



EAR ISC2024 tutorial: Monitoring

Julita Corbalan (julita.corbalan@eas4dc.com)

Benjamin Czaja (benjamin.czaja@surf.nl)



Monitoring

- Execution environment: Snellius
- Use cases and scripts in Snellius shared folder and Github
 - CPU use cases
 - GPU use cases
- Changing CPU frequency
 - CPU cases
- Energy efficiency vs Resource consumption
- Energy efficiency vs architecture





Snellius

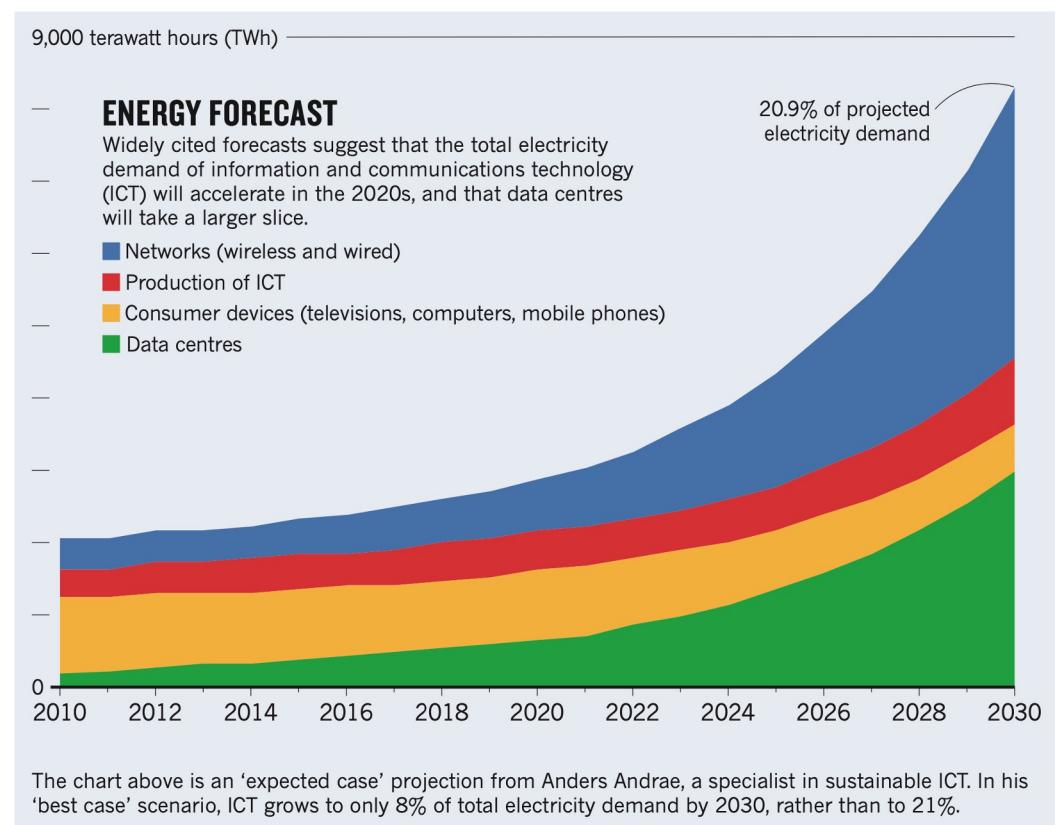
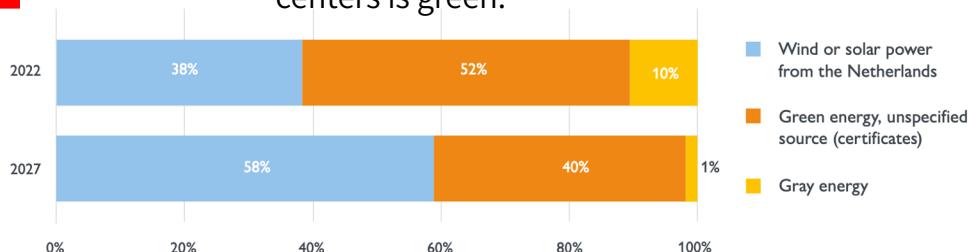
Dutch National Supercomputer

- Jones, Nicola. "How to stop data centres from gobbling up the world's electricity." *Nature* 561.7722 (2018): 163-166.
- Andrae, Anders SG, and Tomas Edler. "On global electricity usage of communication technology: trends to 2030." *Challenges* 6.1 (2015): 117-157.
- Bakkeren, Hanno. "Datacenters verbruiken drie keer zoveel stroom als de NS". NRC, 14 May 2019
- State of the Dutch Data Centers, The Dutch Data Center Report, 2022, <https://www.dutchdatacenters.nl/>



ICT/Data center energy forecast

- Data centre energy usage:
 - ~ 200 TWh for data centres in 2018
 - ~ 3000 TWh in 2030
- Dutch Data Center Usage (2019)
 - 1300 MW installed capacity
 - 0.3 % electricity usage of the Netherlands (CBS)
 - 3x as much power than the NS
- Dutch Data center energy consumption (2022)
 - 90% of all energy consumed by colocation data centers is green.



