Test e benchmarks

Outline

- A few notes on testing
- Preliminary tests
- Computed Tomography
- Astrophysics
- Molecular Dynamics
- HS

Testing: performance

- To avoid dynamic CPU frequency scaling and to maximize CPU performance: use *cpufrequtils*, e.g. cpufreq-set --governor performance (https://wiki.debian.org/HowTo/CpuFrequencyScaling)

- To controll GPU performance (Jetson): set a supported rate with echo 852000000 > /sys/kernel/debug/clock/override.gbus/rate (https://devtalk.nvidia.com/default/topic/747641/anyone-know-how-to-monitor-the-gpu-mhz-/)

- CPUs become online only when the scheduler judges there's enough work to do. To manually force the 4 main CPU cores (Jetson) to run: (http://elinux.org/Jetson/Performance)

Testing: power

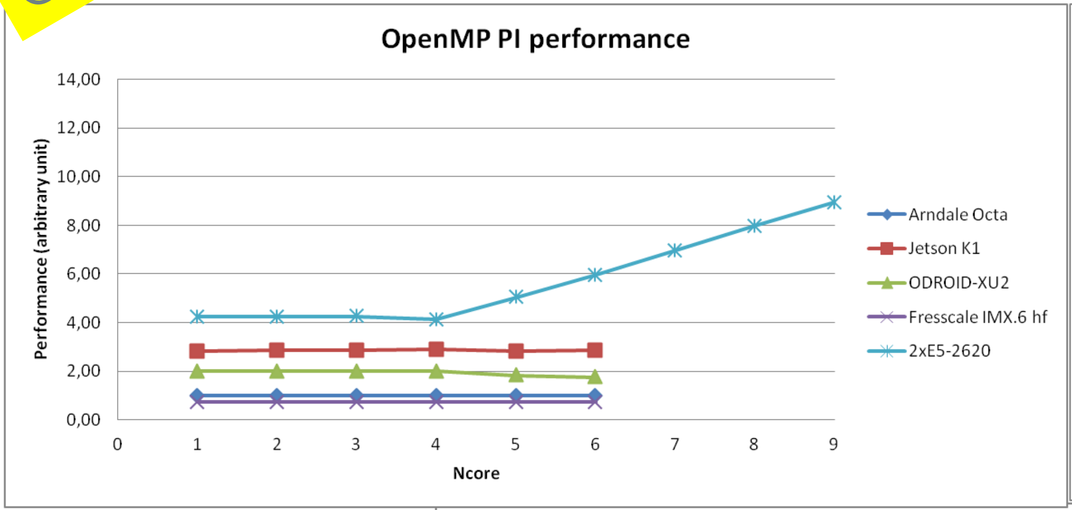
- We compute power consumption by measuring the flow of electrical current and knowing the voltage.

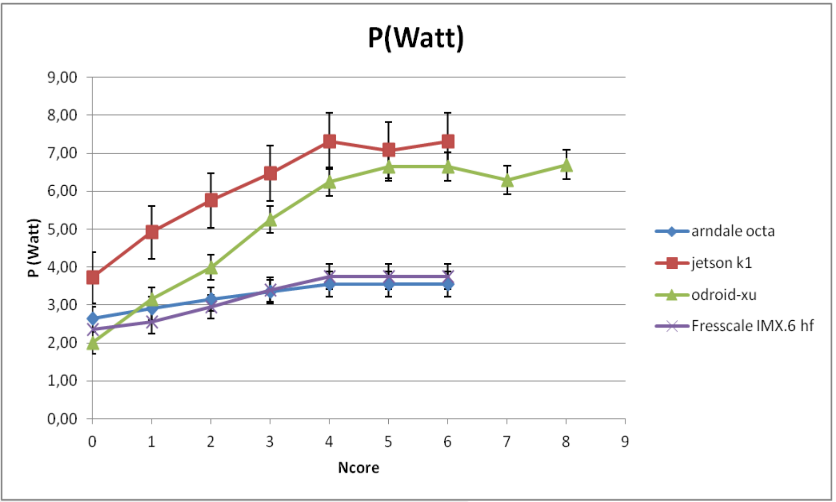
- We do *not* subtract power consumption of the idle board.

