Image source: DALL·E AI

Energy-efficiency in high performance computing

from lattice QCD to large language models

Antonin Portelli — 27/11/2024

UK HEP Forum 2024

erc
CENTRE FOR THEORETICAL PHYSICS · Higgs
THE UNIVERSITY of EDINBURGH
UKRI Science and Technology Facilities Council
DiRAC

- General considerations
- Summer 2022 DiRAC Grid study
- Beyond the Grid library
- GPT large language model training

General considerations

World supercomputing energy cost

3.8 TWh/year

(GREEN500 Nov 2024, 199 systems)

- This is

  1.3% of the UK yearly energy consumption [EIA]

  2.9x CERN yearly energy consumption [CERN]

  1.5 Mt/year of carbon emission (assuming 400 g/kWh)

Software & energy efficiency

- High-performance computing (HPC) has a small energy footprint compared to e.g. manufacturing

- However
  - Integrating energy-efficiency in HPC software design is rarely a priority over raw performances
  - The AI boom will increase dramatically the volume of active HPC hardware in the world

Energy-efficiency and software design

- The energy efficiency of a software is measured as **the amount of work done per unit of energy consumed**

- "Amount of work" is **heavily context-dependent**
  Typically a number of operations (e.g. Flop) for HPC

- Energy-efficiency is influenced by both the **hardware used** and the **software implementation of a given algorithm**

Ethical aspects

- "Amount of work" does not take into account the usefulness of the work
- The Bitcoin blockchain has consumed **140.48 TWh** so far in 2024 [CCAF, 25/11/2024]
- Online ads rendering estimated to be **1.8 — 91 TWh/year** [arXiv:2211.00071]
- For scientific HPC work: *is an expensive scientific computation impactful? redundant? appropriately accurate?*

Improving energy-efficiency: goals

- Understand the energy **footprint of HPC calculations**
- Understand how to improve energy efficiency to help **reaching net zero computing targets**
- Understand how to mitigate the **impact of surging energy prices on scientific outputs**
- **Bottom-up approach**: start from **domain-specific studies**. Energy-efficiency is **domain-dependent**

