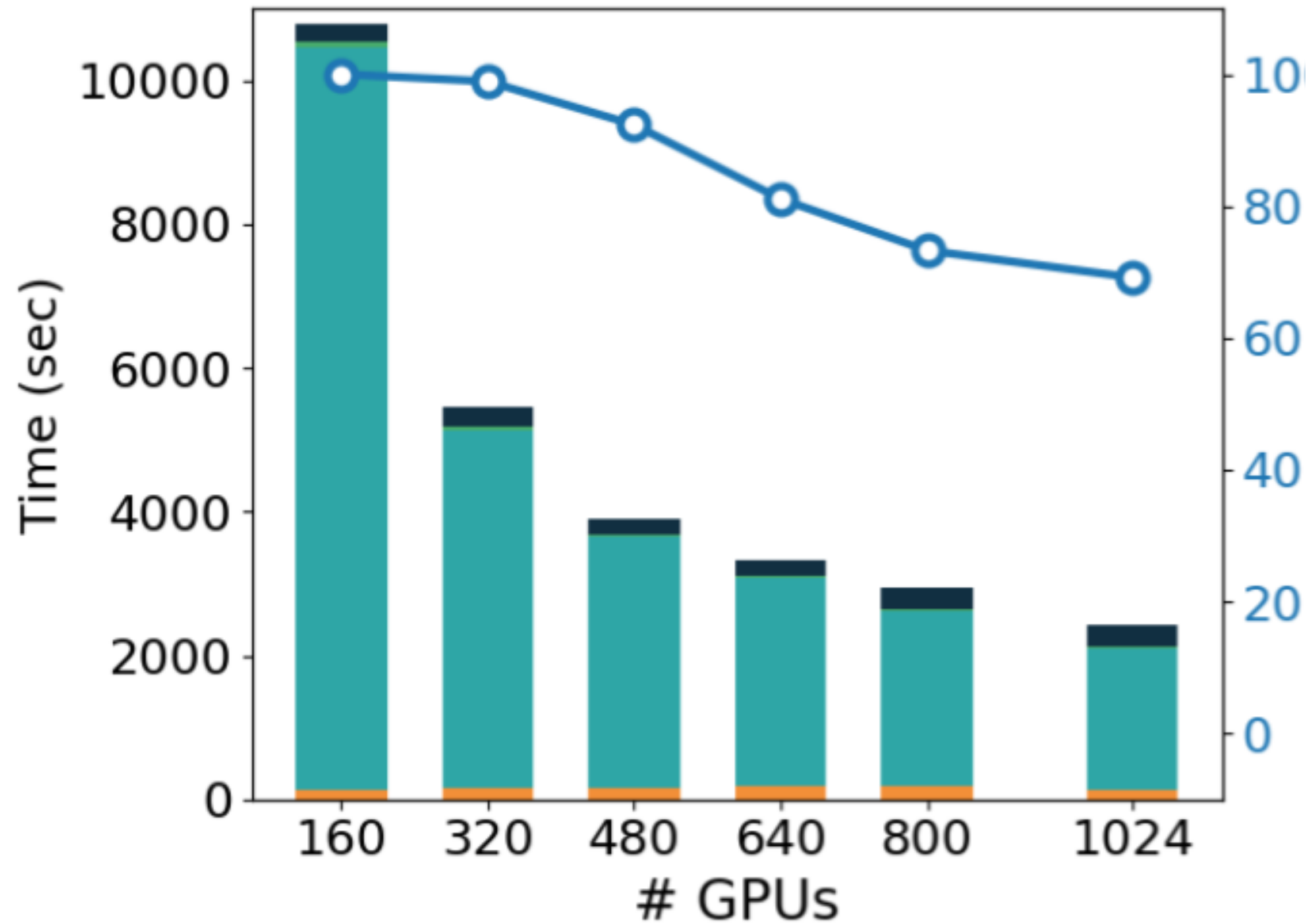
Nodes	Tasks/Node	cuSOLVERMp		ScaLAPACK		SU LA	SU func
		Linear Algebra	getrf/getrs	Linear Algebra			
1	4	118.16	73.58	14760.00	124.9x	200.6x	
4	4	116.46	61.05	4080.00	35.0x	66.9x	
16	4	119.72	60.41	1063.00	8.9x	17.6x	

Figure 12: Comparison of `cusolverMP` and `ScaLapack` in solving a dense linear system of size  $21453 \times 21453$ , as required by the calculation of the response function  $\chi$  in YAMBO.