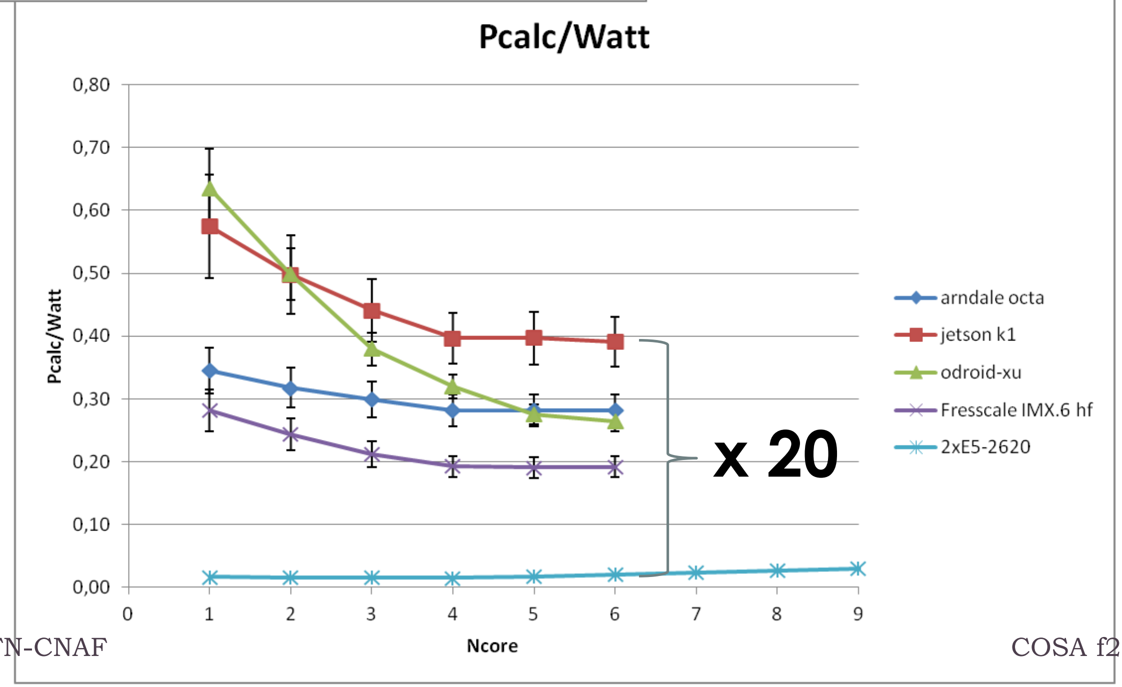
5 Preliminary tests

CPU ONLY

OpenMP π computation

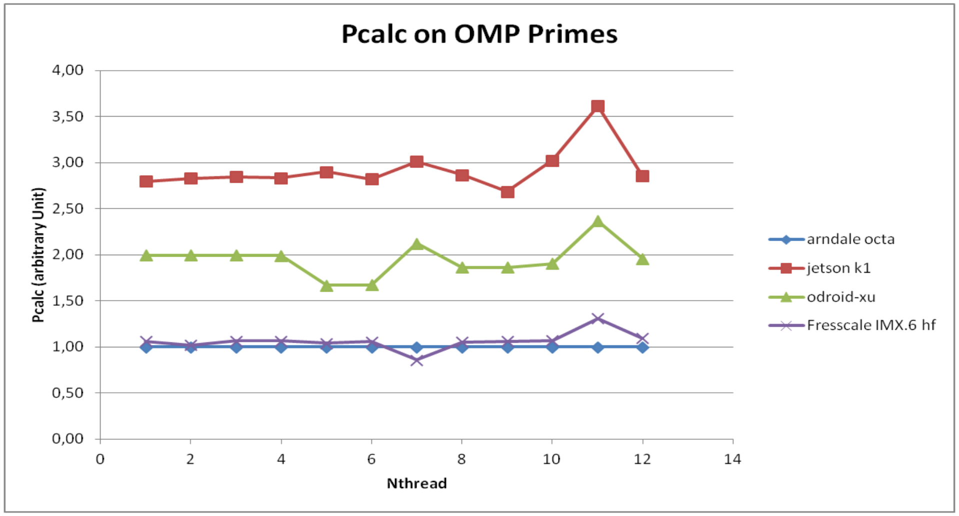


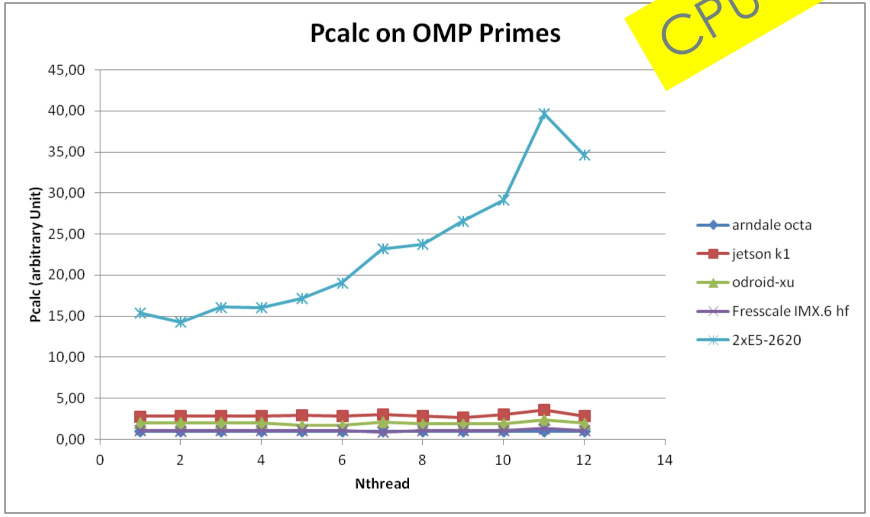


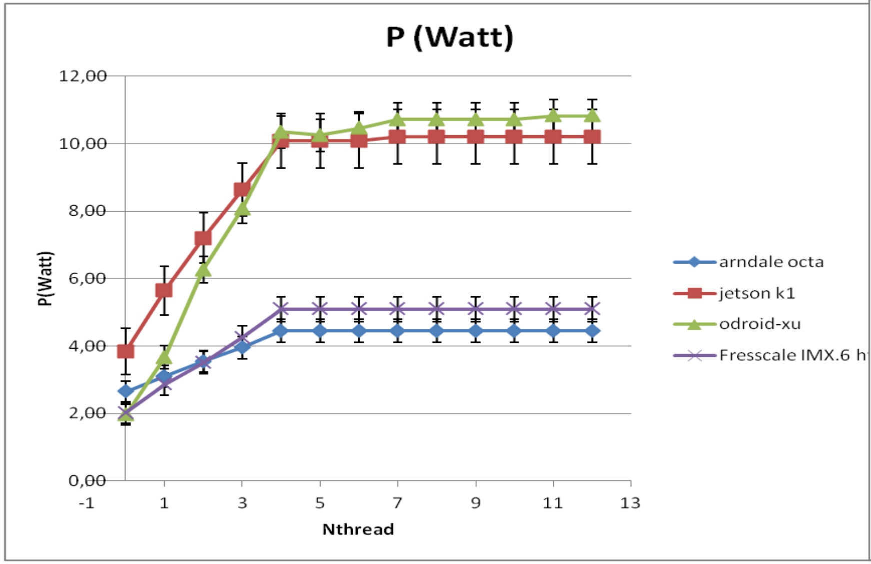


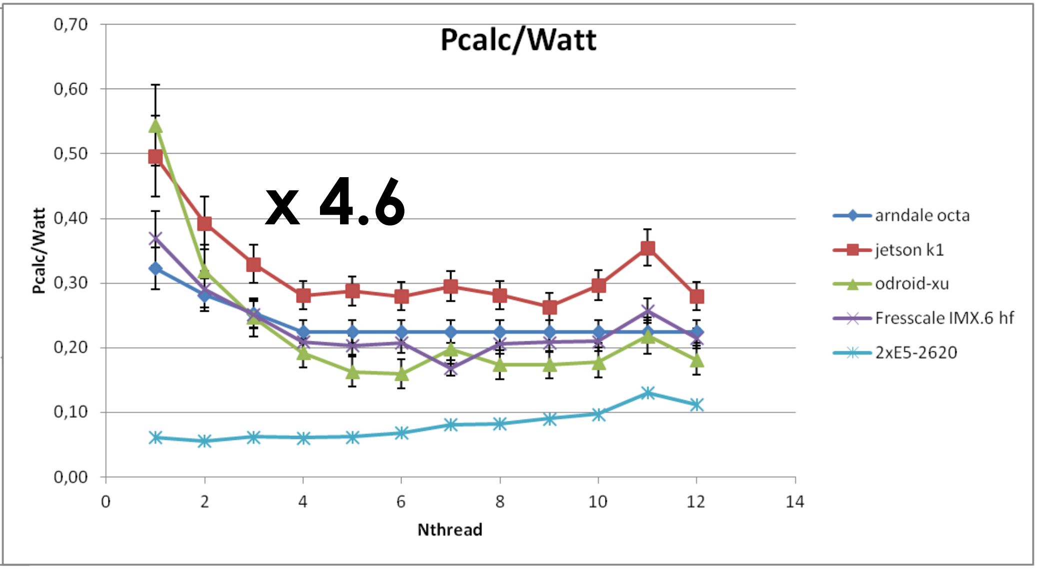
