Energy consumption characterization of subnuclear physics computing workloads

Francesco Minarini, Marco Lorusso, University of Bologna, INFN, Italy

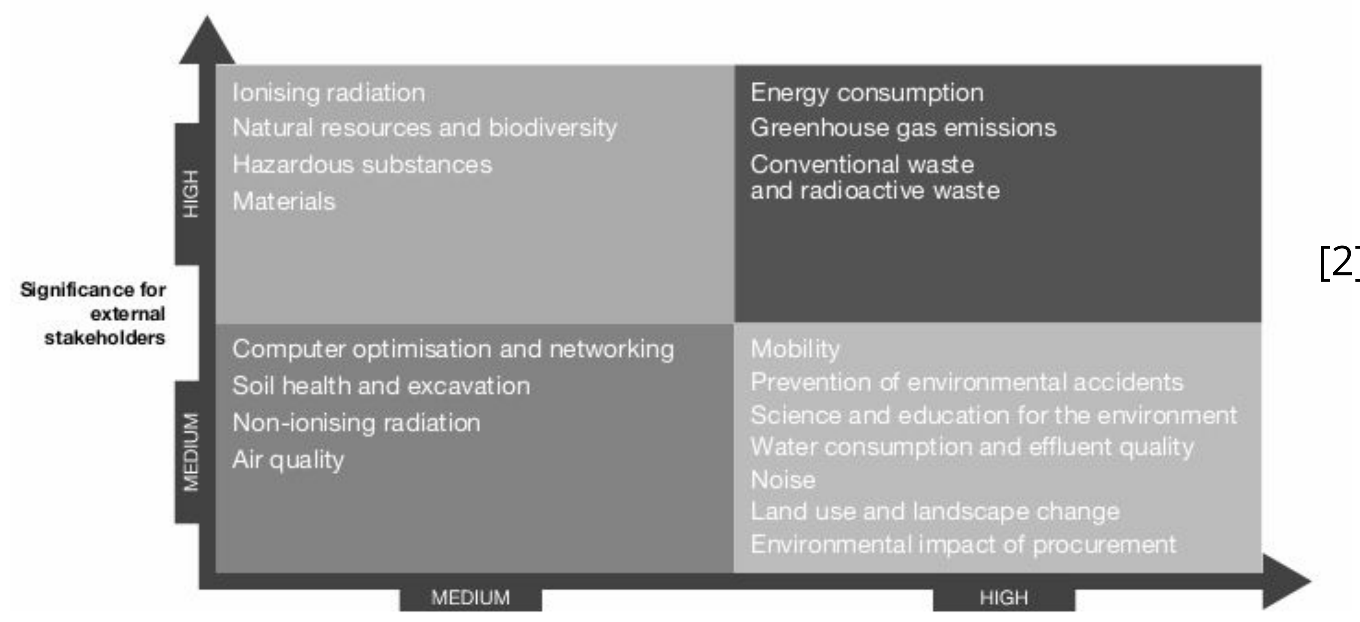
francesco.minarini3@unibo.it, marco.lorusso11@unibo.it

Motivation

Climate change poses a grand challenge to the scientific computing community:

“Given the rapidly growing energy consumption of the ICT sector [1] and the eagerness of popular algorithms, how can computational scientists keep pursuing their research interests without making their work too unsustainable? How can we curb the footprint of our scientific activities?”

high interest from CERN!