# MPS (Multi Process Service)

**MPS** for small workloads minimizes the costs in terms of energy and improves the time to solution

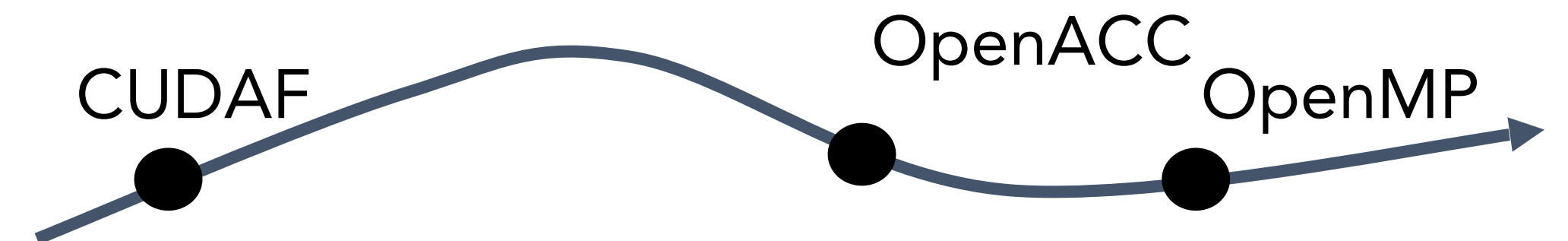
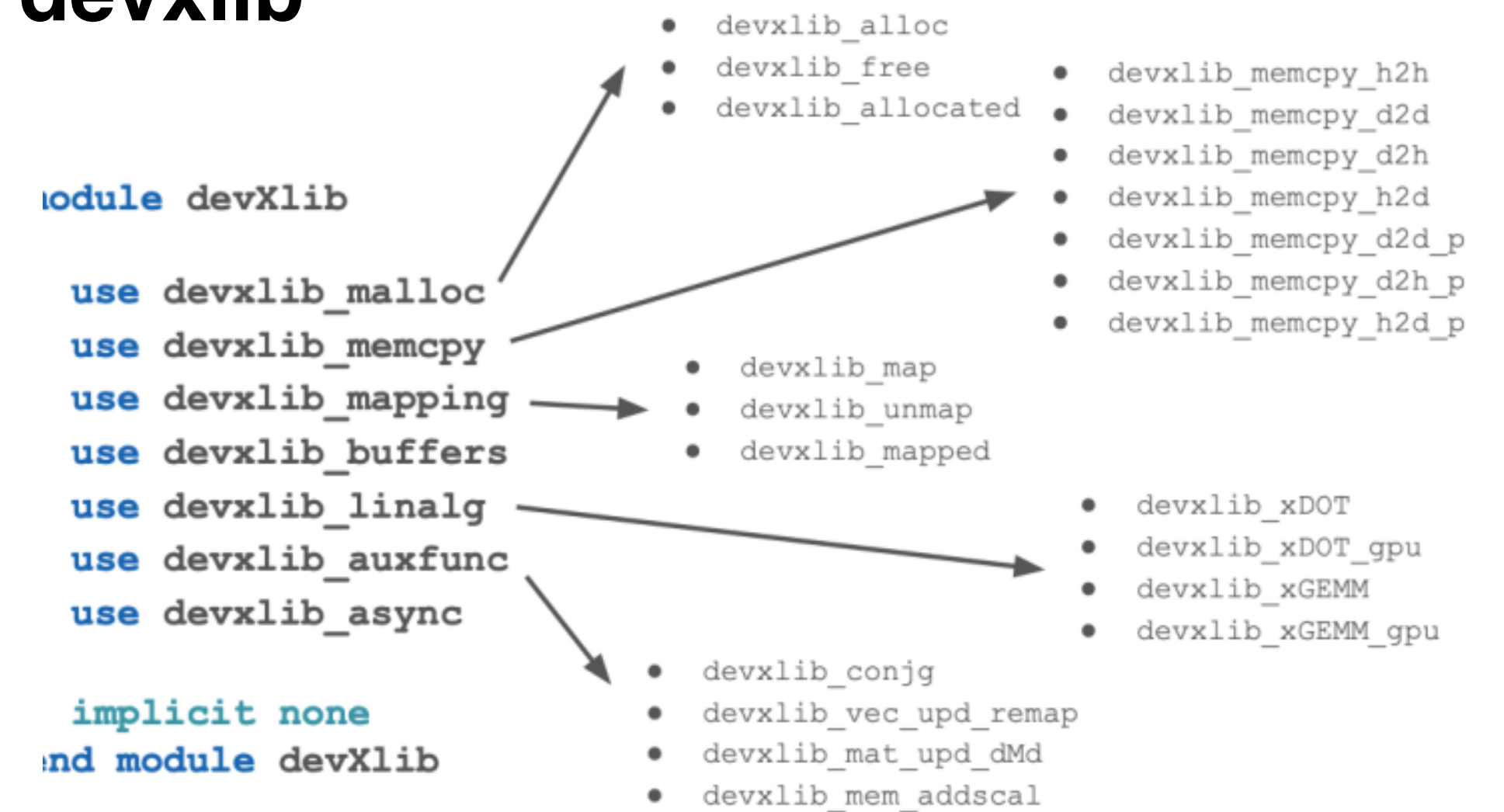




- yambo offers fairly independent parallelization layers
- porting to new programming model streamlined by devxlib