OpenMP Primes computation

CPU ONLY









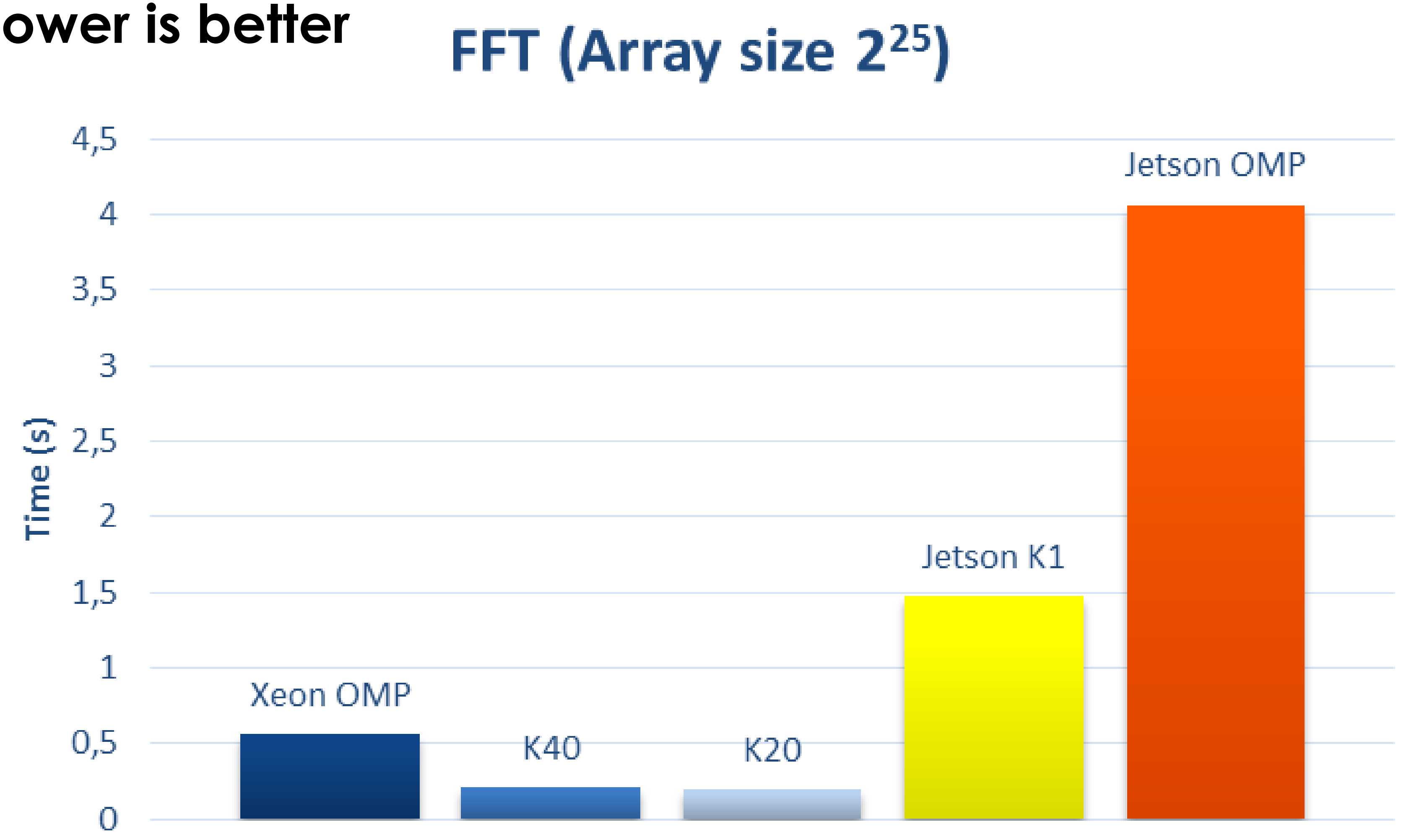
x 4.6

CPU & GPU

FFT

fftw3 on CPUs
cuFFT on GPUs

Lower is better

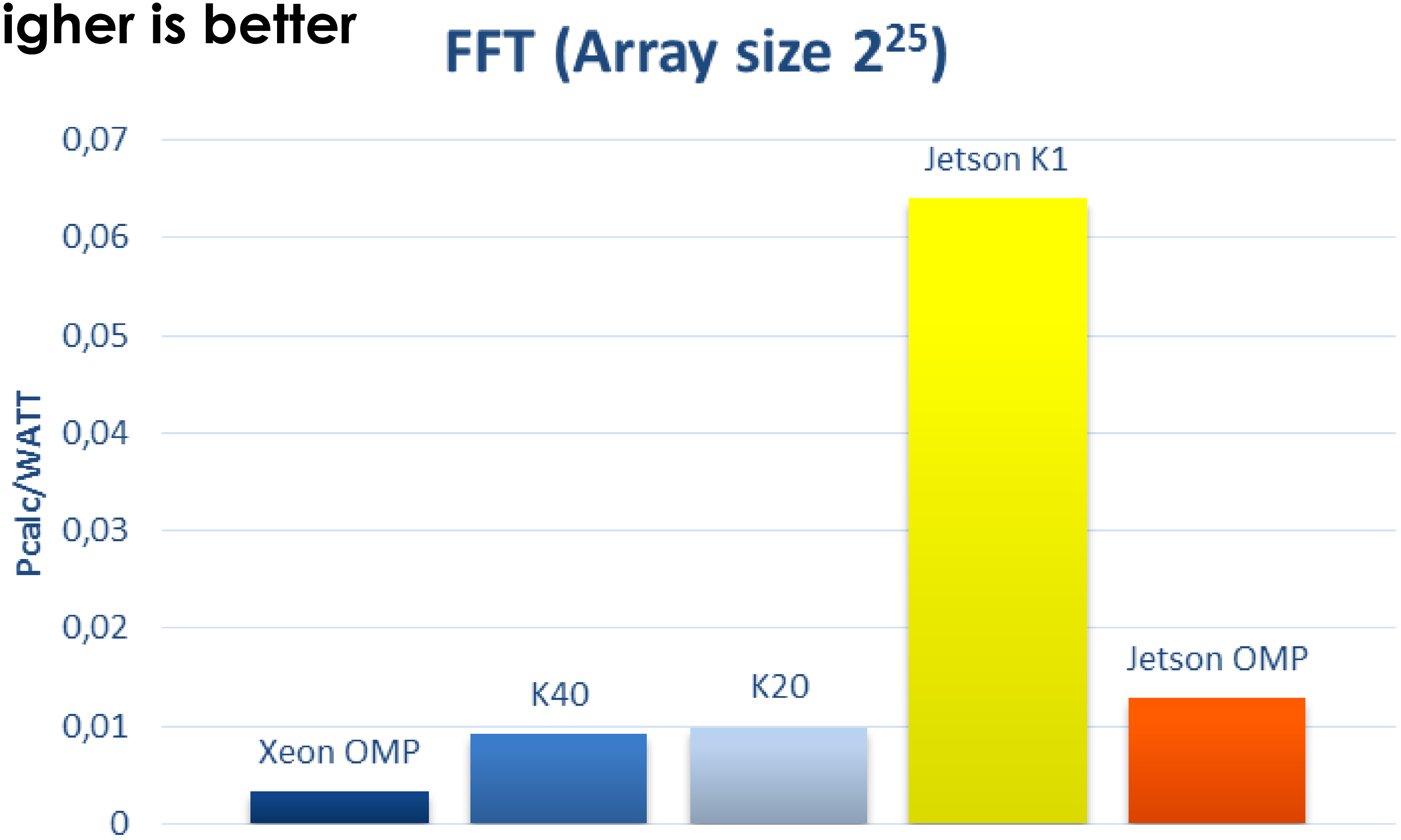


CPU & GPU

FFT

fftw3 on CPUs
cuFFT on GPUs

Higher is better