Summer 2022 DiRAC study

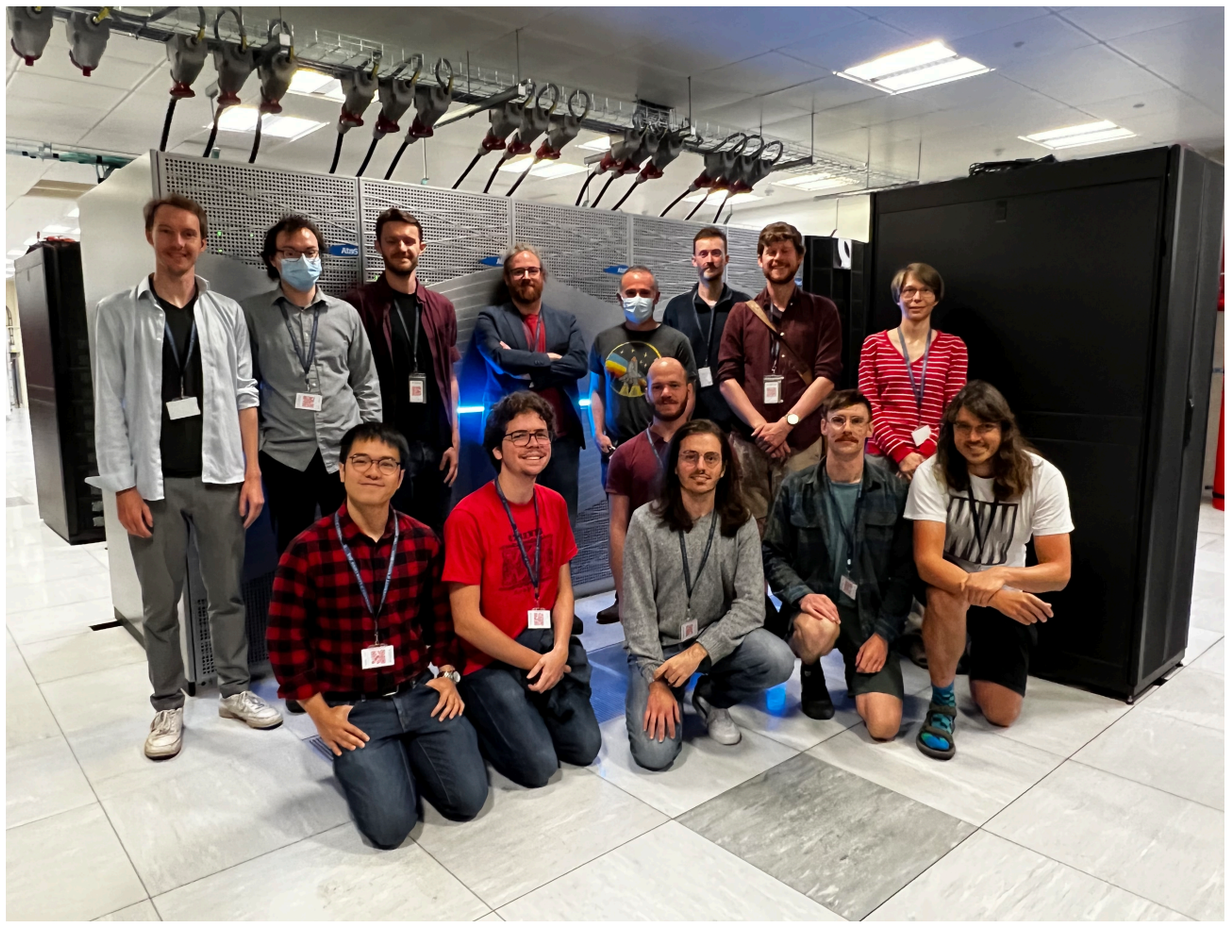
Report and data

- Report commissioned by UK STFC DiRAC
  https://doi.org/10.5281/zenodo.7057318

- Report data and running environment
  https://doi.org/10.5281/zenodo.7057644

- Everything available under CC-BY-NC 4.0

STFC DiRAC Tursa supercomputer



Edinburgh lattice team & Tursa, July 2022

UKRI Science and Technology Facilities Council

DiRAC

epcc

- Eviden BullSequana XH2000
- 724 NVIDIA A100 GPUs
- 4 x HDR200 NICs / node

The Grid library

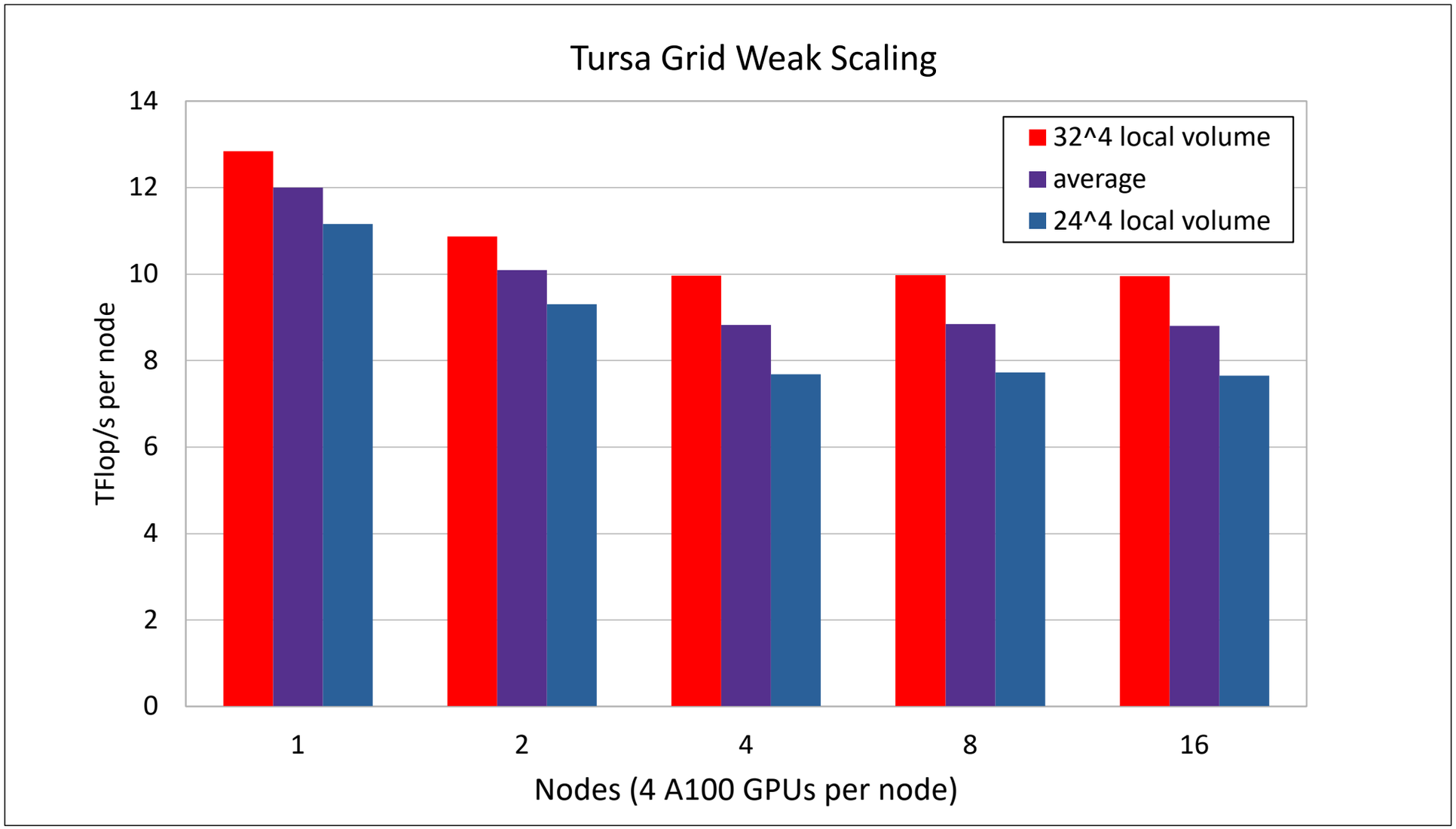
- C++14 data parallel C++ mathematical object library, **targeted at lattice QCD**

- **Cross-platform** with architecture-specific optimisations
  (x86, ARM, NVIDIA & AMD GPUs, …)

- Optimally use MPI, OpenMP and SIMD/SIMT parallelism under the hood

- Free and open-source (GPLv2)
  https://github.com/paboyle/Grid — https://doi.org/10.22323/1.251.0023

Grid performances on Tursa



Benchmark setup

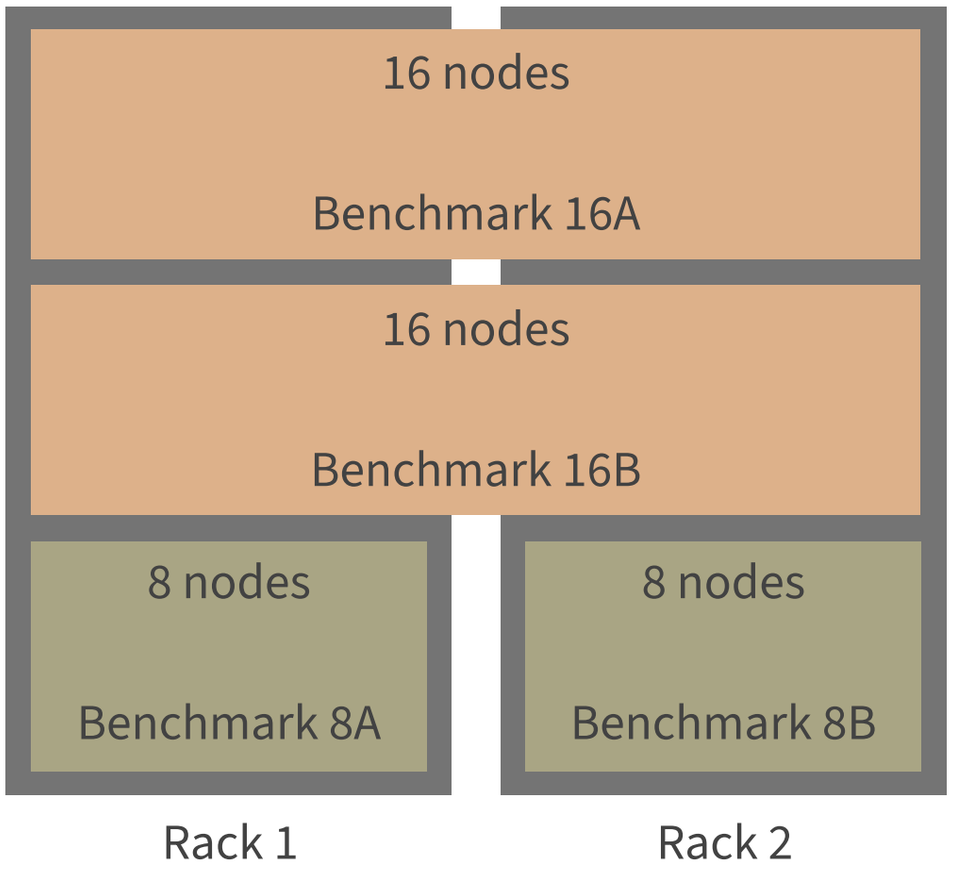
- Grid benchmark `Benchmark_dwf_fp32`, based on the single-precision domain-wall fermion sparse matrix

- 2 full XH2000 racks
  (48 nodes, 192 A100 GPUs)