linear algebra, FFTs on GPUs, specialized backends, custom kernels. data mapping

## devxlib



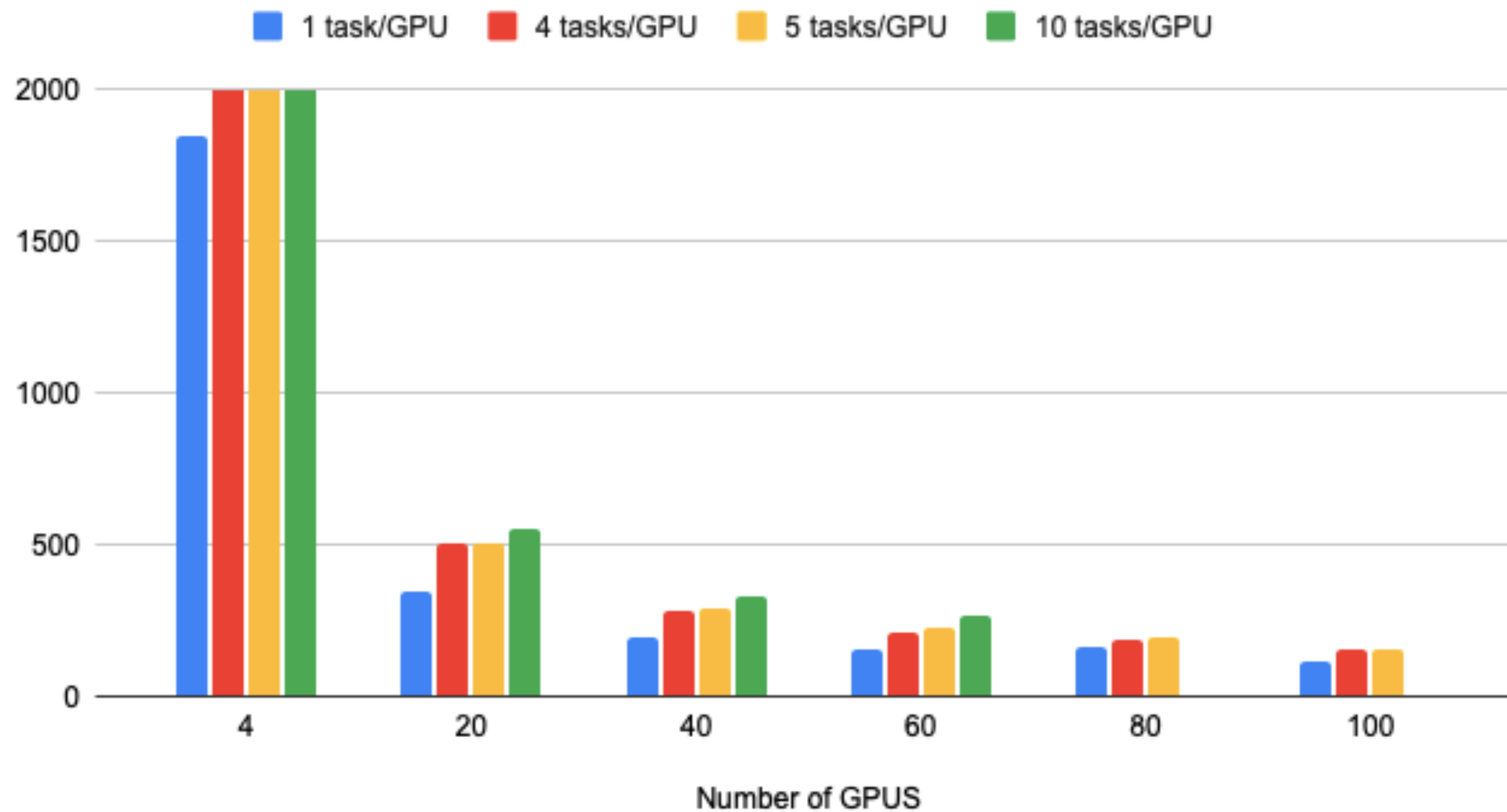


# Yambo @ MareNostrum-V

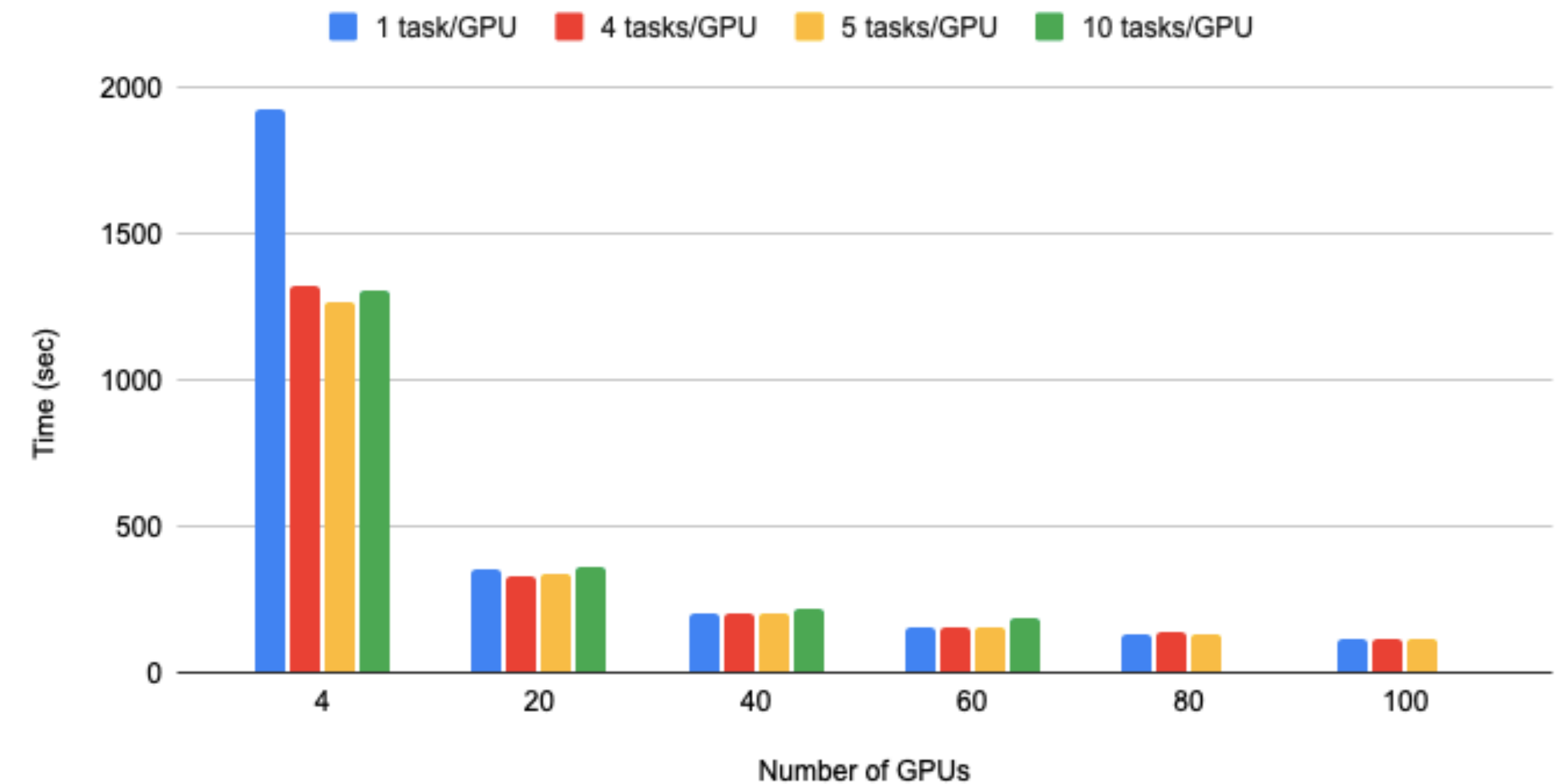
- Featuring **Nvidia H100**
- Use of Nvidia **MPS** tested
- Test relevant also for **porting on Jupiter**

Data: courtesy of J. Gutierrez and R. Grima (BSC)