10

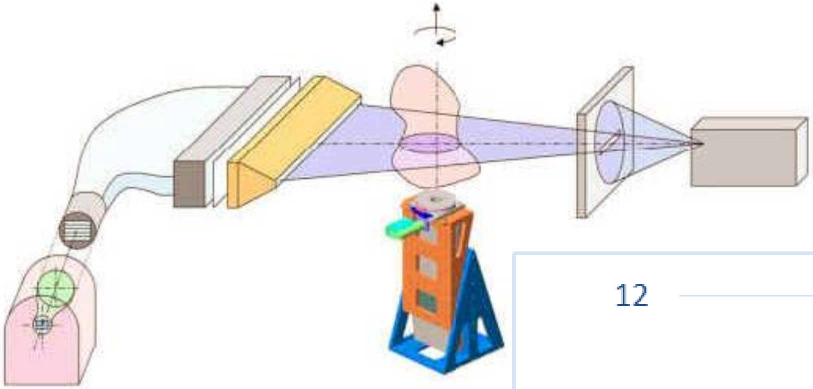
Computed Tomography

Master thesis of Elena Corni:

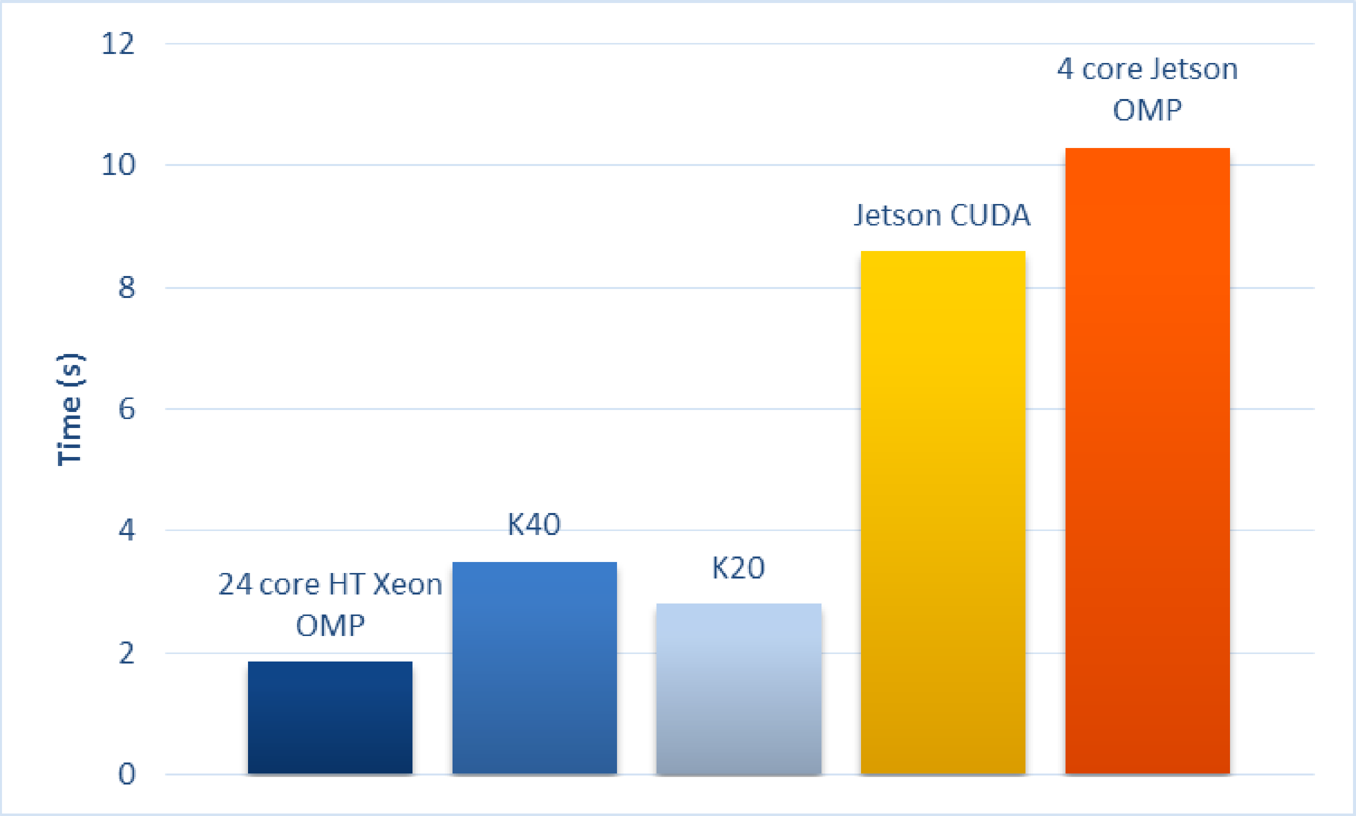
Implementazione dell'algoritmo Filtered Back-Projection (FBP) per architetture Low-Power di tipo Systems-On-Chip