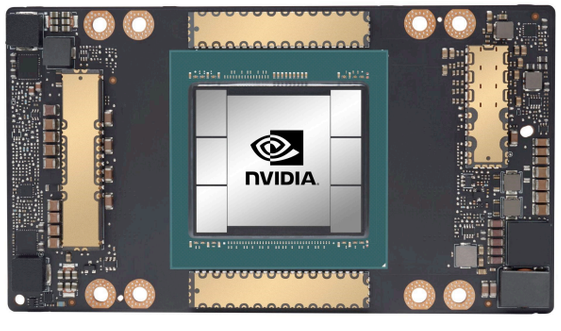
- 2x16 nodes + 2x8 nodes

- Layout based on optimal communication topology

- Constant local problem size

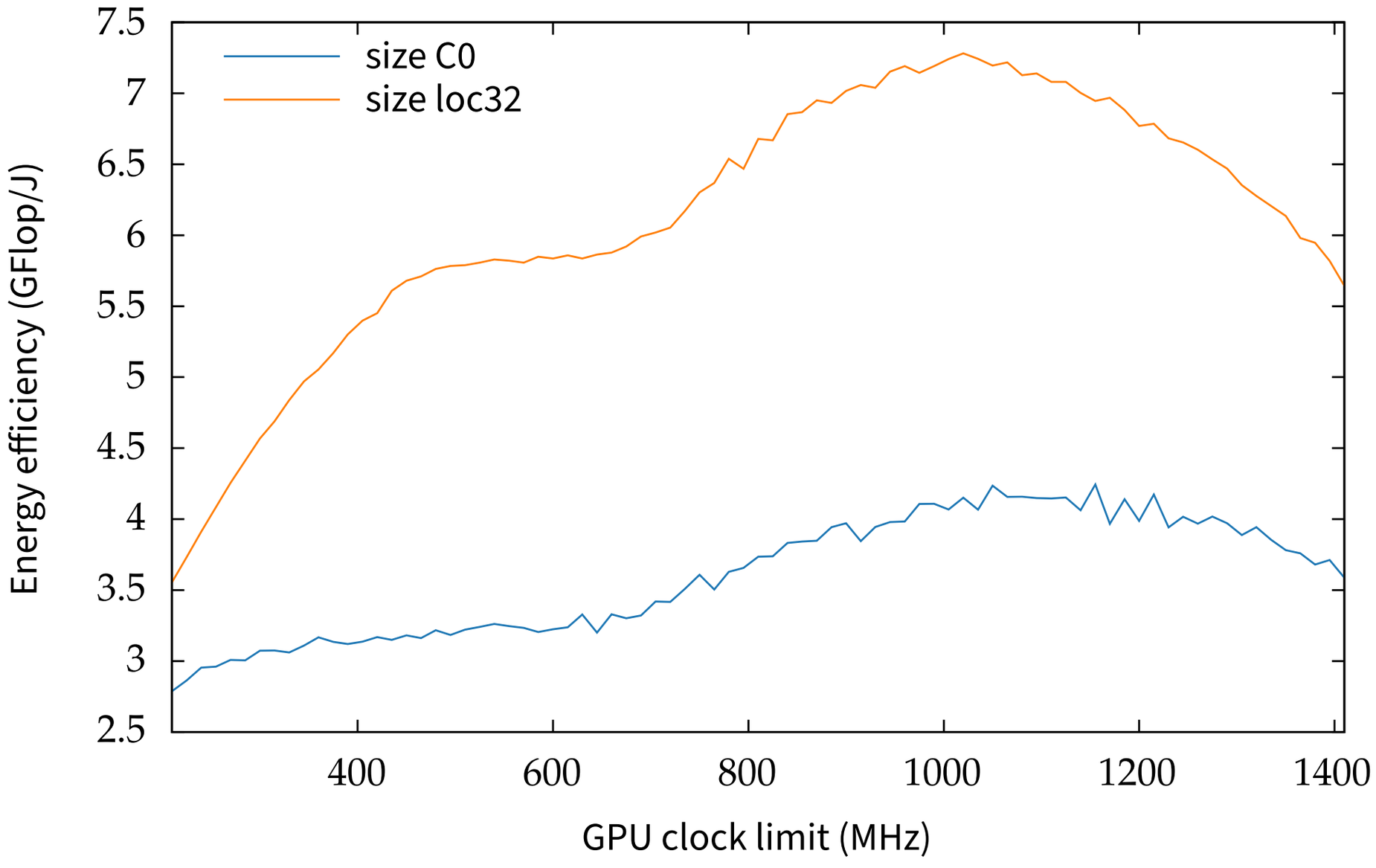


Power control and monitoring



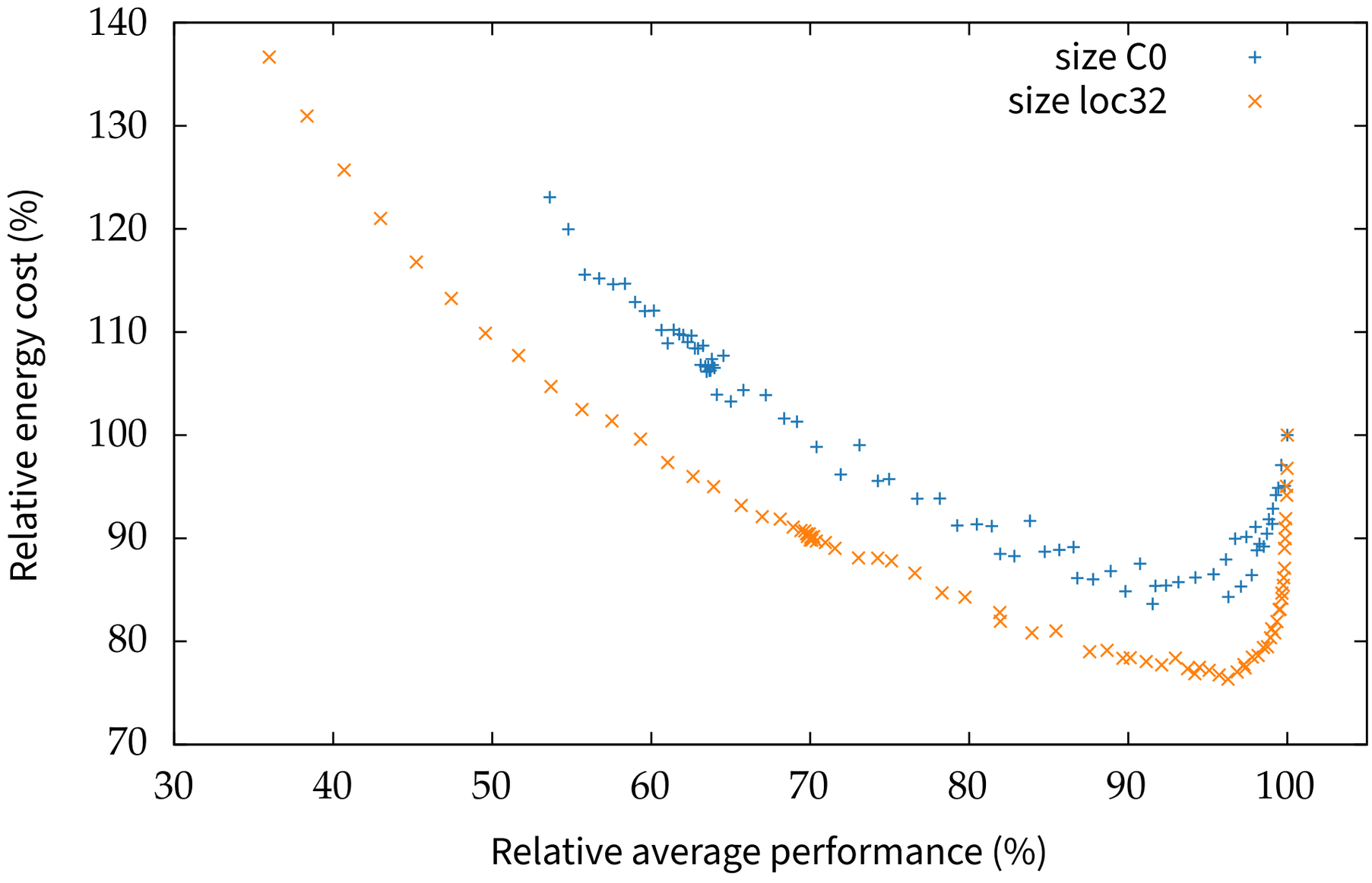
- Power controlled through **under-clocking of GPUs**
- Clock limit from **210 MHz to 1410 MHz** (increment 15 MHz)
- Default setting: maximum frequency 1410 MHz
- **Power monitoring**
  1) per GPU (NVIDIA SMI) 2) per rack (PDU through SNMP)

Energy efficiency vs GPU clock



Default setting not energy-optimal!