GrCo(7k), Yambo v5.3, MarenostumV@BSC  
MPS off



GrCo(7k), Yambo v5.3, MarenostumV@BSC  
MPS on



**MPS off**



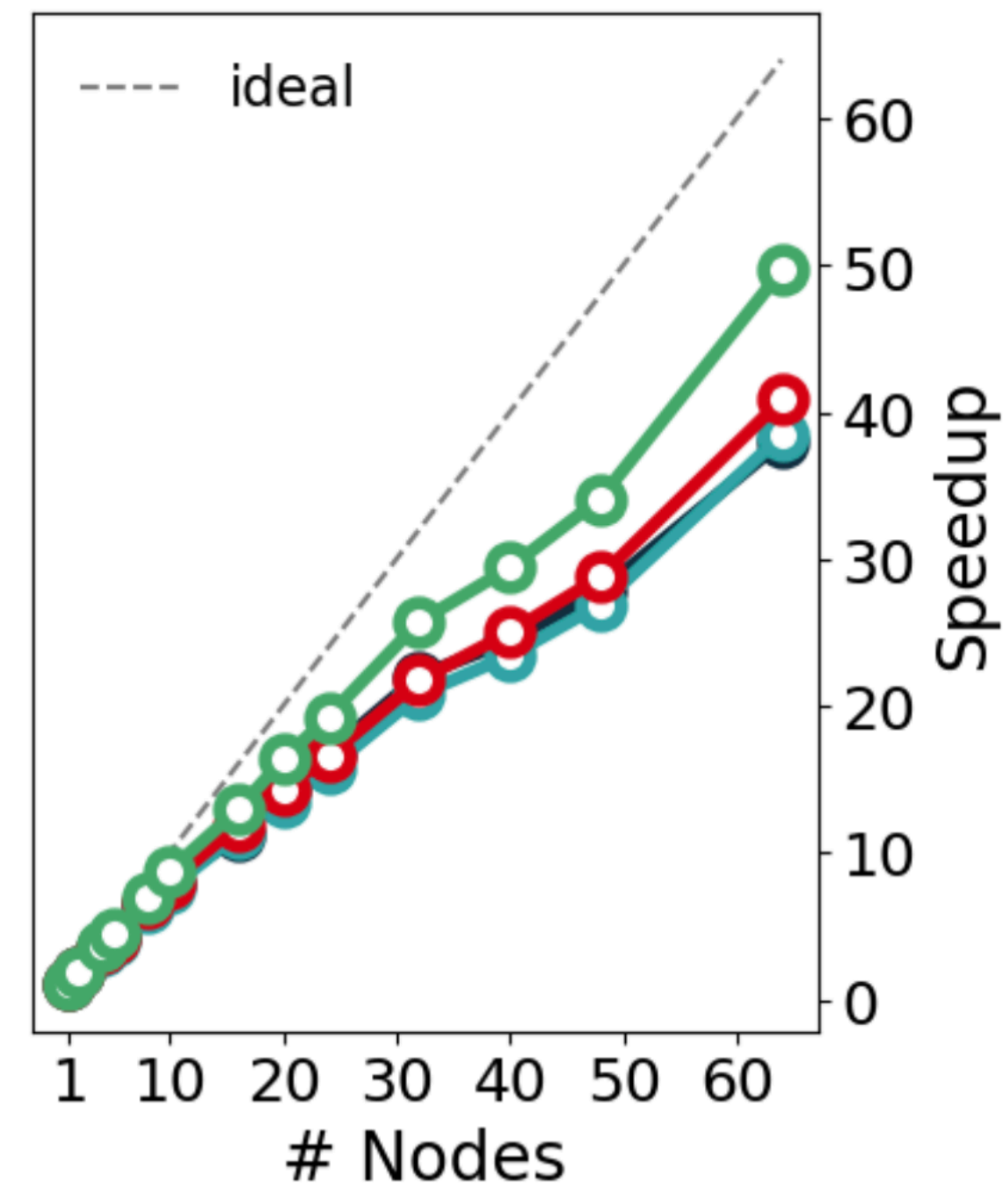
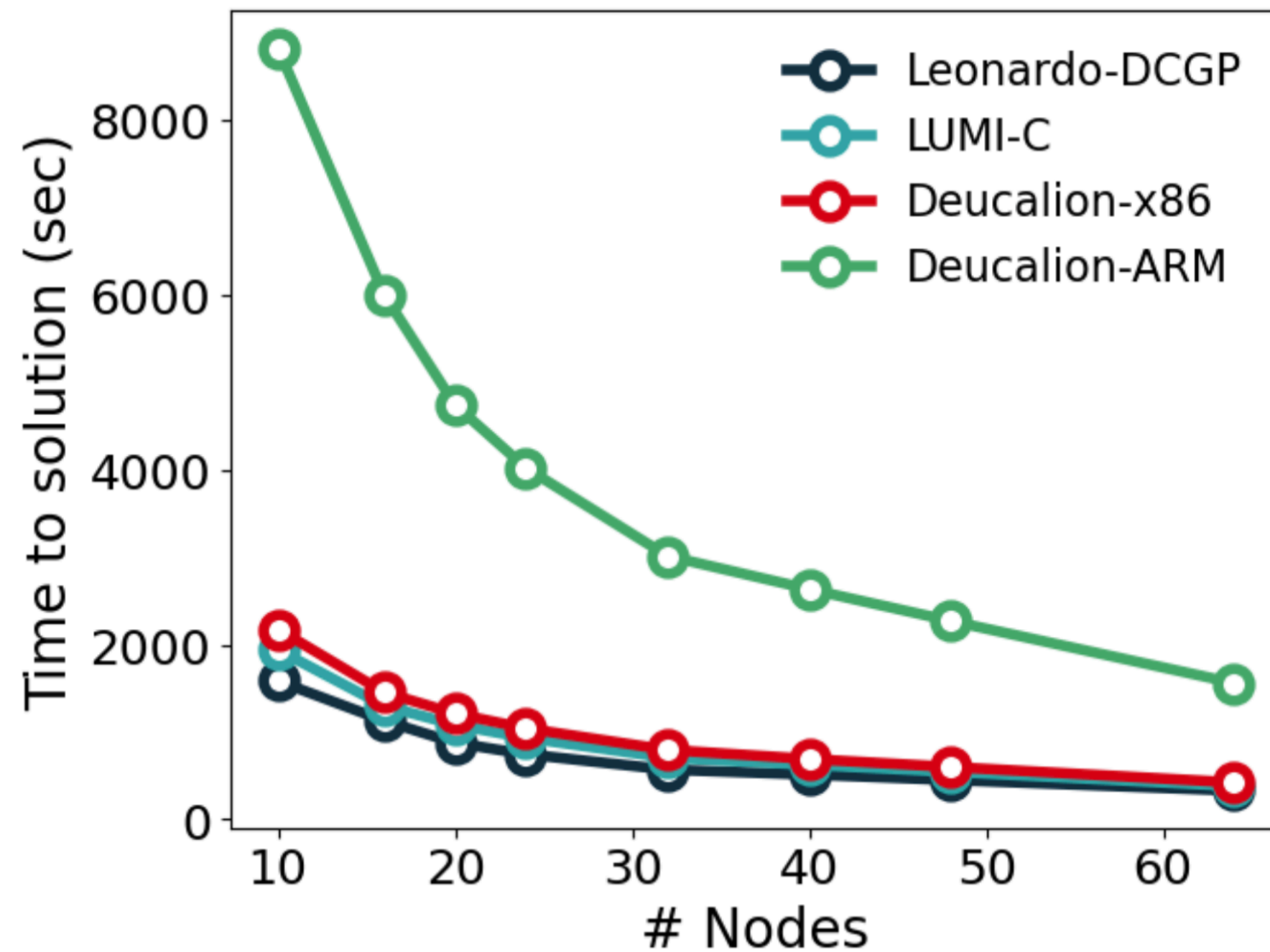
**MPS on**

# Yambo @ ARM-based chips

- Runs on **Deucalion** featuring **Fujitsu A64FX**
- Similar runs also on **Fugaku**
- Comparison with x86 chips (**Leonardo DCGP**, **LUMI-C**, **Deucalion-x86**)