Snellius - Dutch National supercomputer

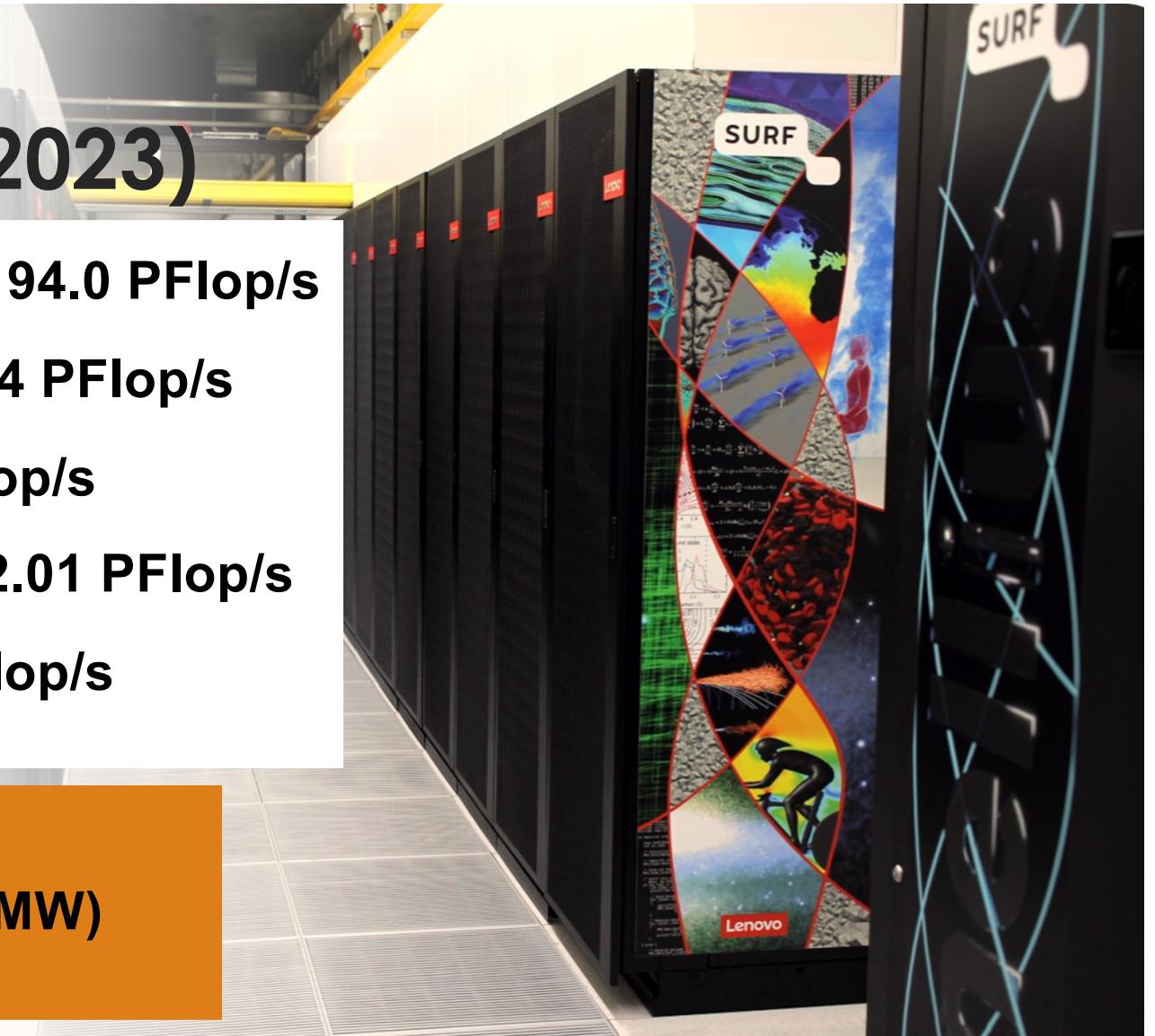
- AMD Rome (EPYC 7H12)
 - 500 nodes (128 cores-per-node)
 - 2.1 PFlop/s 0.5MW
- AMD Genoa (EPYC 9654)
 - 786 nodes (192 cores-per-node)
 - 5.4 PFlop/s 0.8 MW
- NVIDIA A100
 - 72 nodes (4 NVIDIA A100 SXM4 40 GB)
 - 3.6 PFlop/s. 0.2 MW



Top 500 (Nov-2023)

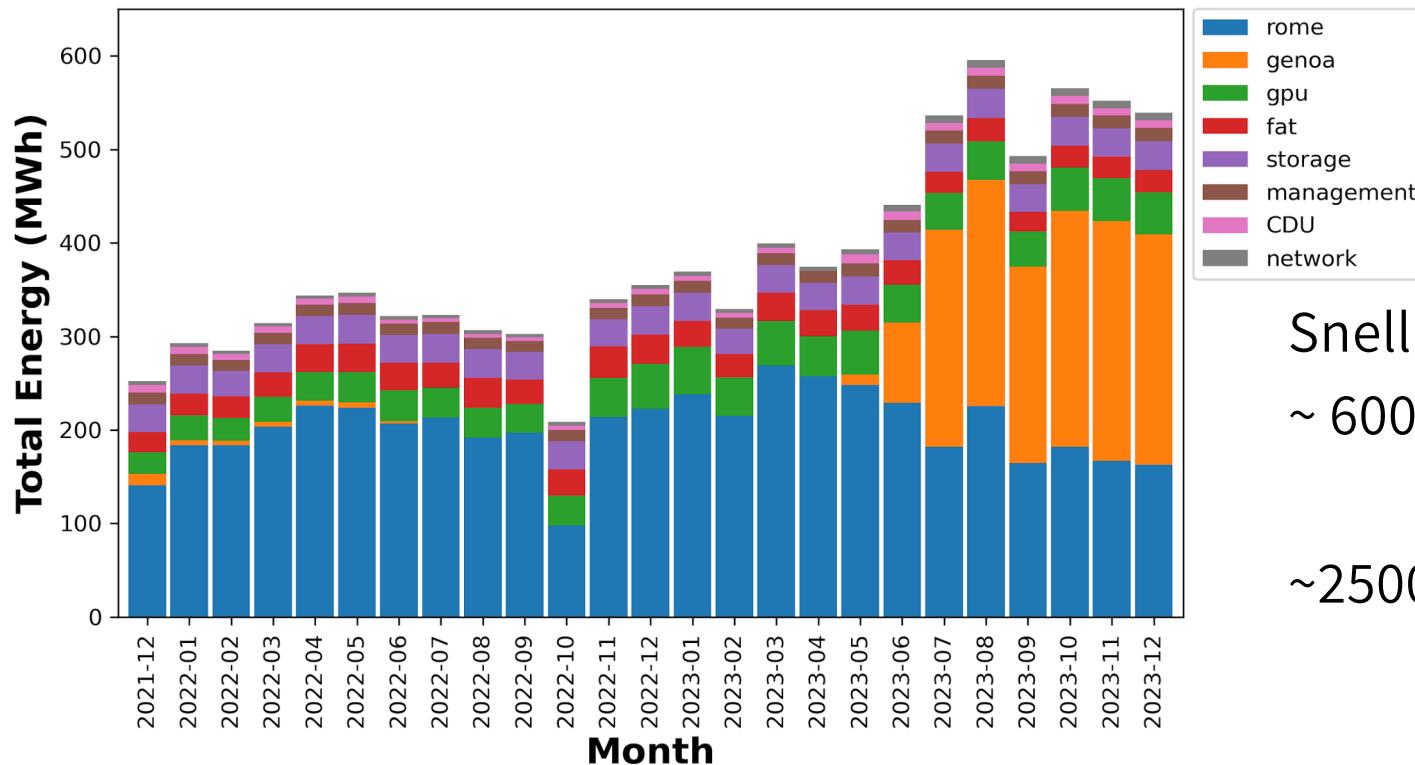
1. Frontier (22.7 MW) 1,194.0 PFlop/s
2. Aurora (24 MW) 585.34 PFlop/s
3. Eagle (??) 561.20 PFlop/s
4. Fugaku (29.9 MW) 442.01 PFlop/s
5. LUMI (7.1 MW) 379 PFlop/s