Energy vs performance landscape



Outcome

- Tursa GPUs set to **1050 MHz by default** since Dec 22
- Monitoring show a 11% decrease in energy consumption
- Users reported no significant changes in throughput
- **Estimated energy savings are ~226 MWh** (today)

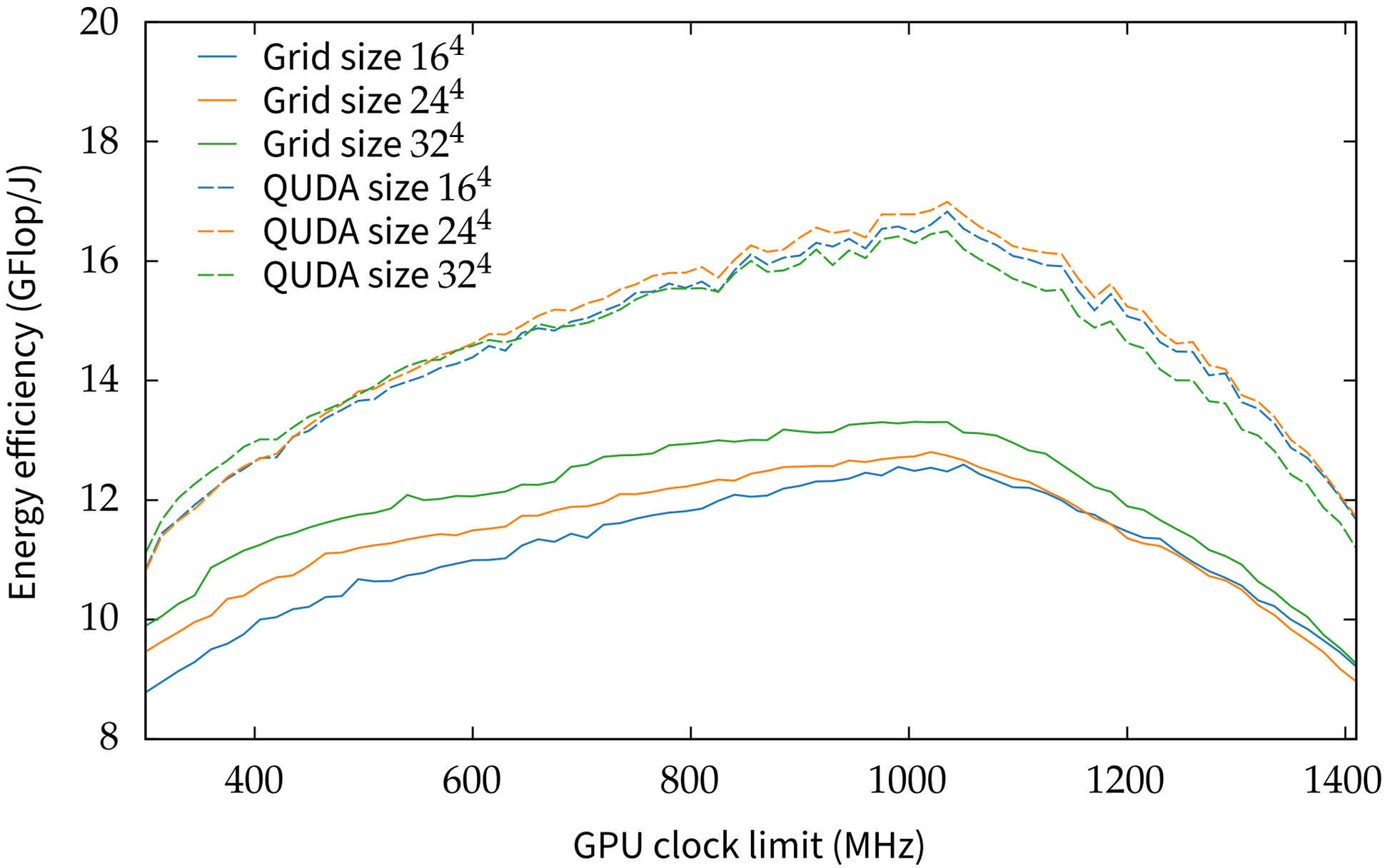
Beyond the Grid library

In collaboration with Simon Bürger (Edinburgh RSE)