Filtered Backprojection

CPU &GPU



Lower is better



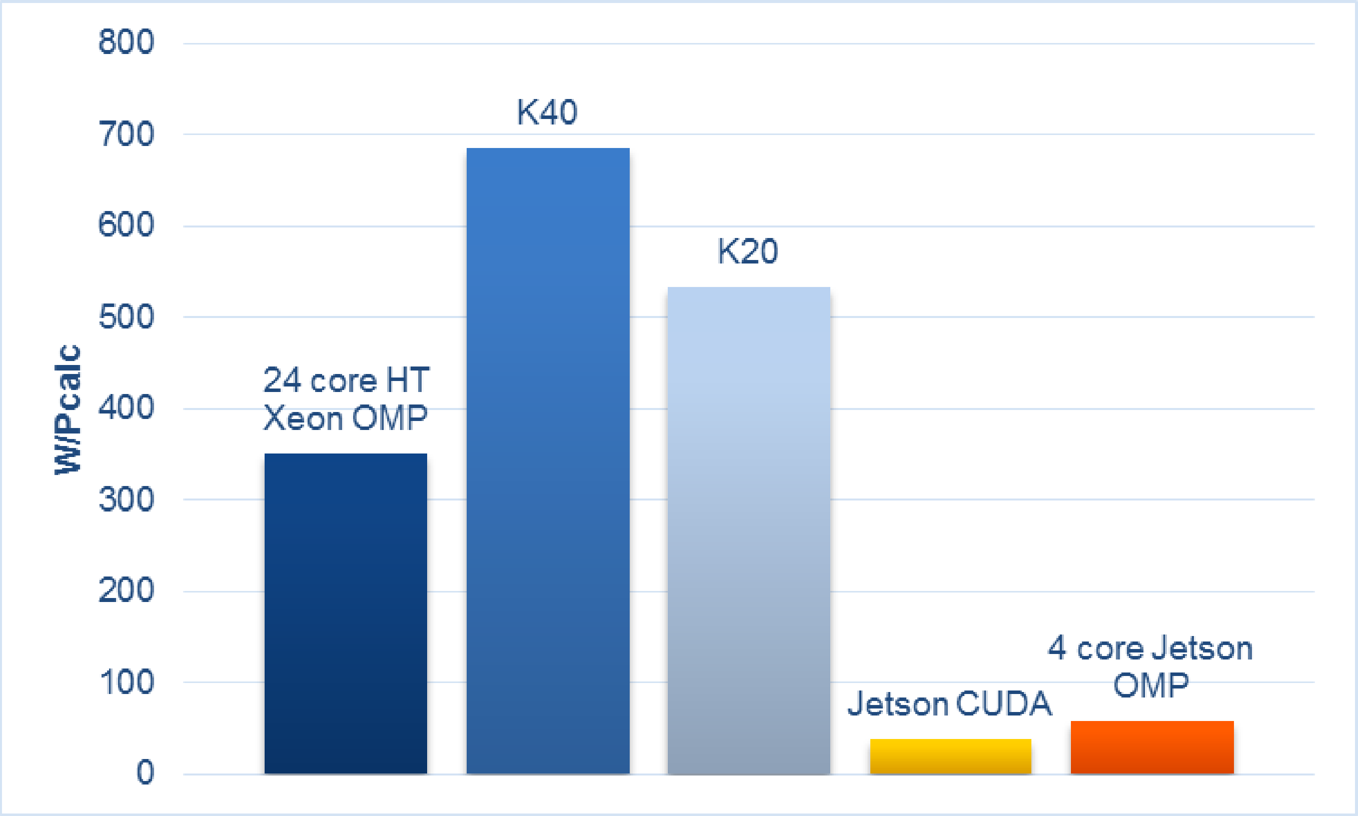
1024 x 1024 pixel

On (2xE5-2620+K20): 3241 slices in 1 h: **350** Wh

On **5** x Jetson-K1: 3840 slices in 1 h: **41** Wh

Lower is better

1024 x 1024 pixel



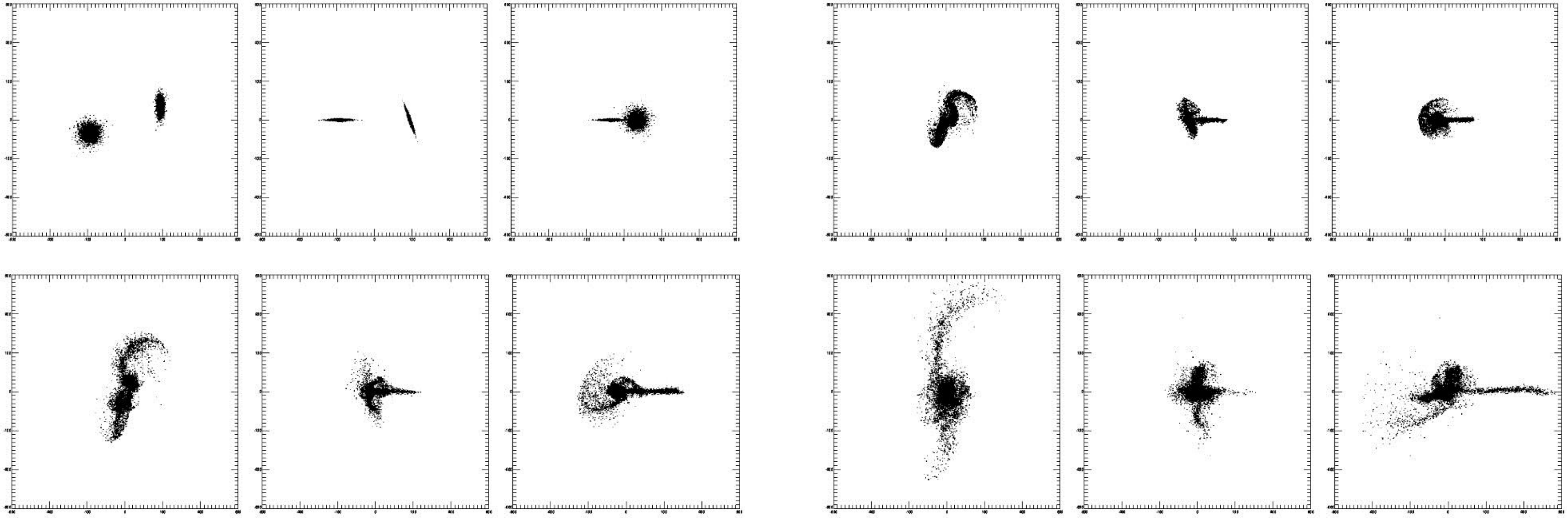
13 Astrophysics

Gadget-2: MPI code for cosmological simulations by Volker Springel