Snellius
~ 11.1 Pflops (1.5 MW)





Snellius - Total Energy Usage



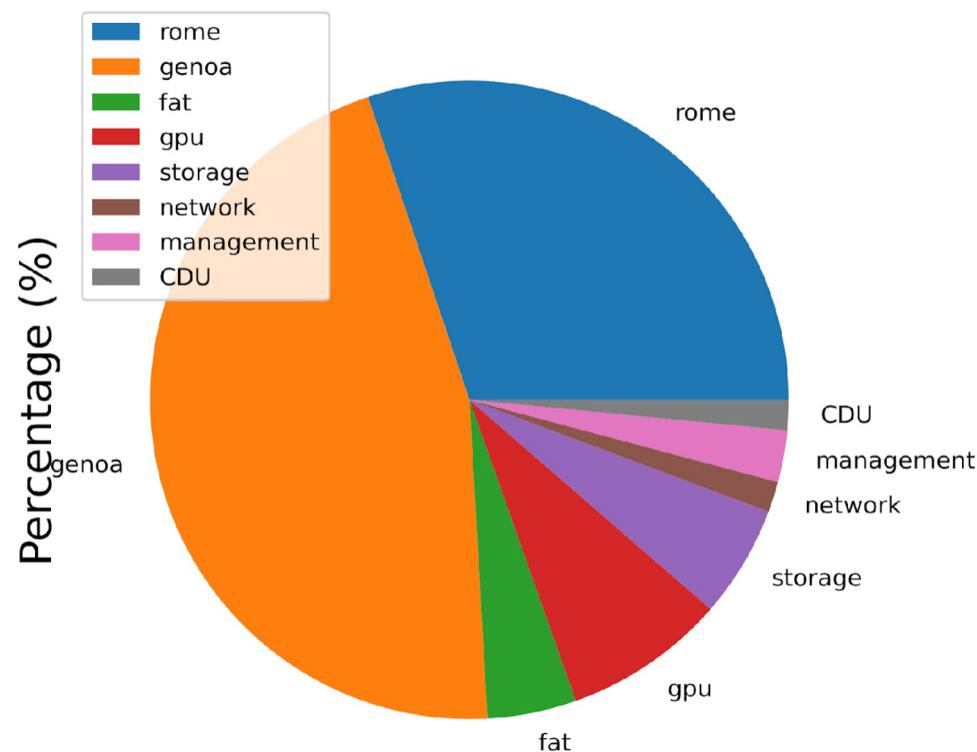
Snellius in December used
~ 600MWh of energy.

