Towards JUPITER: Evaluating Energy Usage of Applications on GH200

Jayesh Badwaik, Sebastian Achilles, Kaveh Haghighi Mood, Mathis Bode, Andreas Herten
Jülich Supercomputing Centre, Forschungszentrum Jülich

Introduction

Energy efficiency and **energy-to-solution** (E2S) metrics are becoming more and more important in the exascale era, where small improvements at the GPU or node level can accumulate to significant **energy and cost savings**. **JUPITER** is the first European exascale supercomputer, currently being set up at Forschungszentrum Jülich. The machine is the first exascale supercomputer powered by **NVIDIA GPUs**, highlighting the need to study energy behavior on these GPUs across a diverse set of applications.

Naively, optimizing for best performance is a good a-priori choice to also optimize for least energy usage. But with complex algorithmic profiles and different hardware-architectural features, the connection between energy and performance is not so direct anymore. **Down-clocking** GPU core frequencies offers a **simple, algorithm-agnostic method** to optimize for energy usage, providing a practical user-configurable approach to reducing energy consumption without interfering with application functionality.

This poster presents **first research** within the JUPITER Research and Early Access Program (JUREAP) to optimize energy usage through GPU clock frequency variation. A **scalable framework**, exaCB, is employed to offer the optimization to a large corpus of applications without manual intervention.
As **case studies**, one synthetic benchmark (HPL, compute bound) and two actual applications (MPTRAC, SOMA) are shown, investigating frequency dependence of E2S on different systems.

Systems

JEDI (with orange light), the JUPITER Exascale Development Instrument, is the first building block of JUPITER.

JUWELS is a multi-petaflop modular supercomputer with its booster partition featuring 936 nodes with 4 NVIDIA A100 GPUs each.

Devices

GH200

- Grace-Hopper superchip
- JUPITER: 6000 nodes, 4×GH200 each
- Grace: Arm Neoverse-V2 CPU (72 cores); 120 GB LPDDR5X memory
- Hopper: H100 GPU variant (132 multiprocessors); 96 GB HBM3 memory

A100

- JUWELS Booster: 936 nodes, 4×A100 each
- AMD EPYC 7402 CPU (48 cores)
- A100 GPU (108 multiprocessors); 40 GB HBM2 memory

Applications

MPTRAC (neither compute nor memory bound)

- Massive Parallel TRAjectory Calculations (MPTRAC) is a Lagrangian particle dispersion model for large-scale atmospheric transport simulations.
- It is used to simulate the transport and dispersion of 100 million particles using ERA5 meteorological data.

SOMA (memory bound)

- SOMA is an HPC Monte-Carlo code for soft coarse-grained polymer simulations
- It is used to investigate rate of phase transformations in self-assembling systems

Methodology

Easy-to-use, comparable energy evaluations are still not standard. The methodology used here combines several tools:

exaCB This template-based CI/CD framework automates jobs launch and data recording through GitLab

jpwr A non-invasive wrapper tool to measure power on Grace-Hopper and A100 systems via vendor functionality, including simple post-processing

Range Markers Functionality included in exaCB for measurement of relevant benchmark portions, for example exlcuding non-representative I/O overhead, or similar

NVML The GPU clock rates are set through the *nvidia-smi* CLI to the NVML library using Slurm functionality

Post-Processing Tooling to post-process recorded power data to generate accurate energy measurements

Range-Based Measurement Example

- Plot of power consumption over time for application
- Start/stop markers provided by application (black lines in plot) indicate start/end of relevant part of benchmark
- Exclusion of start-up costs and other overhead, which are artifacts of benchmark formulation of application and have negligible impact in *real* execution
- Energy measured through integration of power in range