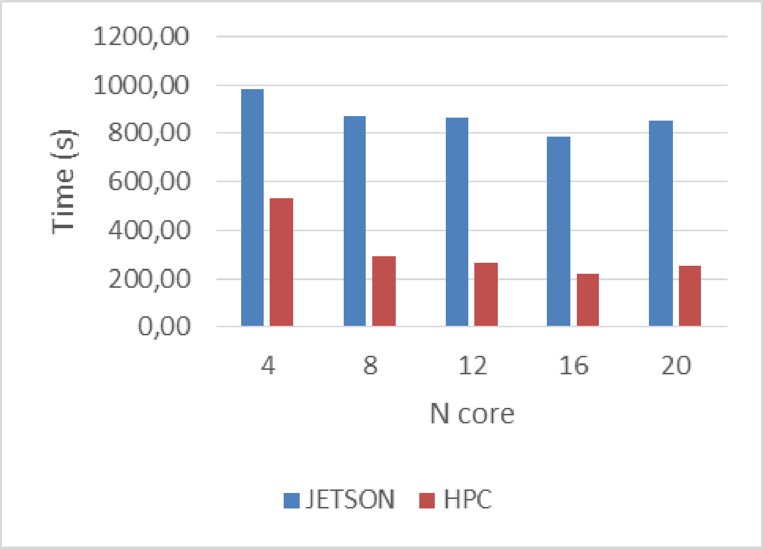
http://www.mpa-garching.mpg.de/gadget/

Galaxy collision

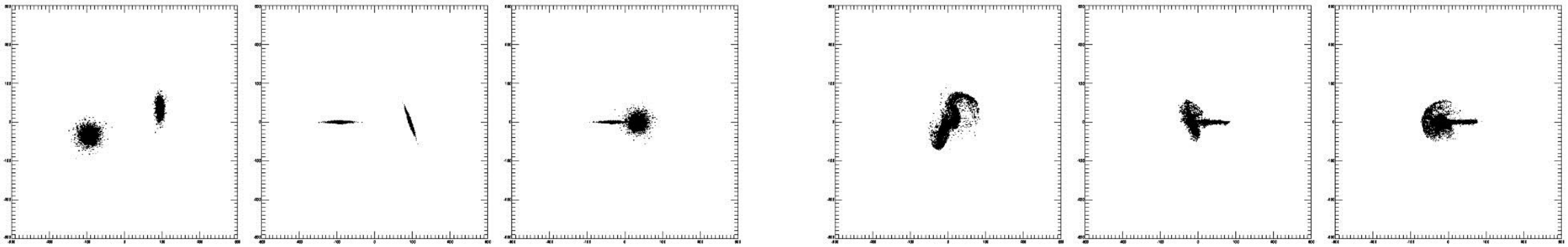
CPU ONLY

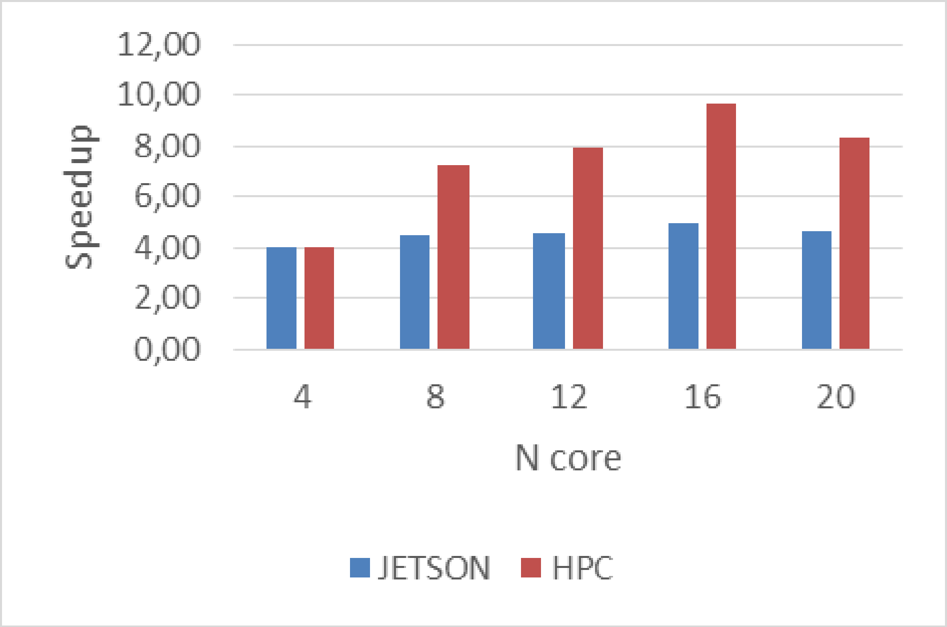


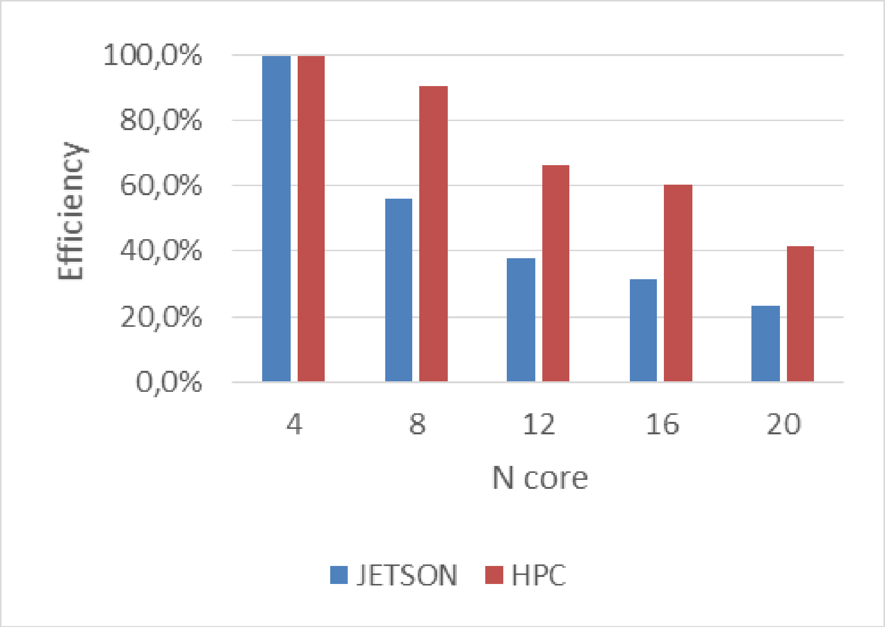
- 60000 disk + halo particles, Barnes Hut N-body simulation
  - Jetson-K1 **10.9 W**
  - Xeon **~220 W**