~2500 house holds worth



Snellius - Energy usage per partition

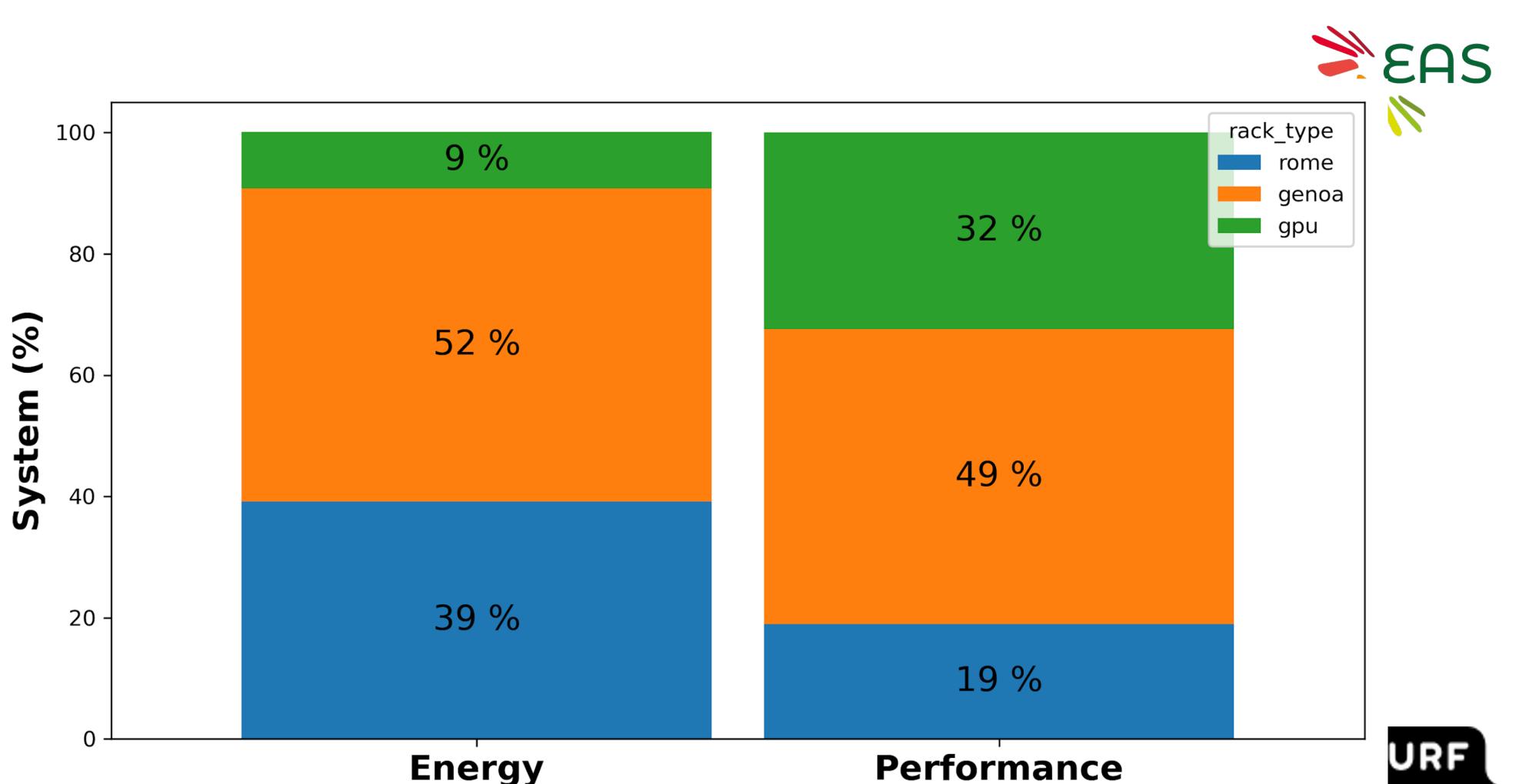
Snellius energy usage December 2024



- 88% Compute
 - 80% CPU
 - 34% Rome (3% fat)
 - 45% Genoa
 - 8% Gpu

- 12% Other
 - 5.0% Storage
 - 2.6% Management
 - 1.5% Cooling
 - 1.5% Network







Exercices

1- Basic EAR monitoring

- Codes in /projects/0/energy-course
- GIT: <https://github.com/sara-nl/ISC-2024-EAR-tutorial/tree/main>
- Get the examples and test them
 - https://github.com/sara-nl/ISC-2024-EAR-tutorial/tree/main/tutorials/monitoring_ear
 - NPB CPU use cases
 - GROMACS-GPU
 - HemePure CPU
 - HemePure GPU
 - PyTorch
 - Or select another use case from the list
- Wait for jobs to finalize
- Get the ear metrics
 - module load ear
 - eacct -j jobid1
- Understand the metrics



11

<https://www.eas4dc.com>

For each case

- Execute with ear=on and get the metrics
 - Per job
 - Per node (if it applies)
 - Runtime metrics
- Understand application characteristics
 - Is my application CPU bound?
 - Is my application Memory bound?
 - Is it a power hungry application? Are we close to the TDP?
 - Does my application shows IO activity?
 - Does my application shows high MPI percentage?
 - Does my application present phases of execution?
- For GPU application
 - Is my application pure GPU? Or CPU/GPU ?
 - What is the GPU activity and power consumption? Are we close to the TDP?



12

<https://www.eas4dc.com>



eacct options

- eacct -j jobid -l → show average metrics (per-node)
- eacct -j jobid -r → shows runtime metrics (per-node)
- eacct -j jobid -r c filename → saves ear metrics in csv file

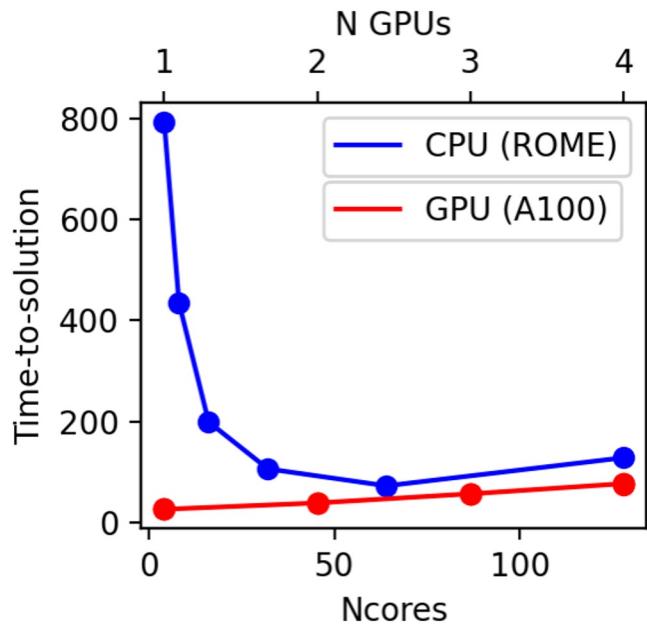
2- Static energy optimization

- What is the effect of changing my resource requirements? (Use GROMACS)
 - Number of Nodes
 - Number of tasks
 - Ratio tasks/cpus-per-task
 - Ratio tasks per GPU
- What is the effect of changing the CPU frequency? (use NPB)
 - EAR provides its own list of CPU frequencies, use enode_info to get the list
 - Is it worth to do it in CPU bound cases?
 - Is it worth to do it in Memory bound cases?

3- Energy efficient architectures

- GENOA nodes consumes more power than ROME nodes but....Are they more energy efficient (use NPB)
 - What is the total energy consumption for the same problem?
 - What's the ratio performance variation vs Energy/Power variation?