[2]

The footprint model

Energy consumption estimation model in [3] was adopted:

$$E_{computing} = T_{compute} \cdot (n_{cores} \cdot P_{cores} \cdot c_{usage} + P_{memory} \cdot m_{usage}) \cdot PUE \quad \textbf{(Eq.1)}$$

$T_{compute}$ = Time-to-solution (h)

n_{cores} = Number of requested cores

P_{cores} = Power draw in W of each core (derived from CPU TDP)

c_{usage} = Core usage expressed as a percentage (i.e. 1 = full usage)

P_{memory} = Power draw in W/GB of RAM memory (derived from RAM TDP)

m_{usage} = GB of memory allocated throughout the computation

And optionally the carbon footprint can be calculated through:

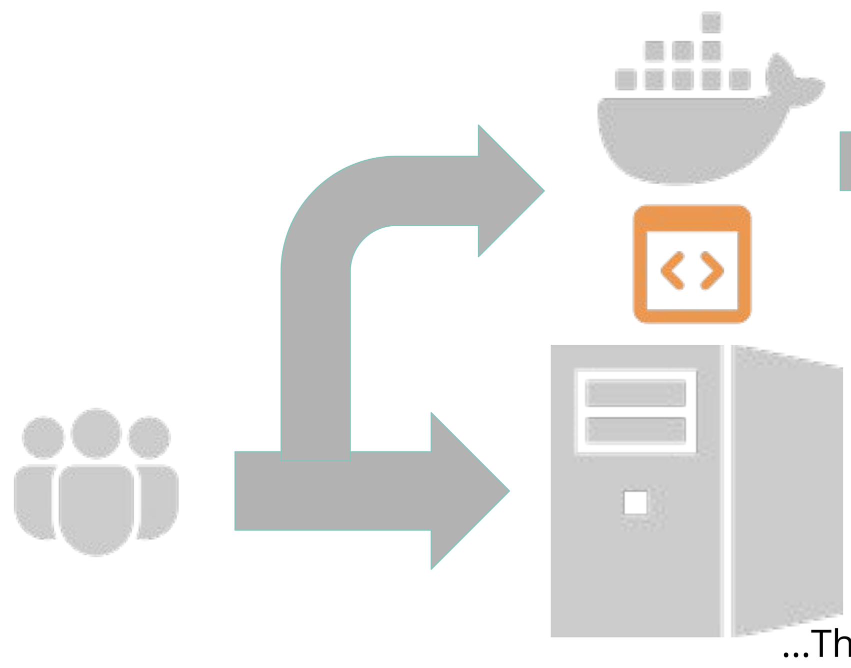
$$CF = E_{computing} \cdot CI$$

Where CI is the “carbon intensity” (gCO2e/kWh), representing the country-wise footprint of the electricity mixture [4].

The tool

In */proc* we can find system information about resource usage. Process-level info is in */proc/<PID>/...* *PID* can be retrieved with bash commands

This data + hardware technical data –> populate (Eq.1) = Energy consumption estimation!



The output of the containerized C++ SW [5]:

1) Workload Power Draw *(W)*
2) Workload Exec time *(h)*
3) Workload Carbon footprint *(gCO2e)*