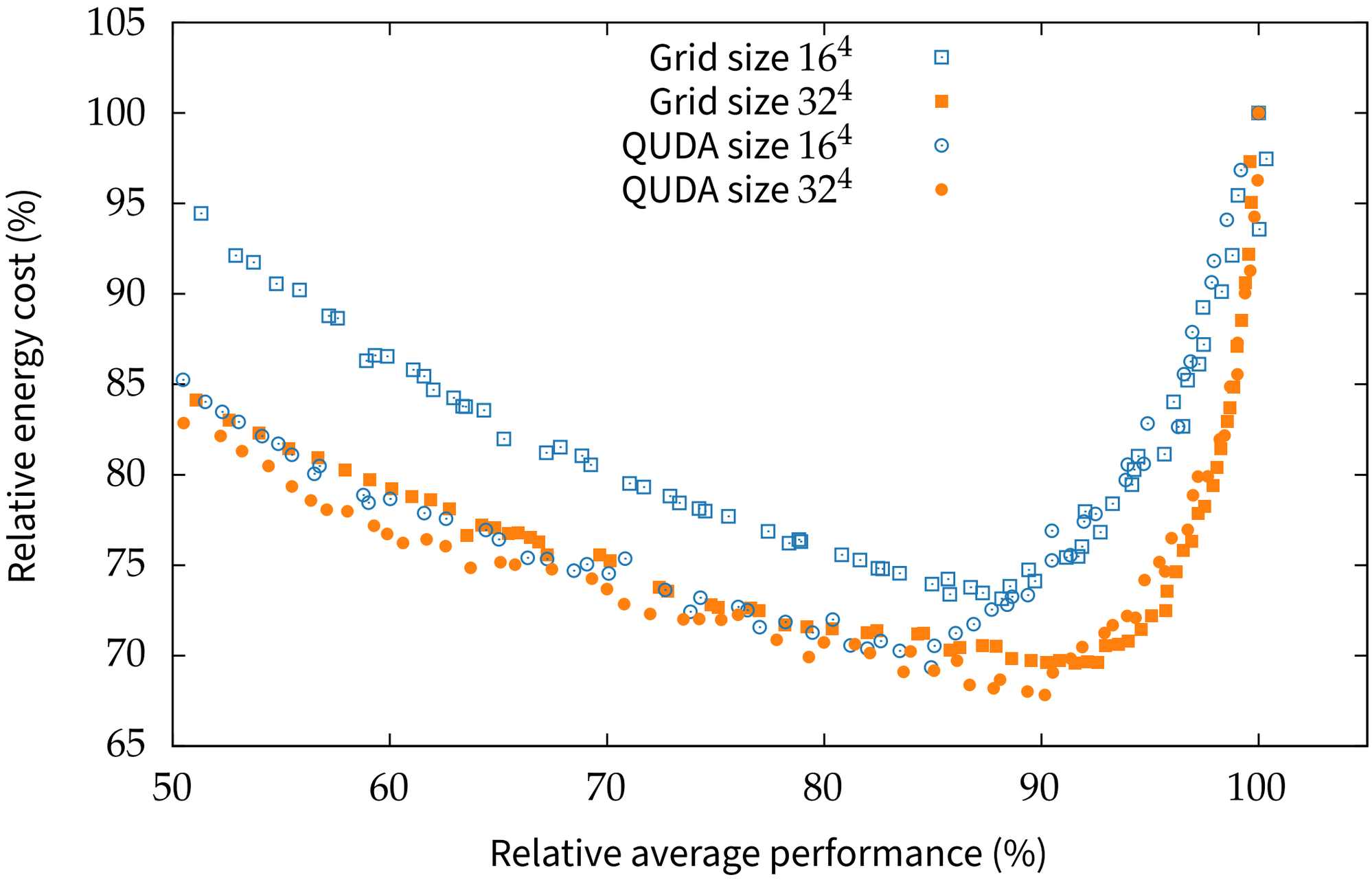
QUDA benchmark

- QUDA is one of the main library for lattice QCD on GPUs
- Open-source, developed and supported by NVIDIA
  https://github.com/lattice/quda
- Here: **custom QUDA benchmark**, matching Grid benchmark flop count and problem sizes
- Still using A100 GPUs on Tursa
- **Single node, GPU power only**

Energy efficiency, QUDA vs Grid



Energy vs performance landscape, QUDA vs Grid



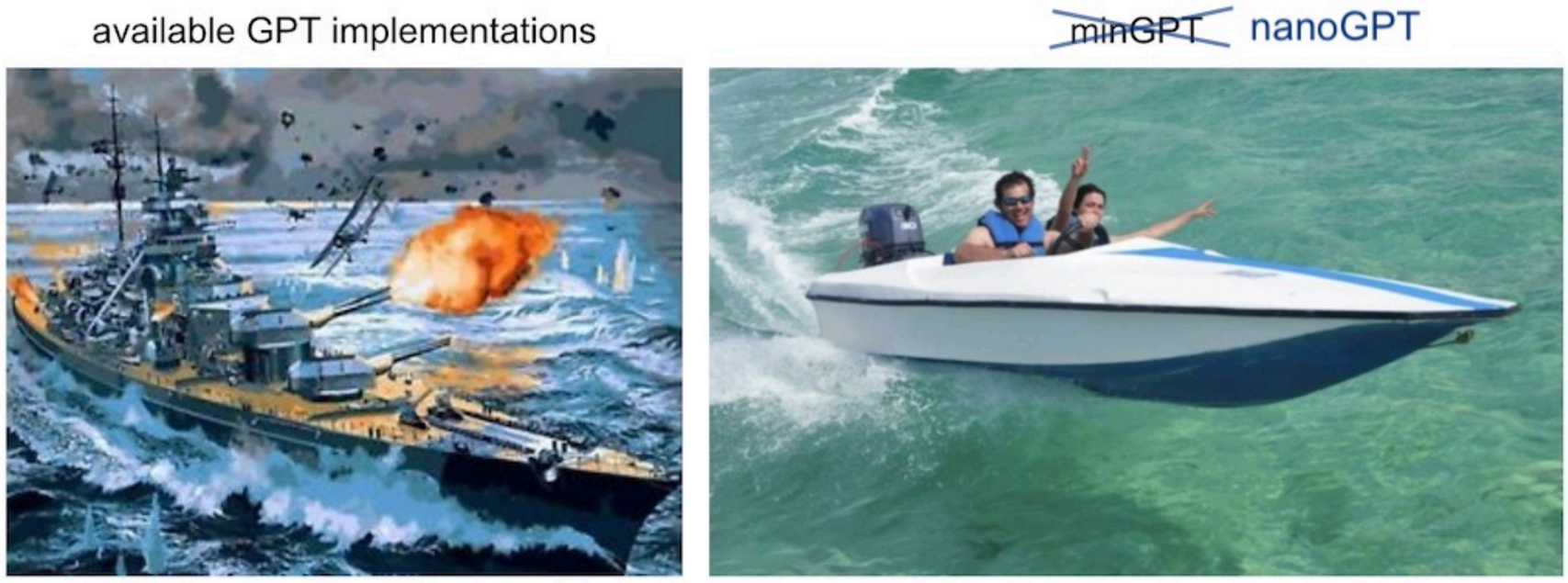
Conclusion

- QUDA and Grid share an **energy-optimal point at 1 GHz**
- QUDA significantly faster than Grid for small sizes, more similar for large sizes
- Different energy profiles for small sizes, almost identical at large sizes
- To be extended on multiple nodes!

GPT language model training

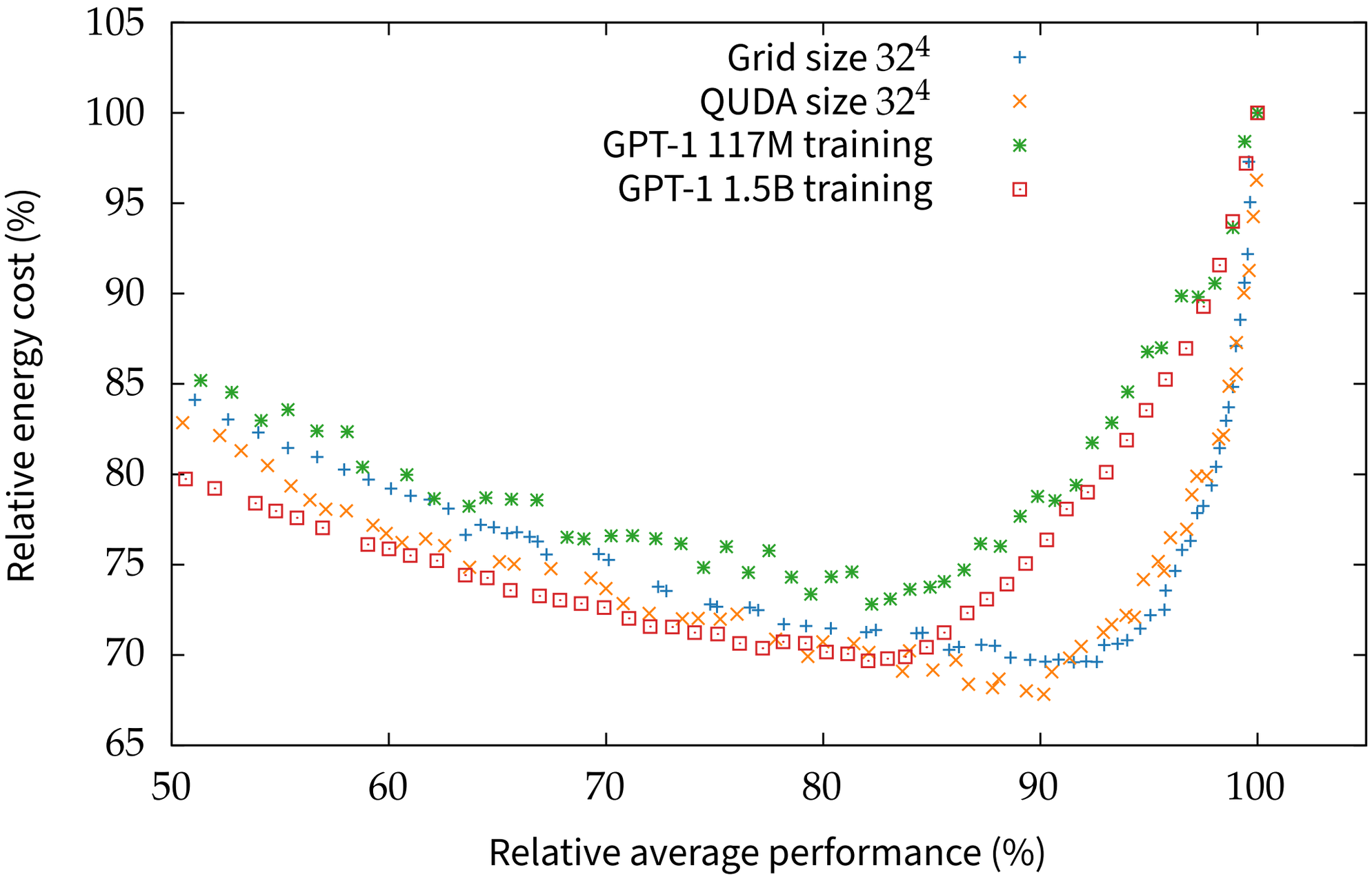
In collaboration with Fabian Joswig (DeepL, formerly Edinburgh)