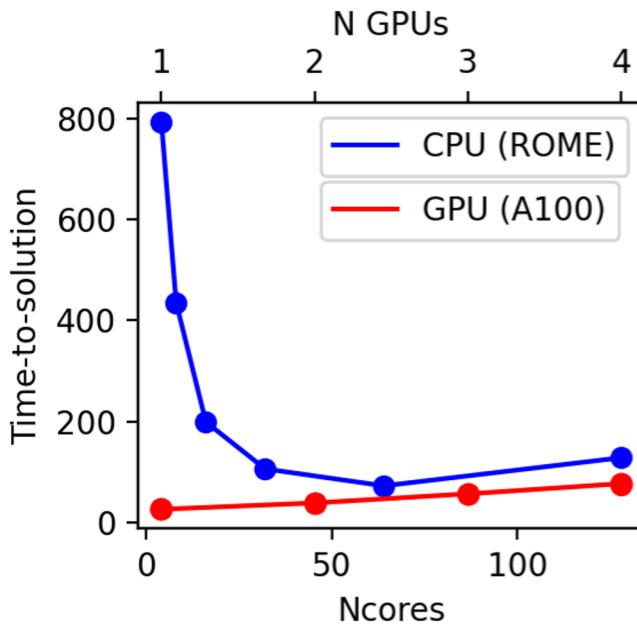
HemeLB: A Computational Science Use-case



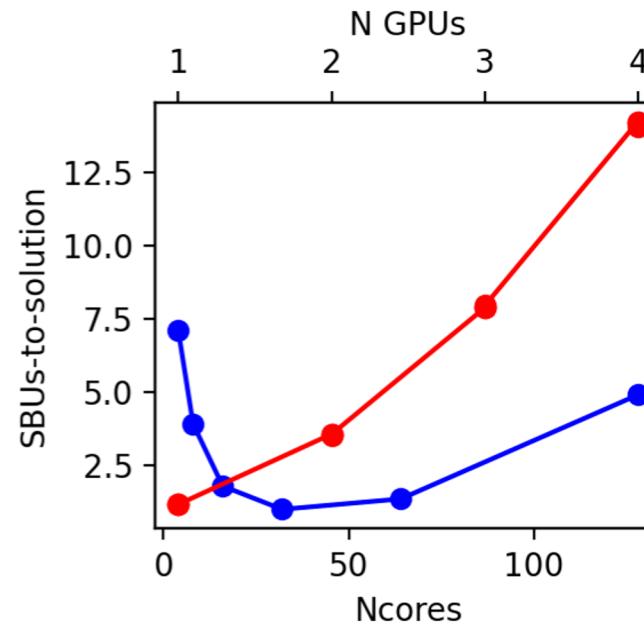
Performance:
1 NVIDIA A100 GPU



HemeLB: A Computational Science Use-case



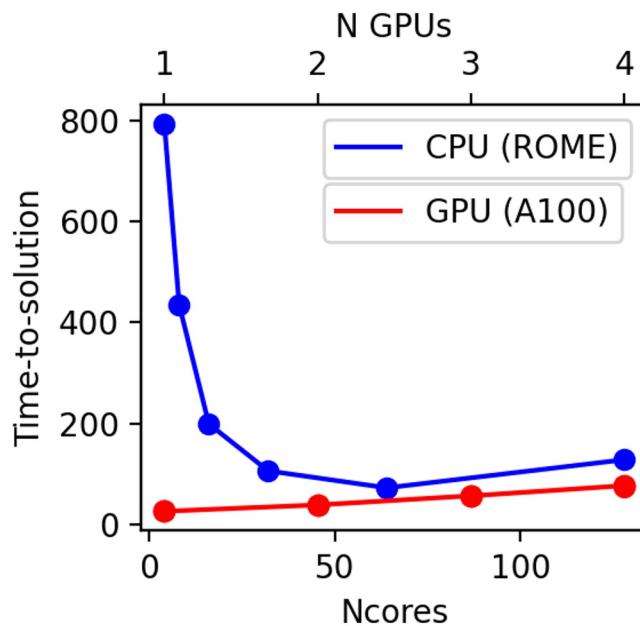
Performance:
1 NVIDIA A100 GPU



Budget:
32 AMD Rome cores

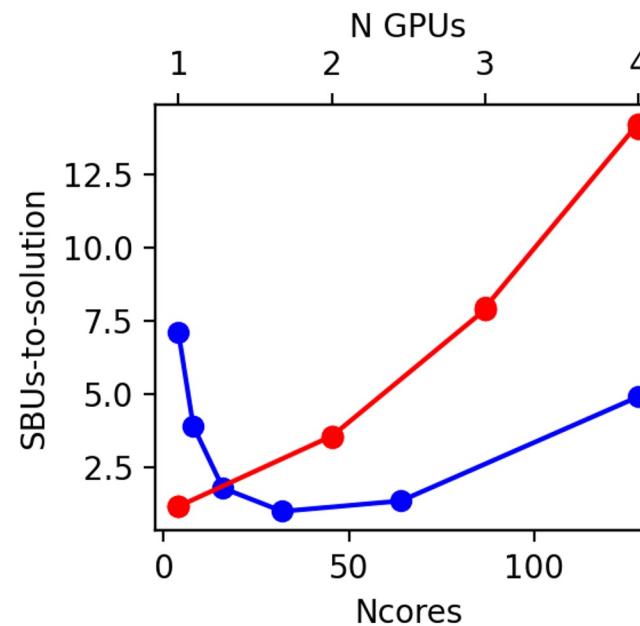
HemeLB:

A Computational Science Use-case

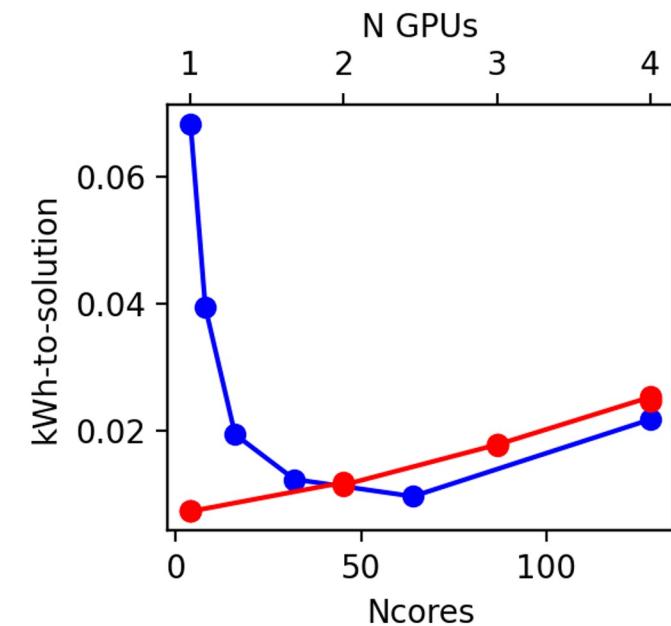


Performance:
1 NVIDIA A100 GPU

ISC 2024 EAR tutorial



Budget:
 $\frac{1}{4}$ AMD Rome

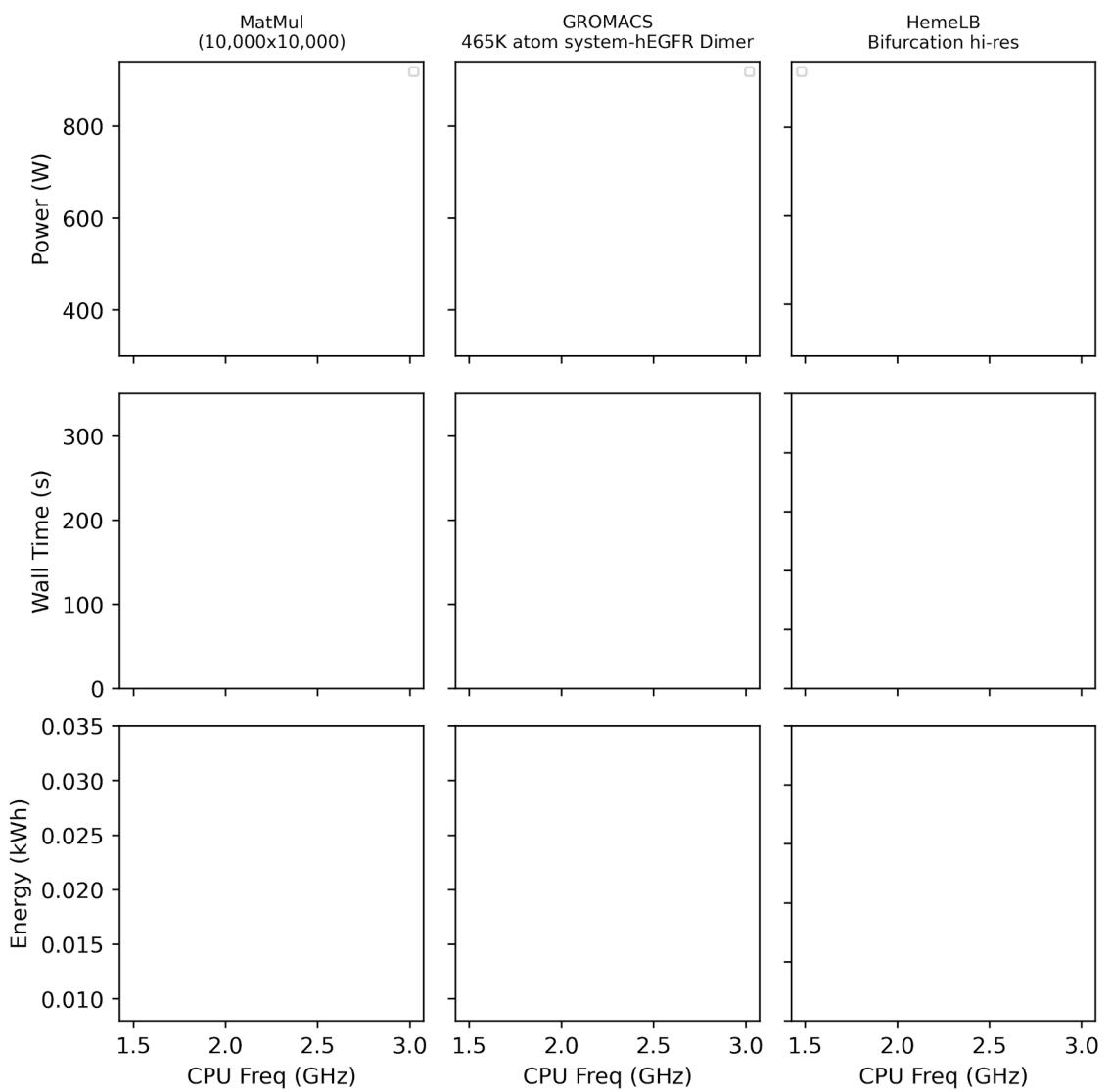


Energy:
1 NVIDIA A100 GPU
<https://www.eas-tutorial.com>

Dynamic Voltage Frequency Scaling on Snellius:

To boost, or not too boost, that is the question

- Rome “Zen2” Released 07 August 7 2019
 - Nominal Freq (2.6 Ghz)
- Genoa “Zen4” Released 10 November 2022
 - Nominal Freq (2.4 Ghz)