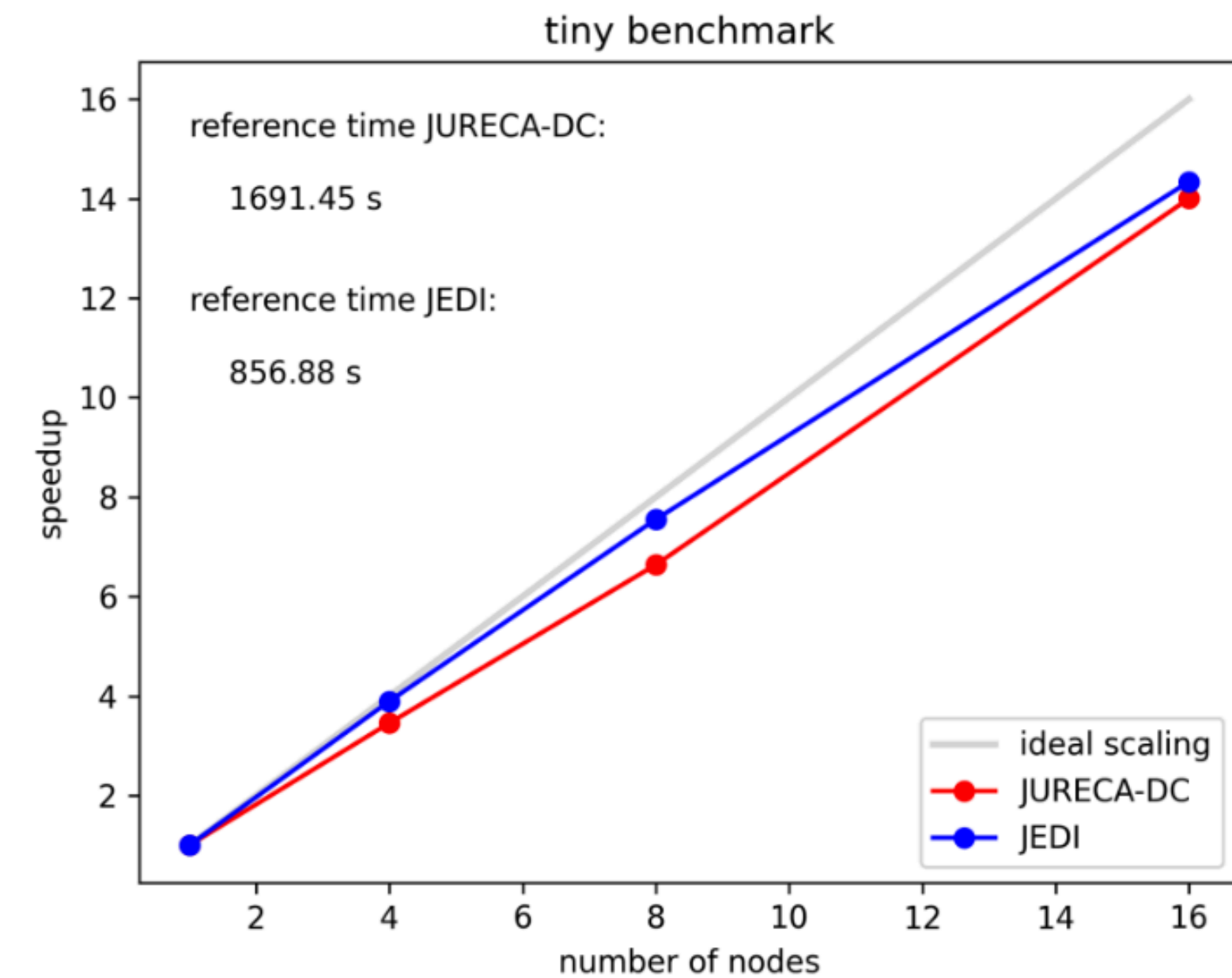
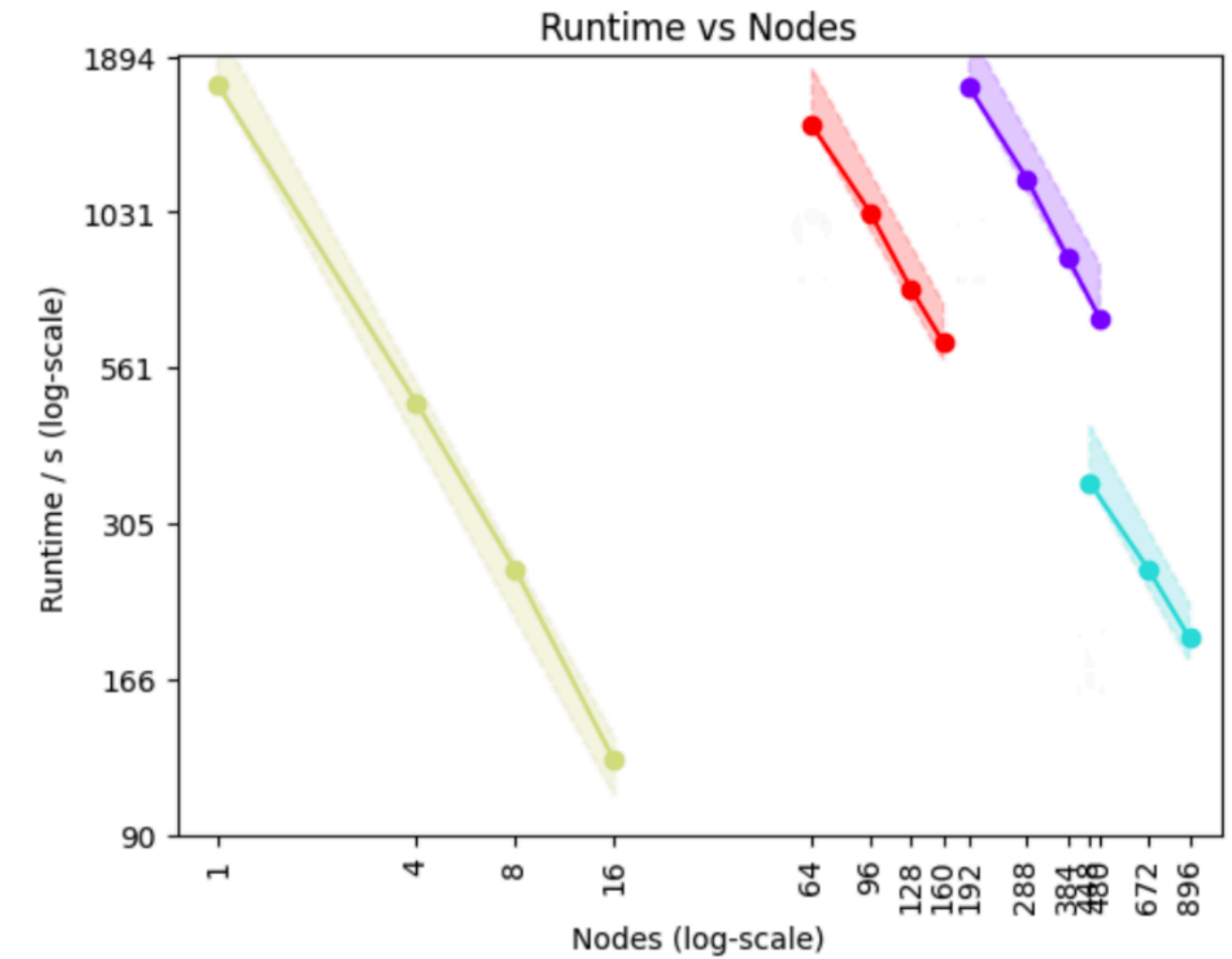
(Courtesy of Costanza Borghesi and Nicola Spallanzani)