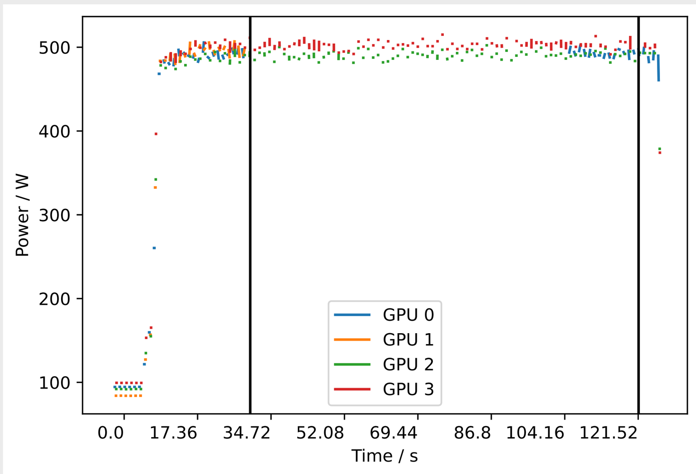

Figure: Ranged-based Measurement Example

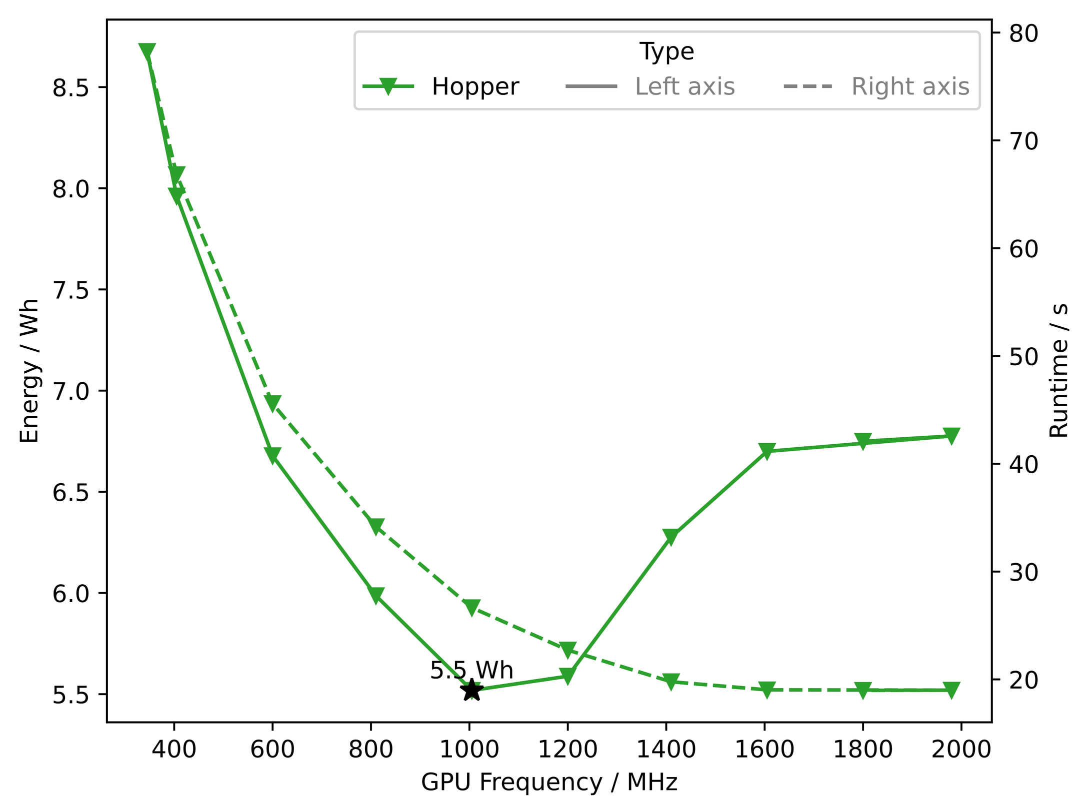

Figure: **HPL** E2S (left axis) and runtime (right axis).