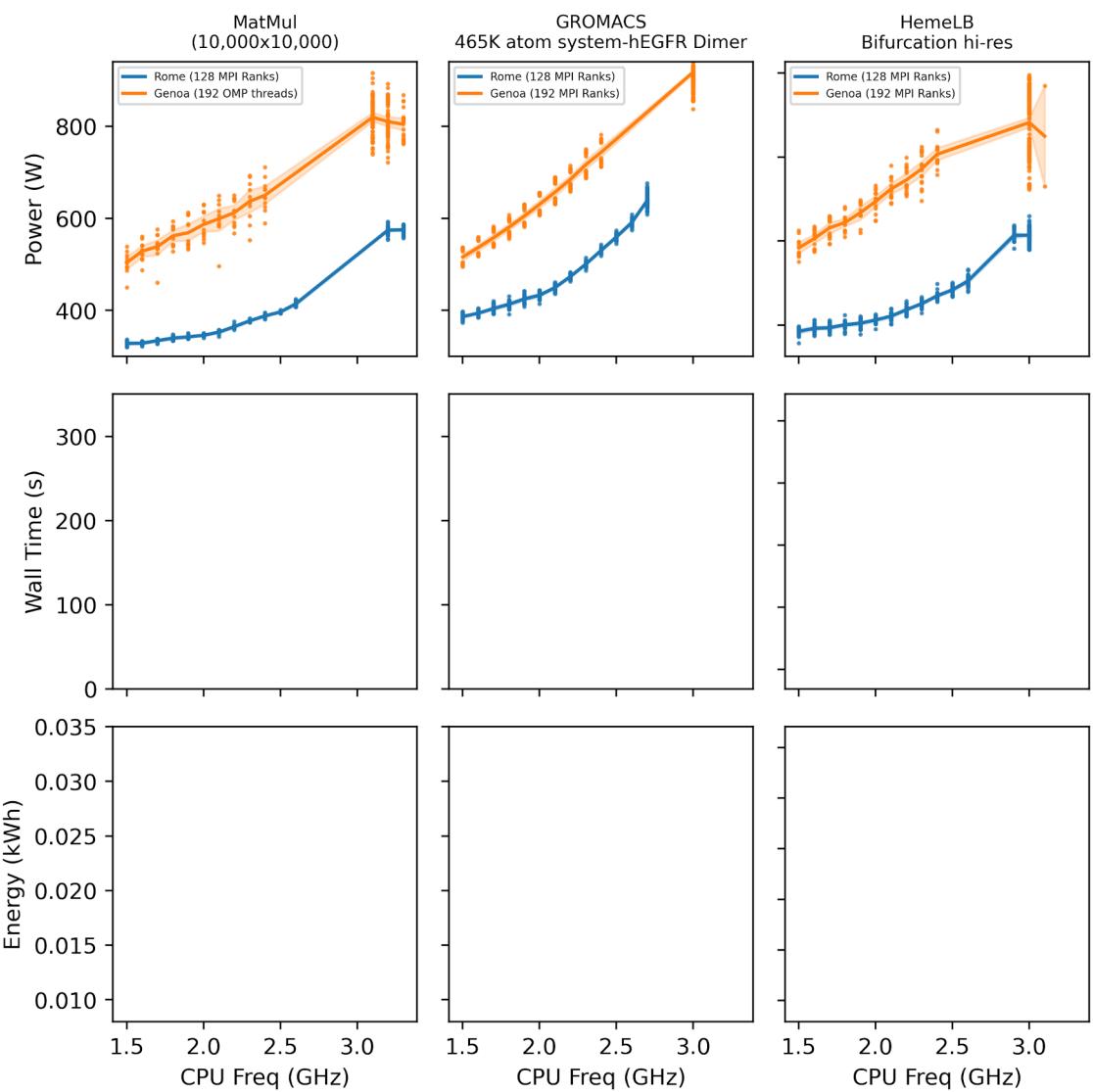
Dynamic Voltage Frequency Scaling on Snellius:

To boost, or not too boost, that is the question

- Rome “Zen2” Released 07 August 7 2019
 - Nominal Freq (2.6 Ghz)
- Genoa “Zen4” Released 10 November 2022
 - Nominal Freq (2.4 Ghz)

Observations:

1. Genoa draws more power
2. Each application has a different “power draw signature”



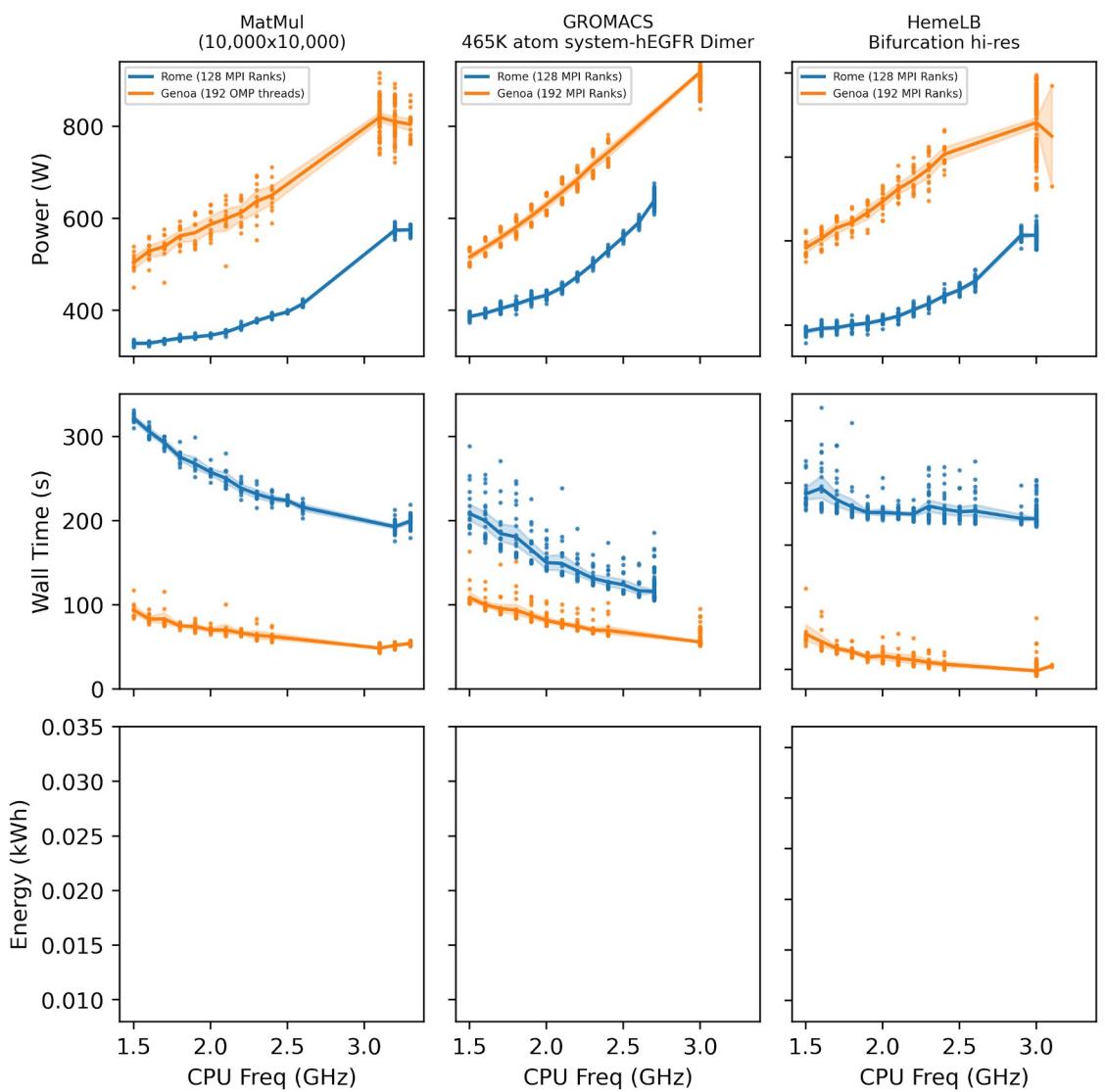
Dynamic Voltage Frequency Scaling on Snellius:

To boost, or not too boost, that is the question

- Rome “Zen2” Released 07 August 7 2019
 - Nominal Freq (2.6 Ghz)
- Genoa “Zen4” Released 10 November 2022
 - Nominal Freq (2.4 Ghz)

Observations:

1. Genoa is more performant
 - More Logical cores to devote to problem
2. In the 3 cases, Boost does not have the similar effect in performance vs power draw. i.e. its performance increase is flat as compared to its power draw



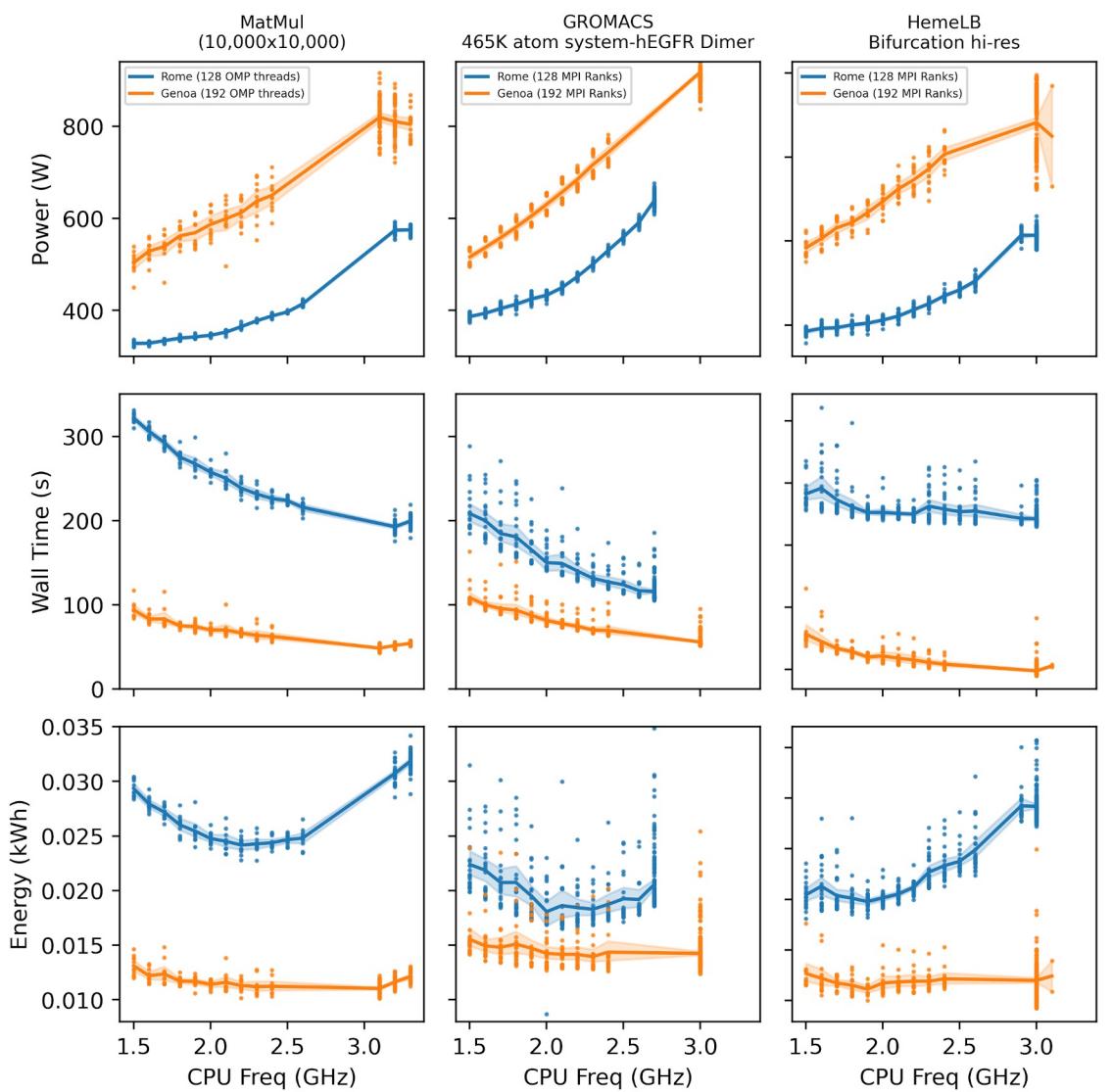
Dynamic Voltage Frequency Scaling on Snellius:

To boost, or not too boost, that is the question

- Rome “Zen2” Released 07 August 7 2019
 - Nominal Freq (2.6 Ghz)
- Genoa “Zen4” Released 10 November 2022
 - Nominal Freq (2.4 Ghz)

Observations:

1. Genoa is more energy efficient.
2. Each application has its own energy minimum.
3. DVFS “by hand” is **HARD!!!!!!**





Go to ISC tutorial github and do
the exercises

https://github.com/sara-nl/ISC-2024-EAR-tutorial/tree/main/tutorials/monitoring_ear



23

<https://www.eas4dc.com>