# Fleur @JEDI / JUPYTER

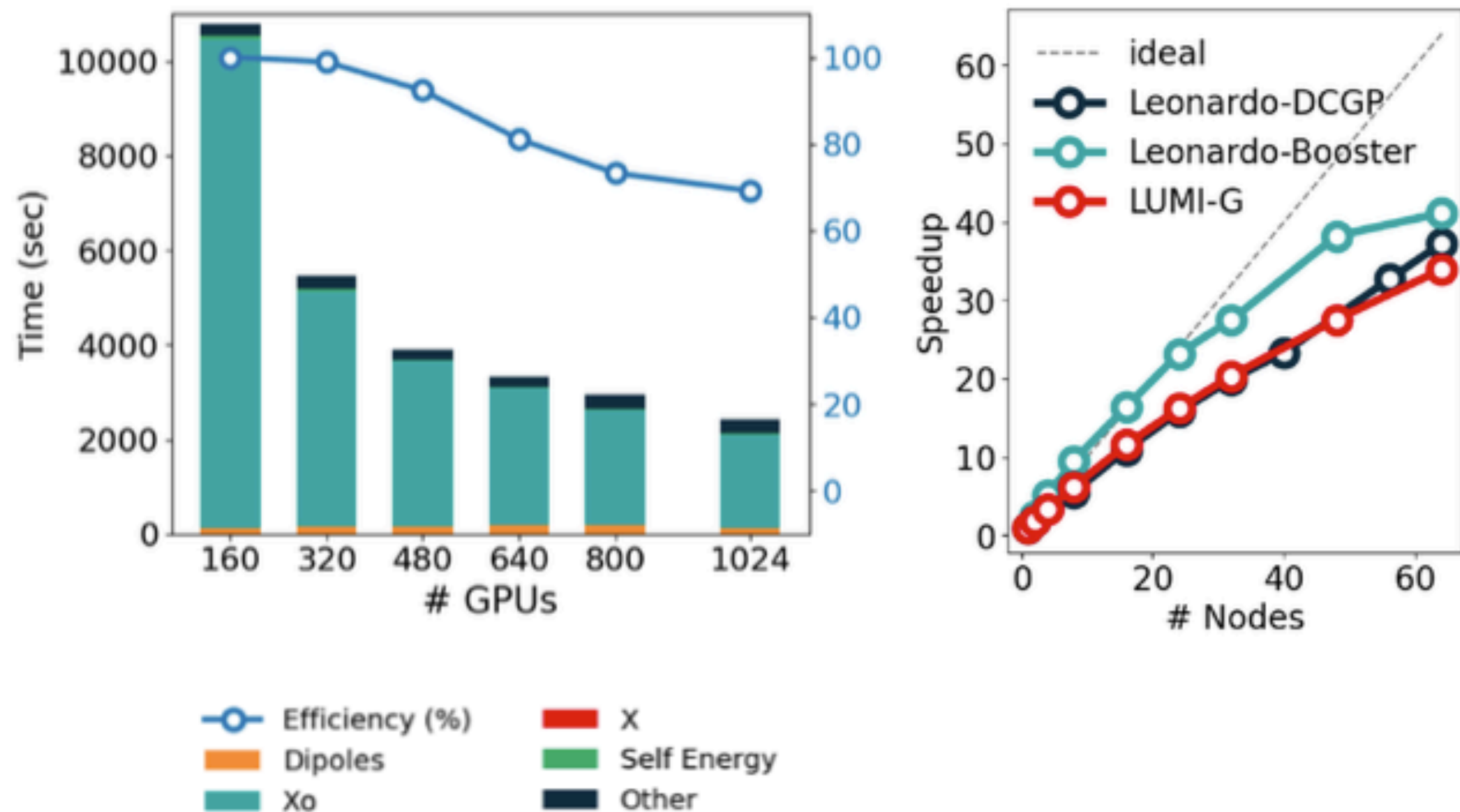


Figure 9: JUREAP certificate on FLEUR's readiness to run on JUPITER.