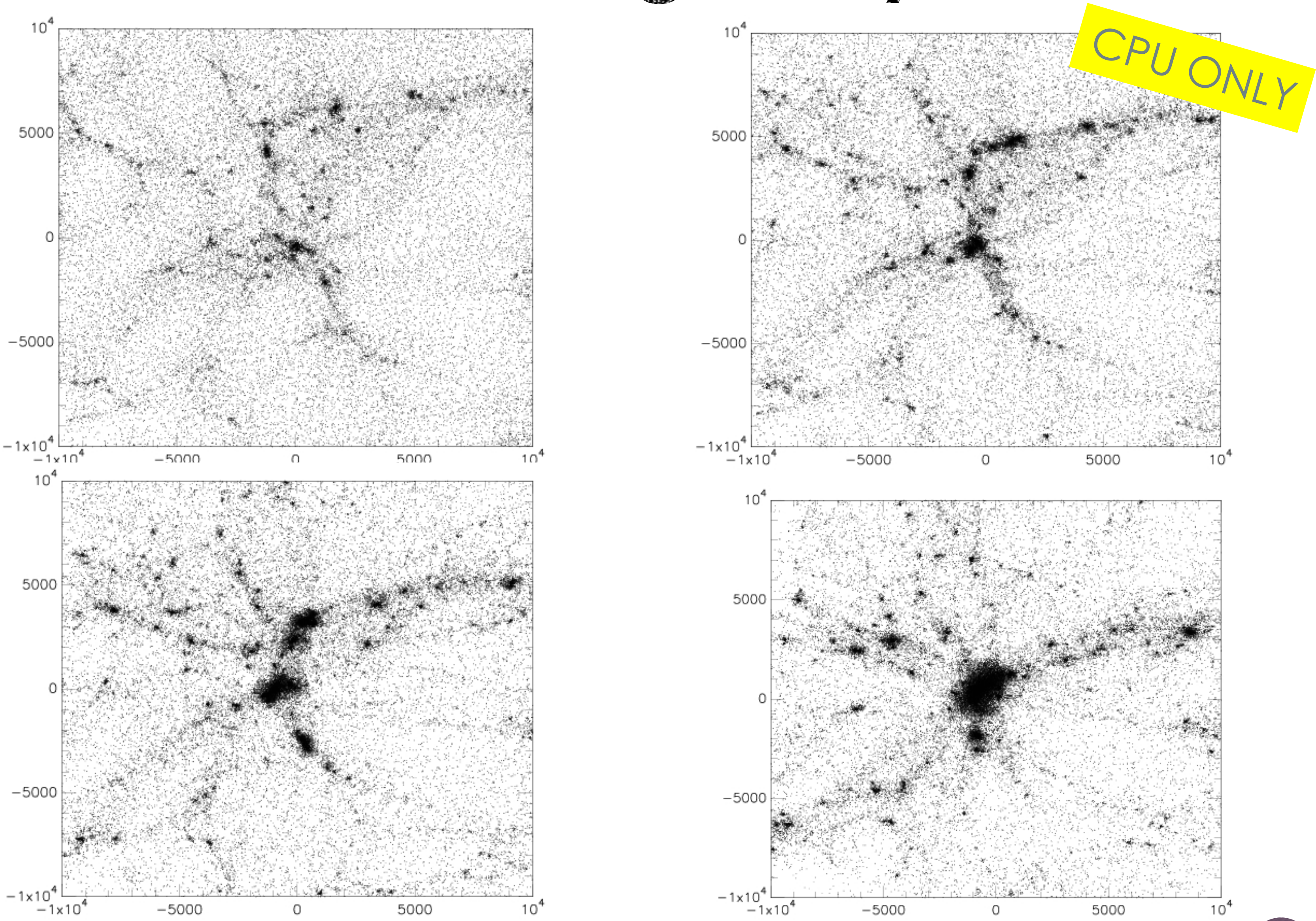
Galaxy collision







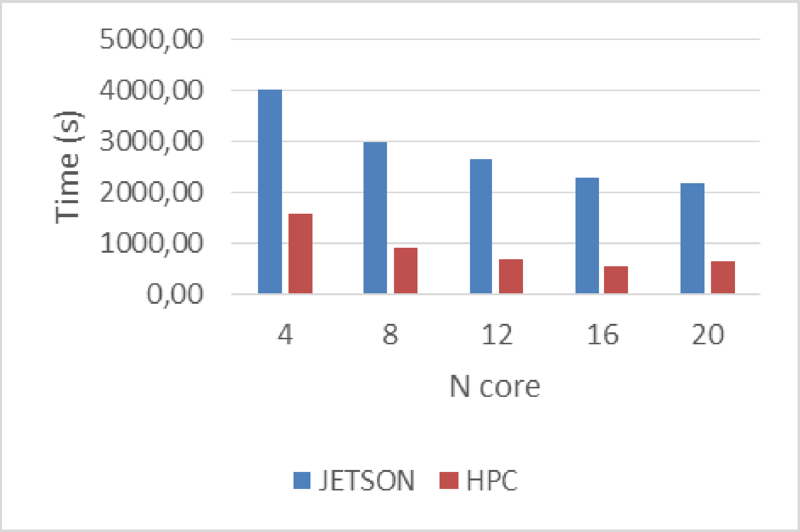
Formation of a galaxy cluster

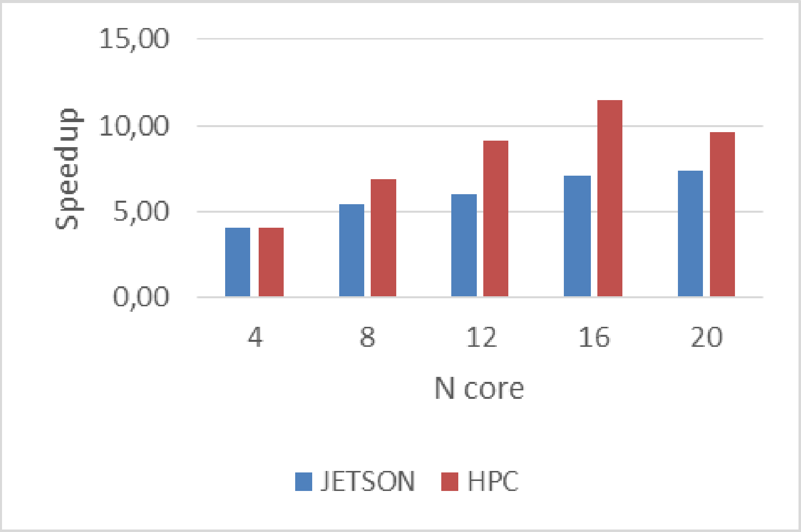


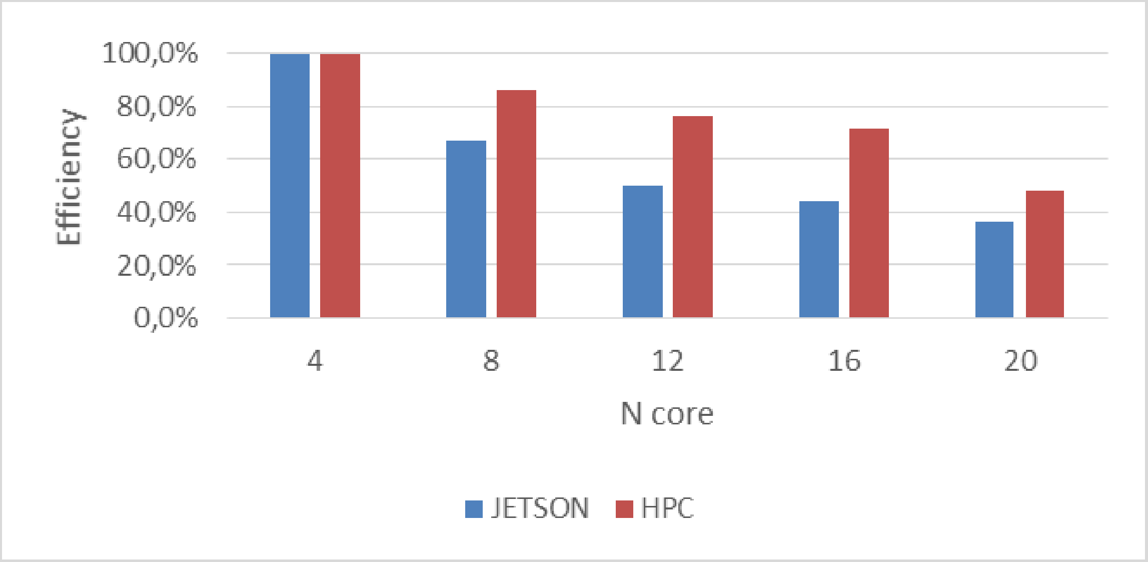
CPU ONLY

276498 particles, formation of a cluster of galaxies in an expanding Universe