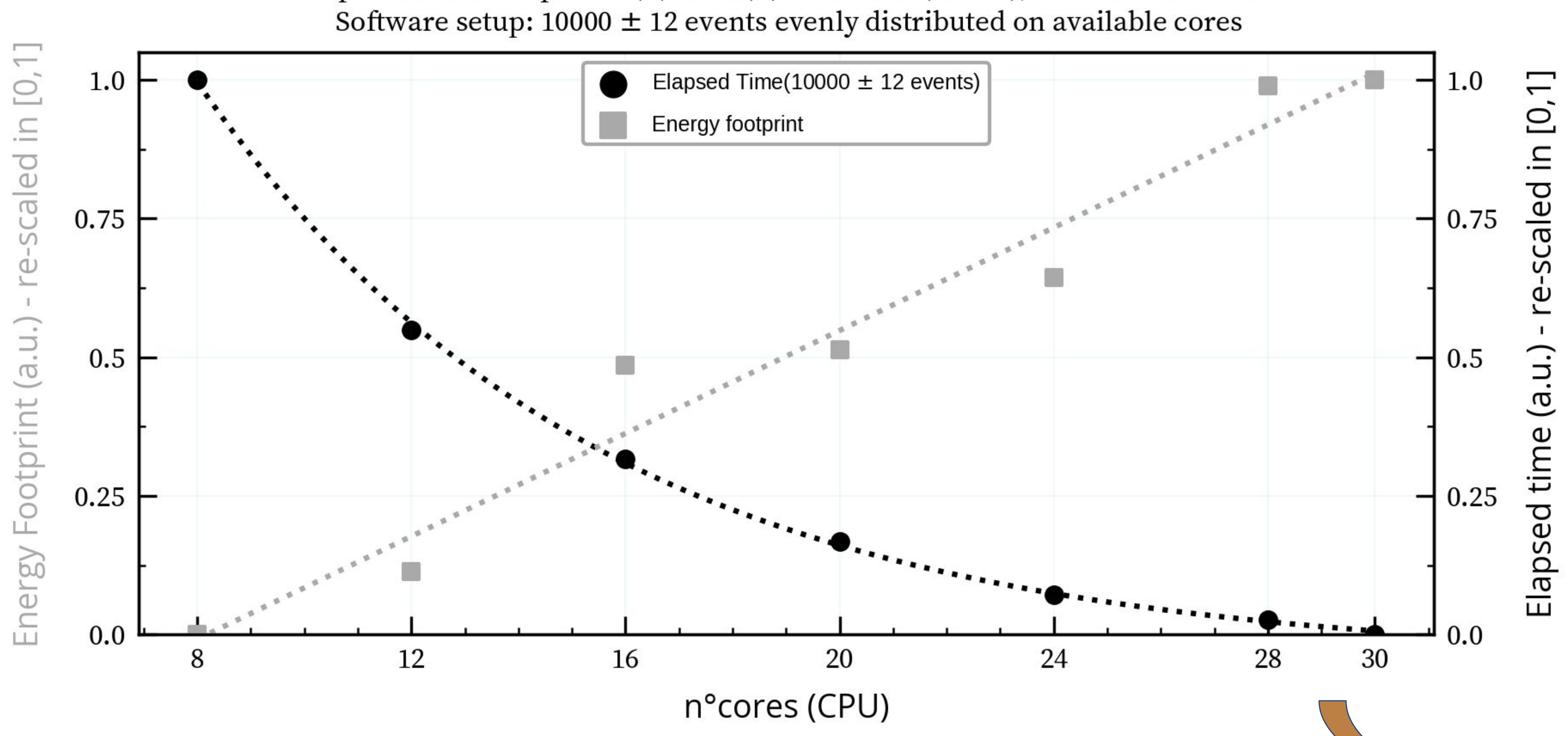
...The workload executes on the node gracefully

Results

To explore the energy consumption of physics computing, benchmark physics payloads described in [6] were used.