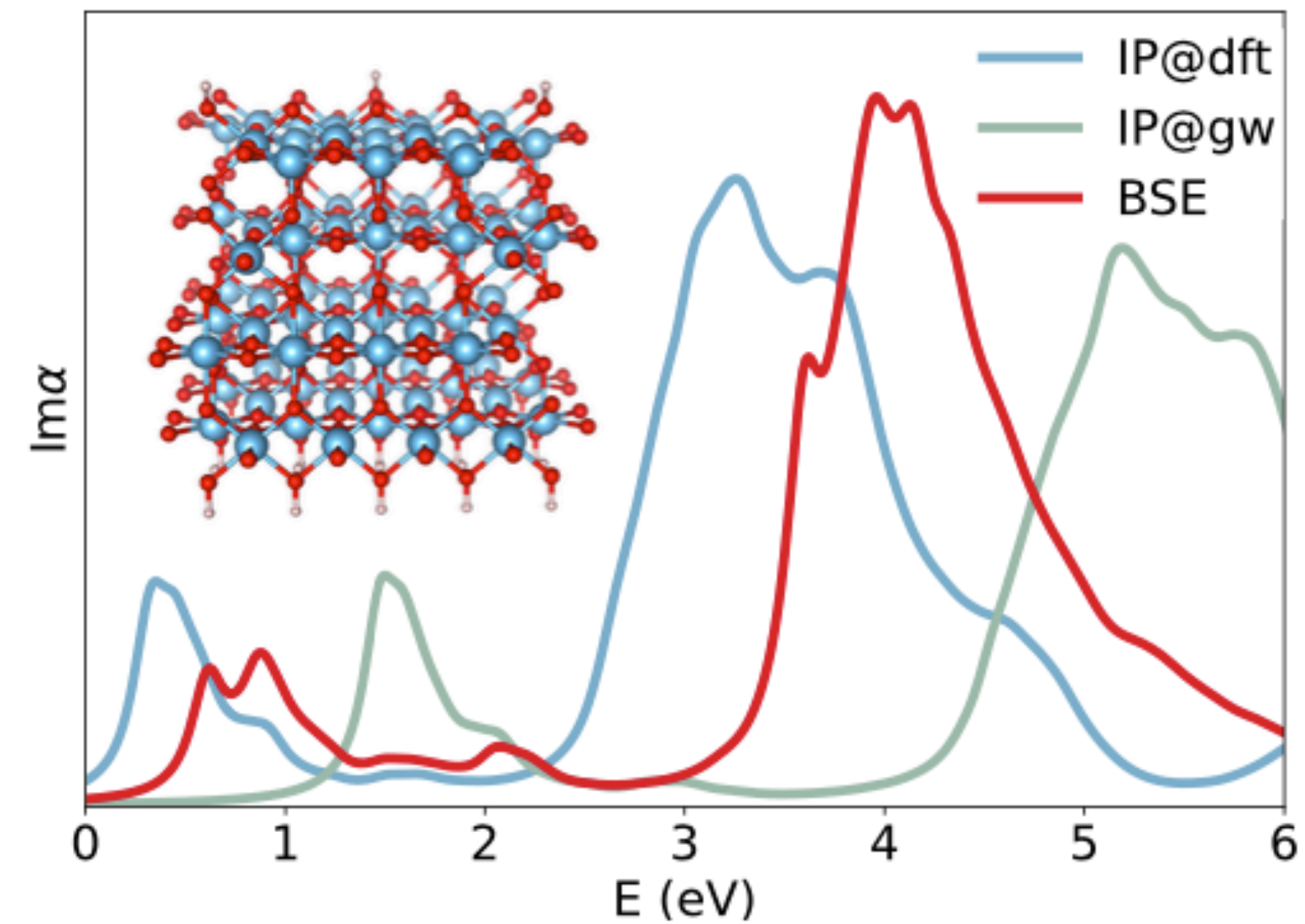
## STRONG SCALING AND SPEEDUP

Calculation of quasi-particle corrections on a graphene/Co interface (GrCo). The test involves the evaluation of the response function, Hartree-Fock, and correlation self-energy, treated at the GW level. On the left, the scalability test on LEONARDO-BOOSTER (CINECA), while on the right a speedup comparison between LEONARDO-DCGP, LEONARDO-BOOSTER and LUMI-G (CSC) for the same system but with reduced parameters.