Setup



- nanoGPT: open-source reproduction of GPT-2
- OpenWebText2 training set (whole of Reddit 2005-2020)
- Setup to reproduce GPT-1 (117 M) and GPT-2 (1.5 B)
- Single node 4x GPUs, ~700 TFlop/s for GPT-2 🤯

Results

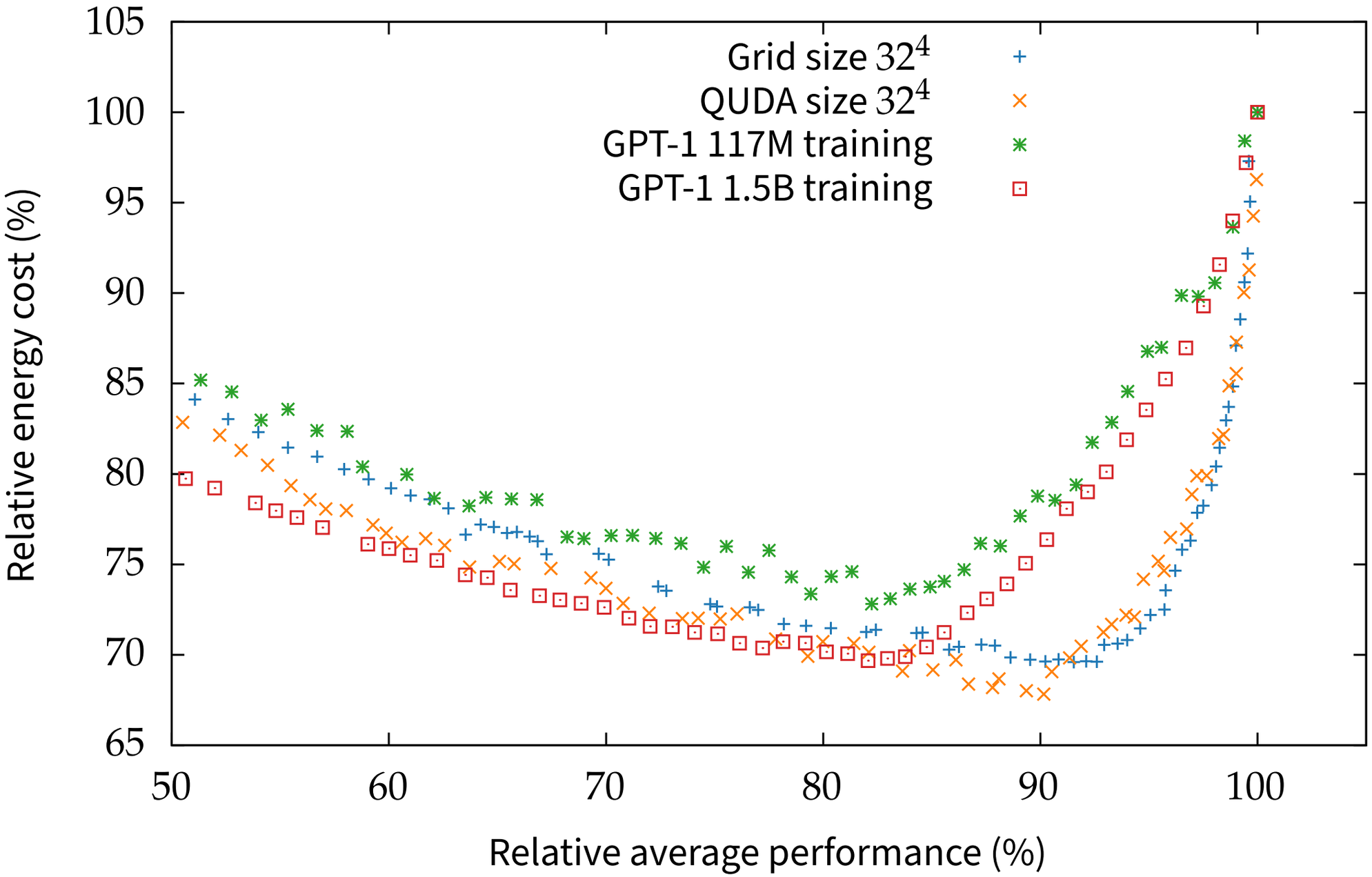

Grid size 32^4
QUDA size 32^4
GPT-1 117M training
GPT-1 1.5B training
Relative energy cost (%)
105
100
95
90
85
80
75
70
65
50
60
70
80
90
100
Relative average performance (%)

Conclusions

- A100 frequencies around **1 GHz** generally lead to 20-30% more energy efficient computations (GPUs only)
- Energy saving potentially reduced by non-GPU elements
- Impact on floating-point performances within
  10% (lattice) & 20% (LLM training)
- **Lower default frequencies** recommended on GPU clusters