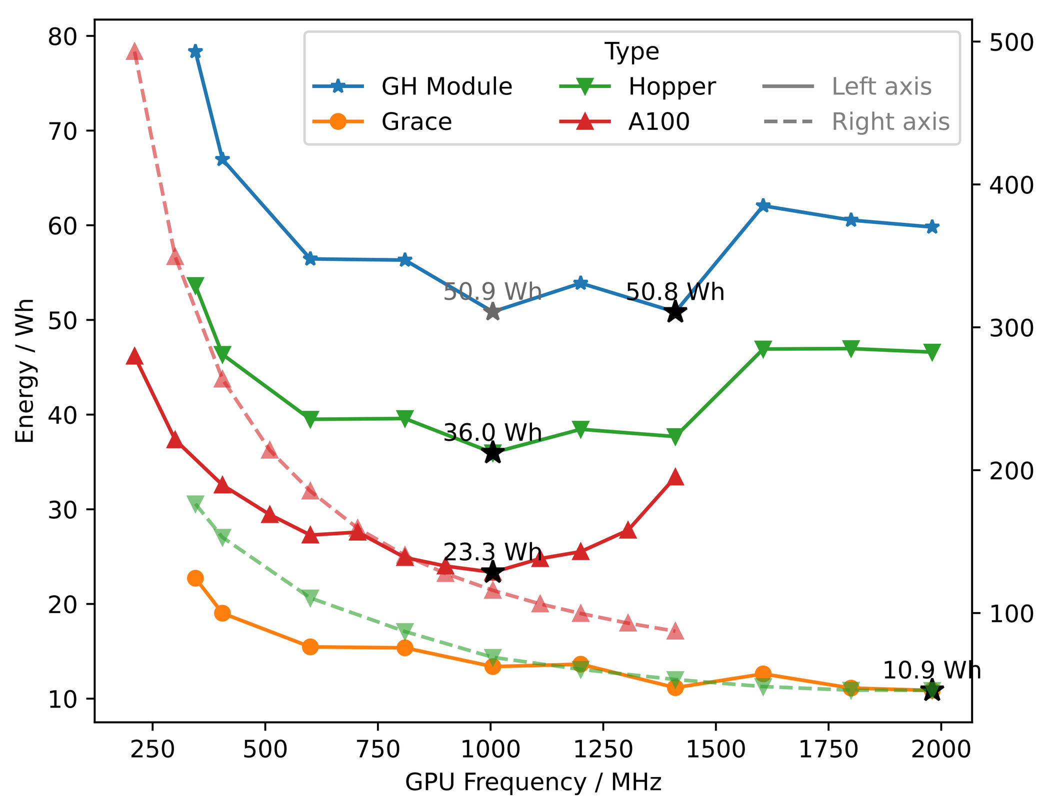

Figure: **MPTRAC** E2S (left axis) and runtime (right axis).