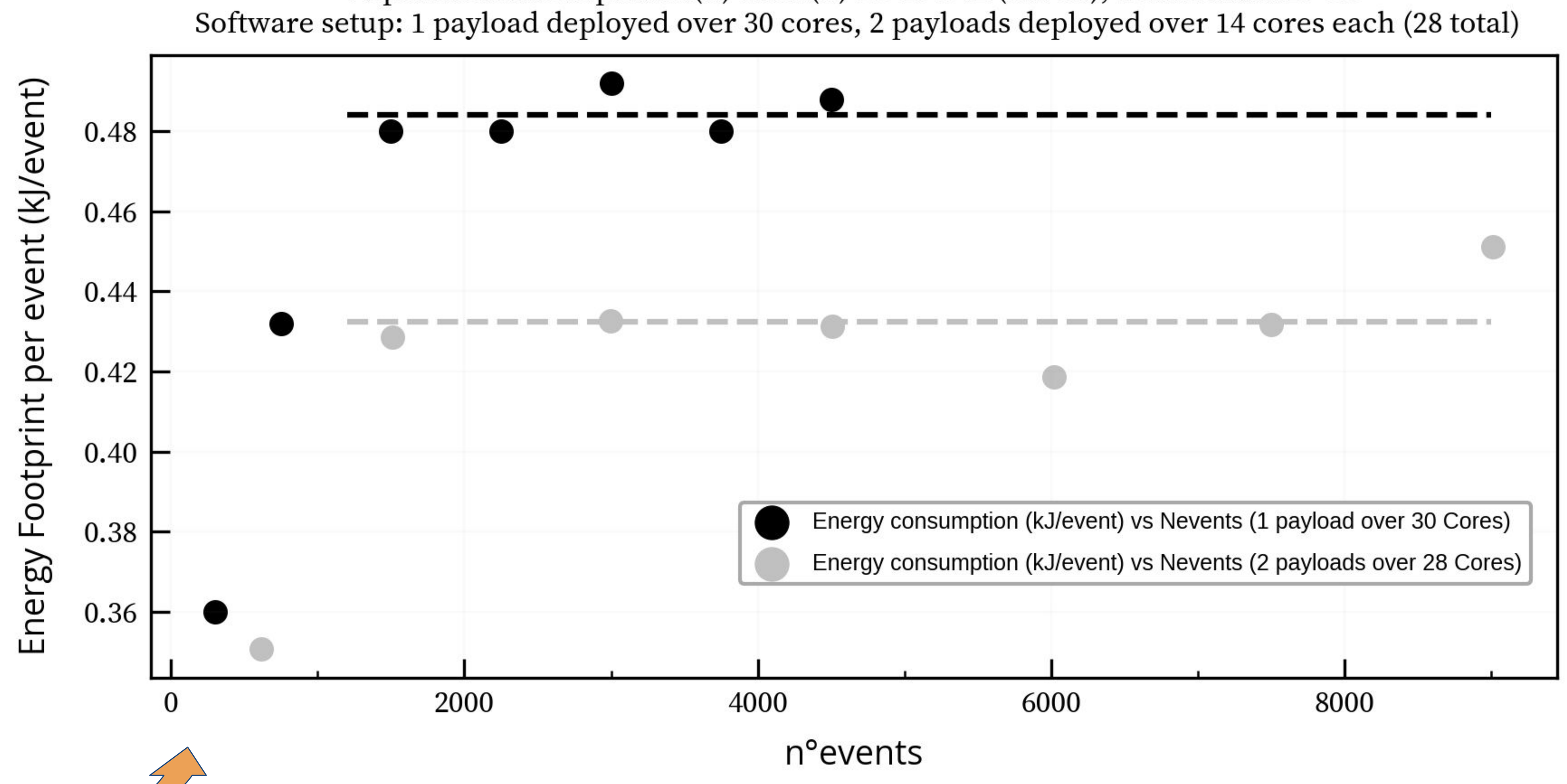
Ideal Core working-point for a CMSSW GENSIM computing payload (multi-threaded GEANT application)

Experimental setup: Intel(R) Xeon(R) E5-2640v2 (HT on), Total RAM:128 GB
Software setup: 10000 ± 12 events evenly distributed on available cores



Footprint of CMSSW GENSIM computing payload (multi-threaded GEANT application)

Experimental setup: Intel(R) Xeon(R) E5-2640v2 (HT on), Total RAM:128 GB
Software setup: 1 payload deployed over 30 cores, 2 payloads deployed over 14 cores each (28 total)



Conclusions

- The tool provides a node-level energy consumption estimation of computing payloads (in this case physics).
- This helps **raising energy awareness** about computing infrastructure usage (we get more throughput if we optimize submission including an energy perspective)

References

[1] Freitag et al. (10.1016/j.patter.2021.100340)

[2] CERN (10.25325/CERN-Environment-2023-003)

[3] Lannelongue et al. (10.1002/advs.202100707)

[4] https://www.nowtricity.com/country/italy/

[5] https://github.com/fminarini/KIG

[6] Valassi et al. (10.48550/ARXIV.2004.01609)