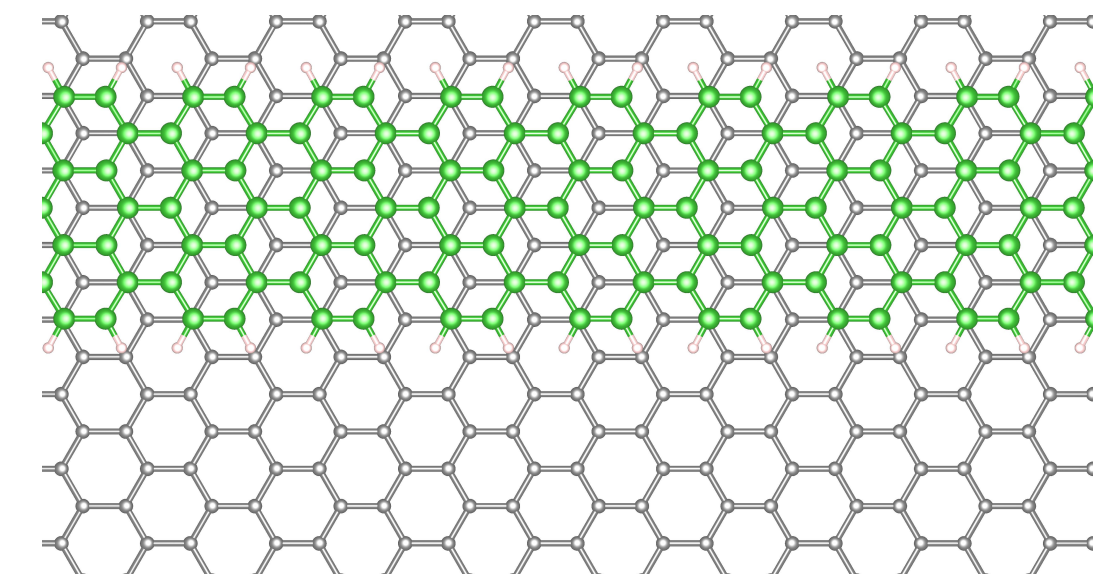
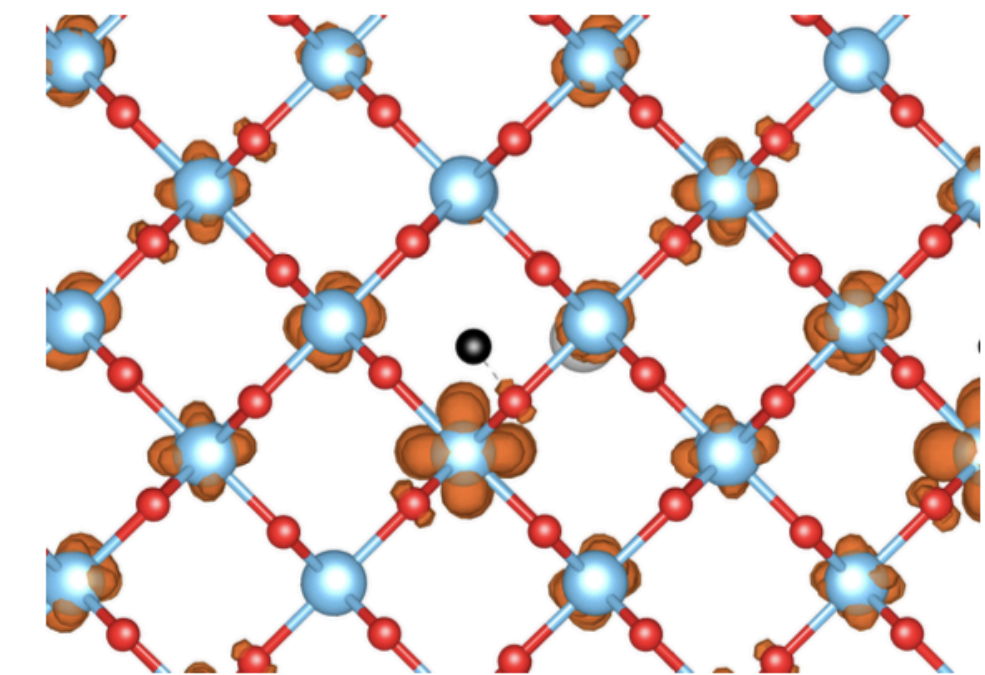
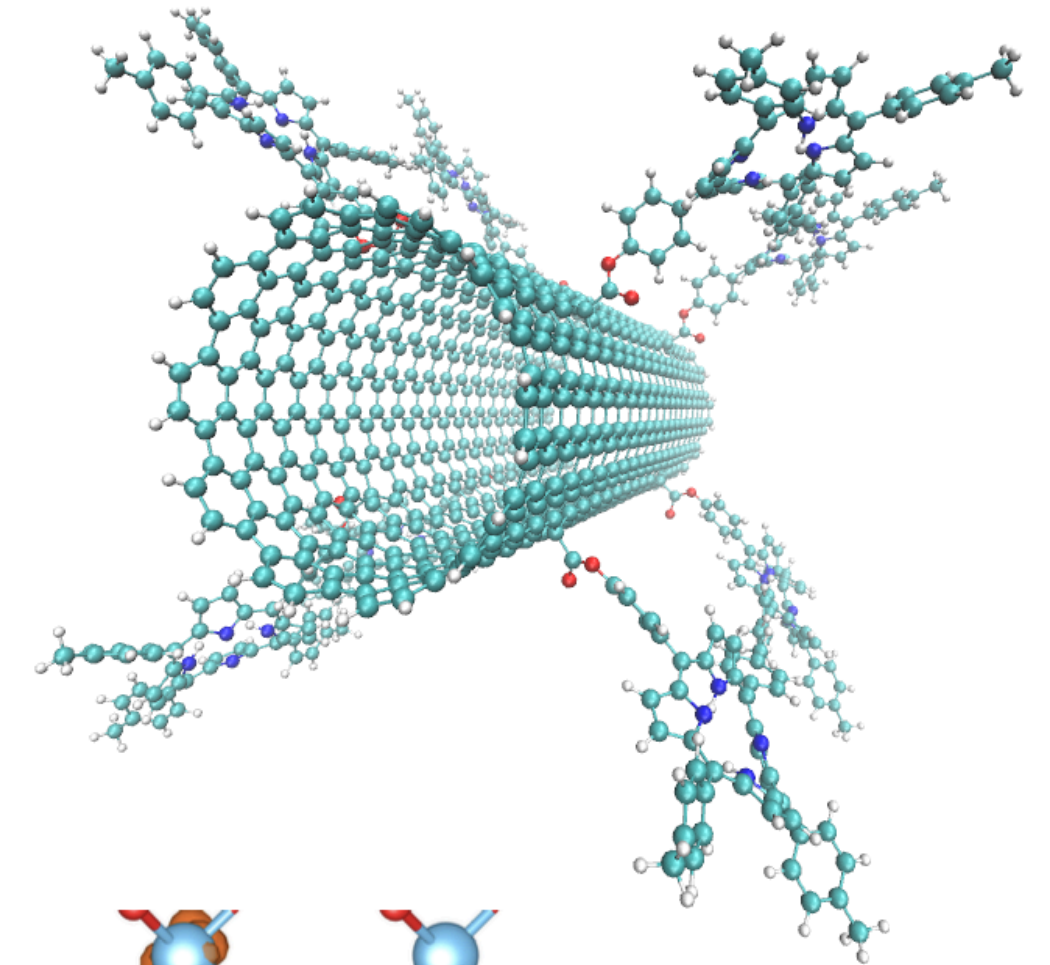
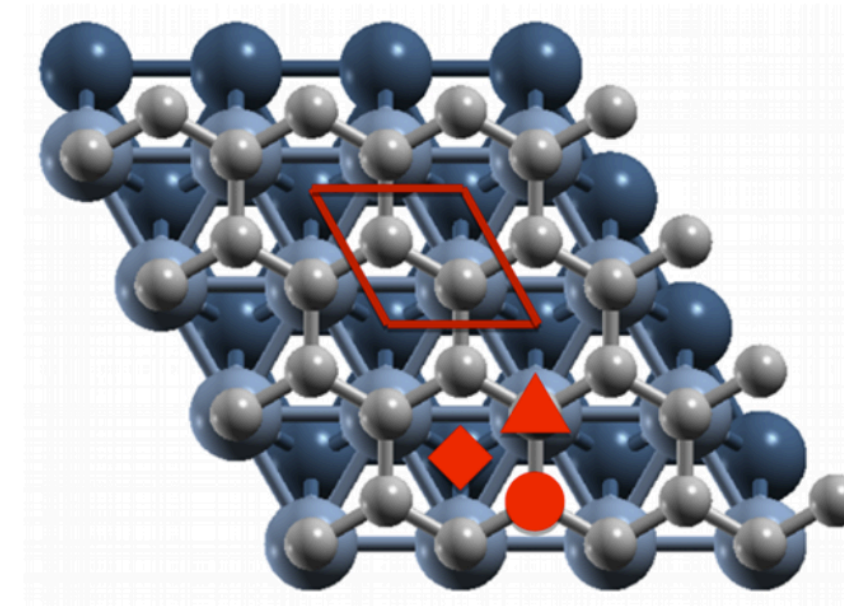
## OPTICAL ABSORPTION SPECTRA

Optical absorption spectra computed at different levels of theory using YAMBO. IP@dft, IP@gw: independent particle response, evaluated on top of Kohn-Sham DFT and GW electronic structure. BSE: response function from the Bethe-Salpeter equation. The system studied is a defected (hydroxylated) TiO<sub>2</sub> slab, relevant for photocatalysis, counting 210 atoms in the unit cell.



# Benchmarking: summary

- **Test cases identified** early on and **regularly updated**
- **data available** at: <https://gitlab.com/max-centre/benchmarks-max3/>
- Adoption of **JUBE**
- **Mostly TIME-TO-SOLUTION** so far, but also **COST-VS-TIME**
- Access to **hardware counters** and different metrics
- Analysis and tuning of **ENERGY-TO-SOLUTION**
- Used to validate the **parallel performance** and **performance portability** on the codes across different architectures
- Significant bulk of **data collected** and made available
- **Non-maturity** and **inhomogeneity of the software stacks** need to be taken into account
- Additional **assessment** needed when dealing with **workflows**





### Special thanks to:

- F. Affinito, L. Bellentani, I. Carnimeo, N. Spallanzani for sharing slides and information
- All MaX developers

Thanks !

### Follow us on:



[@max\\_center2](#)



<http://www.max-centre.eu/>



[youtube/channel/MaX Centre eXascale](https://www.youtube.com/channel/MaX_Centre_eXascale)