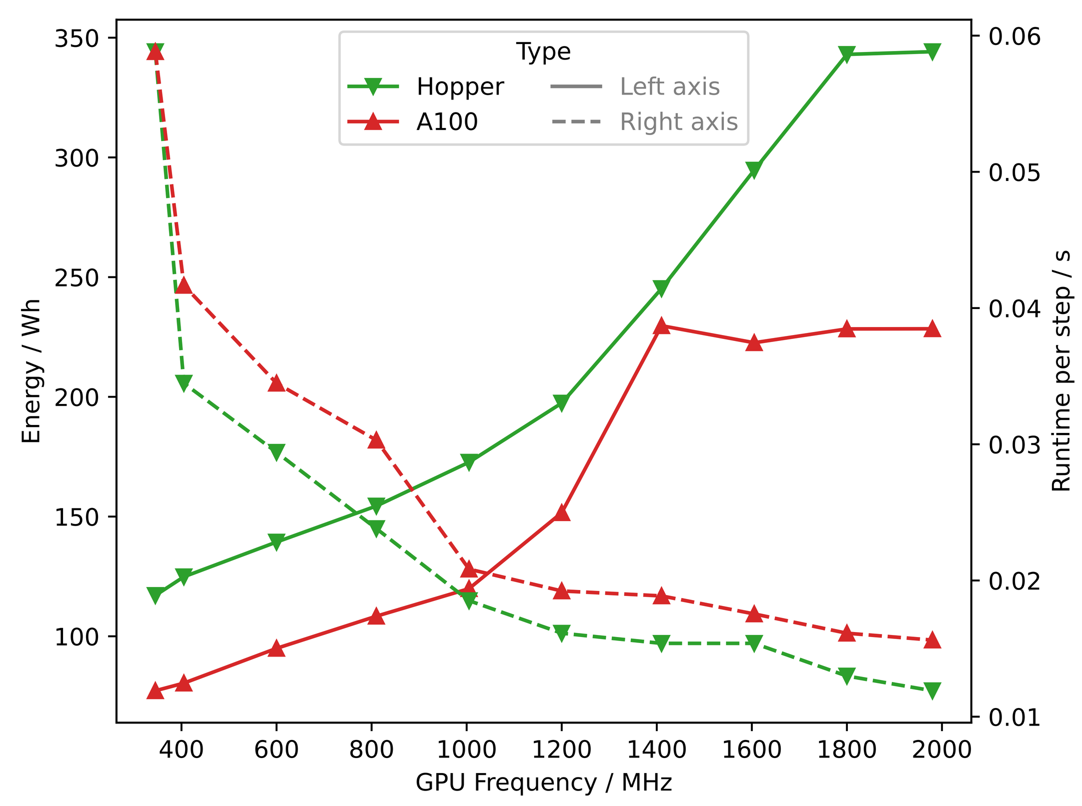

Figure: **SOMA** E2S (left axis) and runtime (right axis).

Results

- E2S mostly affected by used GPU frequency (lower frequency → lower GPU power consumption) and resulting Time-to-Solution T2S (generally higher frequency → lower runtime → lower energy consumption); search for *sweet-spot*
- Every GPU has additional constant idle power consumption not affected by used frequency
- Different behavior for application profiles: Effect of reducing T2S by higher GPU frequencies is different for compute-bound (large effect) and memory-bound (small/no effect) applications

⇒ Non-memory-bound examples (HPL, MPTRAC) show parabolic energy consumption profile (Hopper, A100)

- Memory-bound application example (SOMA) runs most energy efficient with very small GPU frequencies, as the unchanged memory clock is the dominant limitation

⇒ As a result, memory-bound applications show especially large energy saving potential

- Inclusion of CPU energy measurement (Grace) and full module (GH Module; includes CPU, GPU, memory, regulators) changes results slightly, but not systematically
- Expectedly, Hopper has better T2S, but A100 has better E2S for applications shown; investigation with larger workloads needed

Future Work

- Further automation for data recording and post-processing; accurate energy profile estimation based on application characteristics
- Analysis of more applications through framework and different GPUs
- Study of power capping-mechanisms (JUPITER nodes capped to 680 W); power distributed dynamically between Grace and Hopper
- Review relevance of E2S, Time-to-Solution, and energy efficiency with different stakeholders

Acknowledgements

JUPITER is funded by EuroHPC JU and BMBF and MKW; it is hosted by Forschungszentrum Jülich. Compute time on the system is awarded through EuroHPC JU and GCS. The work is part of JUREAP, the JUPITER Research and Early Access Program.

References

SOMA Schneider and Müller, Multi-architecture Monte-Carlo (MC) simulation of soft coarse-grained polymeric materials: SOft coarse grained Monte-Carlo Acceleration (SOMA). Comput. Phys. Commun. 235, 463 (2019)

MPTRAC Hoffmann, Lars, et al. "Accelerating Lagrangian transport simulations on graphics processing units: performance optimizations of MPTRAC v2. 6." EGUsphere 2024 (2024): 1-26.

HPL Dongarra, Jack J., et al. "Introduction to the HPC Challenge benchmark suite." Mar. 2005