Perspectives

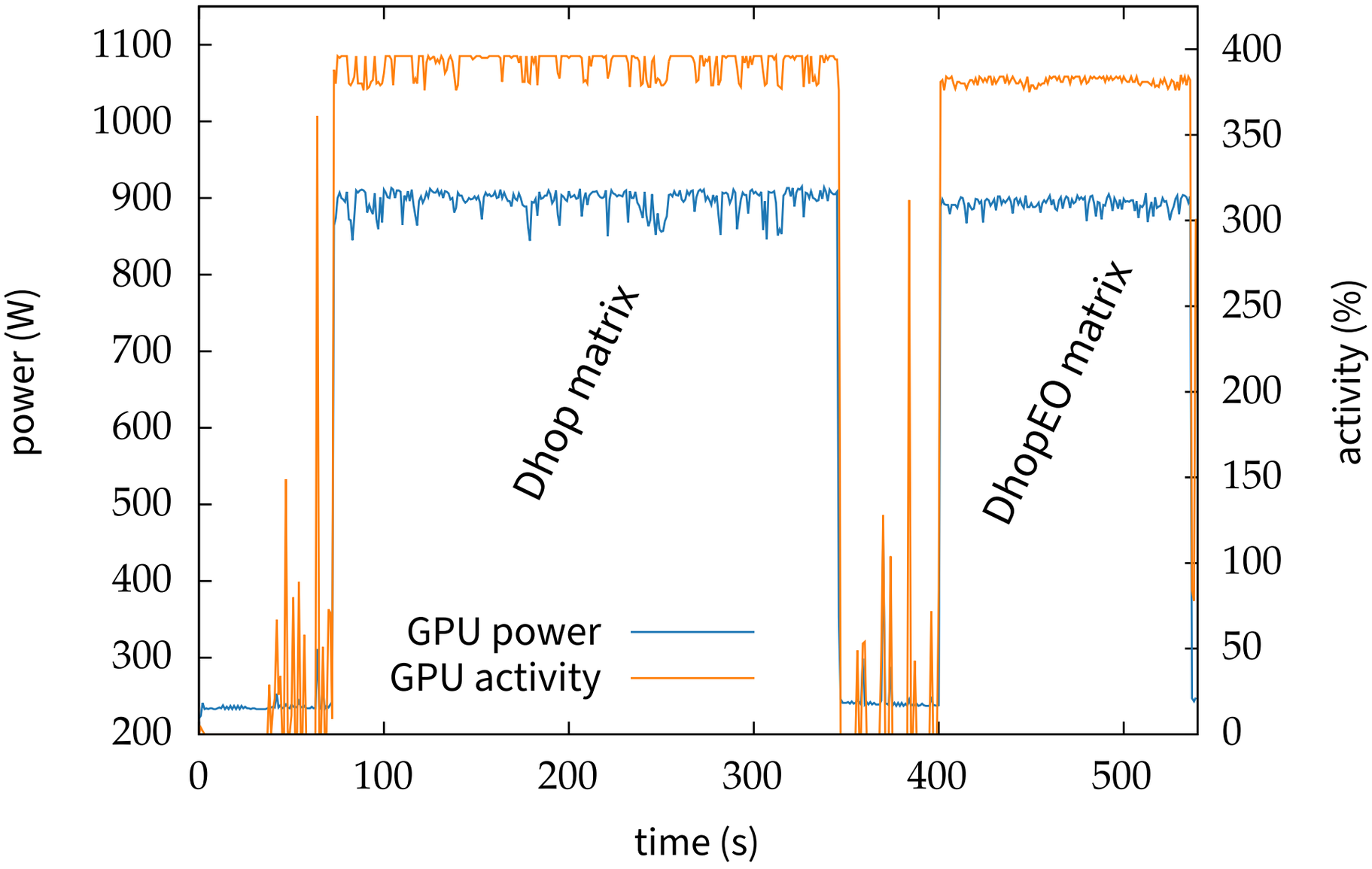
- Should energy-efficiency become a standard performance figure in benchmarks?
- Should energy-efficiency become a stronger constraint in supercomputer procurement?
- Should energy-efficiency scaling and benchmarks become part of peer-reviewing processes for resource allocation?



Thank you!

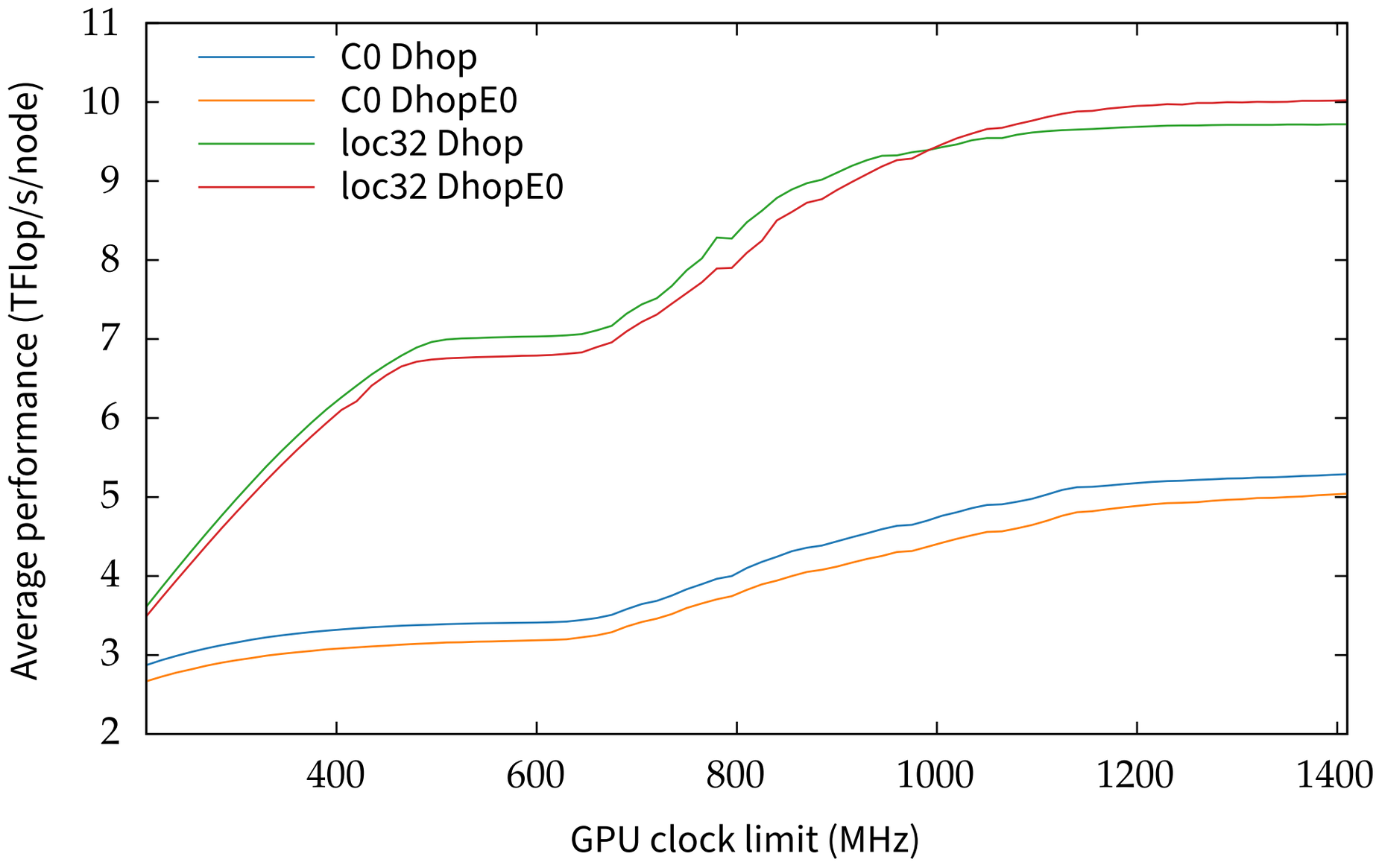
This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreements No 757646 & 813942.

Raw data: GPU activity & power

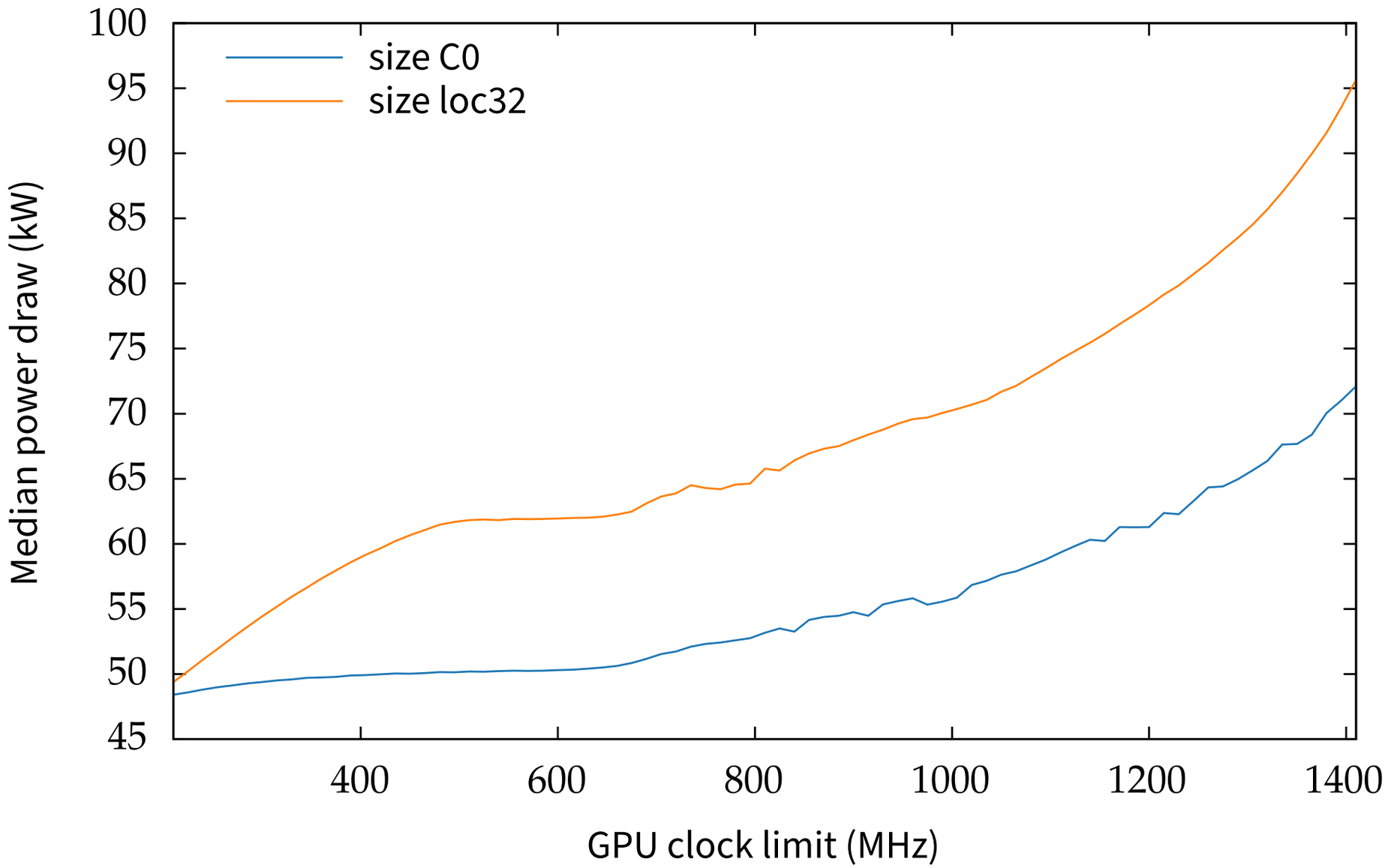


Master node — GPU clock limit 1020 MHz — size loc32

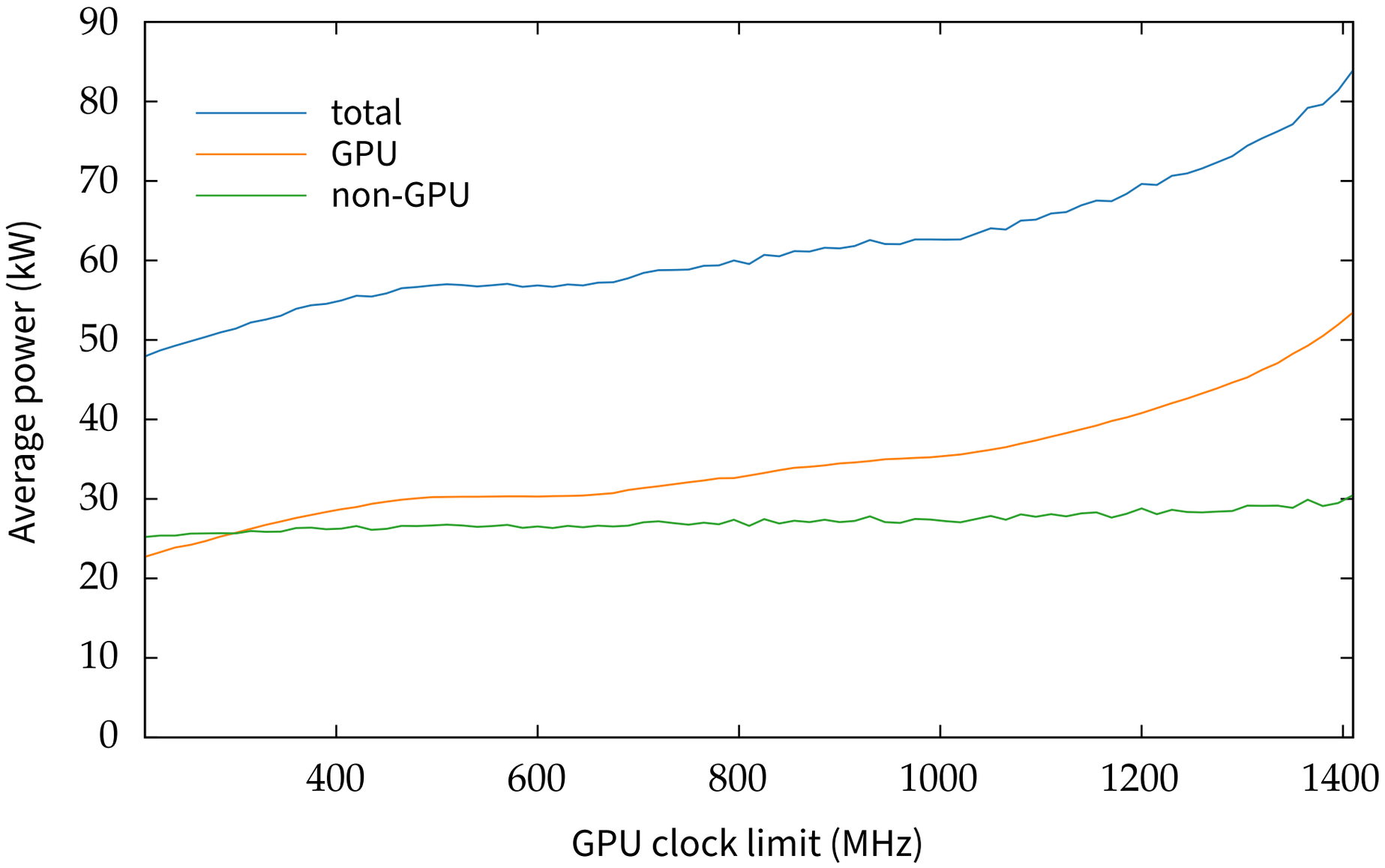
Performances vs GPU clock limit



Power draw vs clock limit



Power draw breakdown



Non-GPU almost constant & consistent with idle power draw