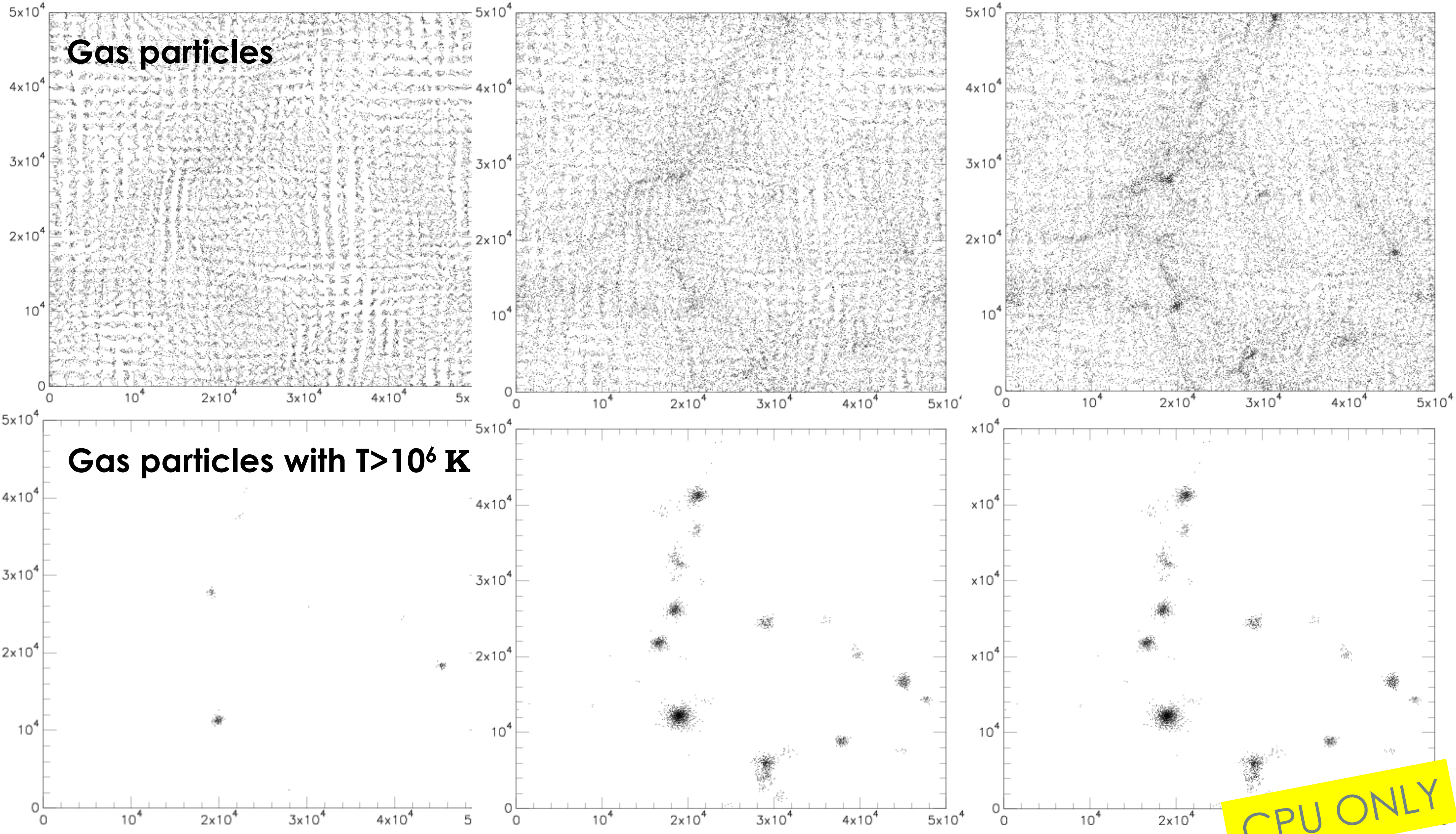
Formation of a galaxy cluster







Large-scale structure formation with gas



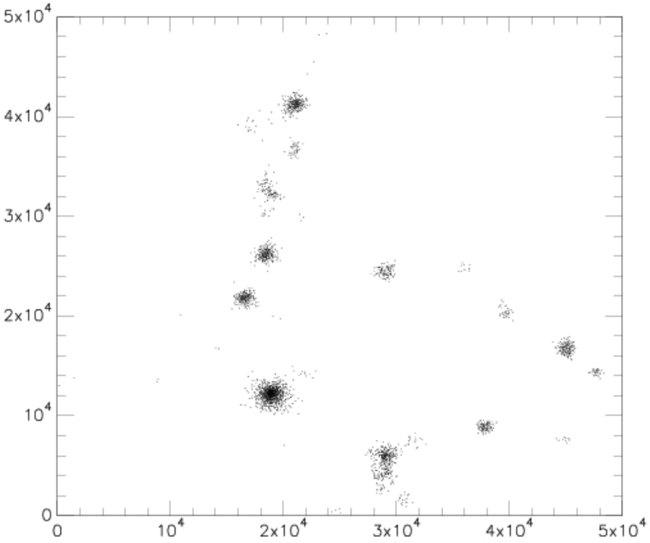
Gas particles

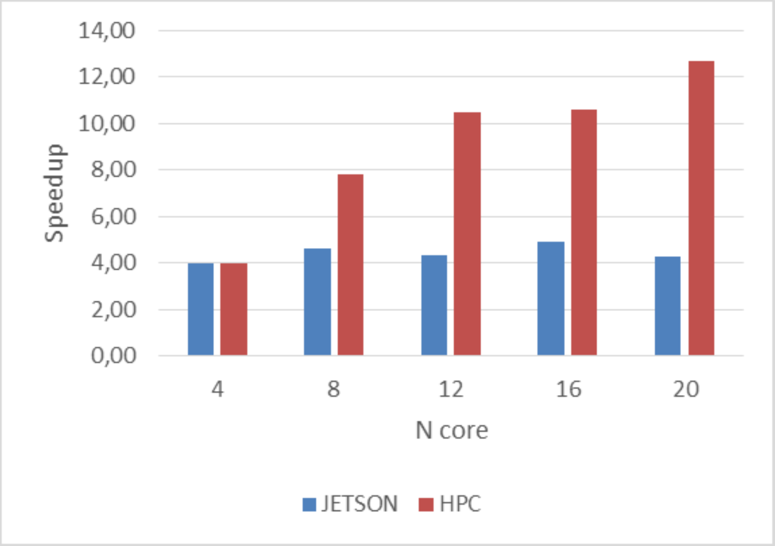
Gas particles with T>10⁶ K

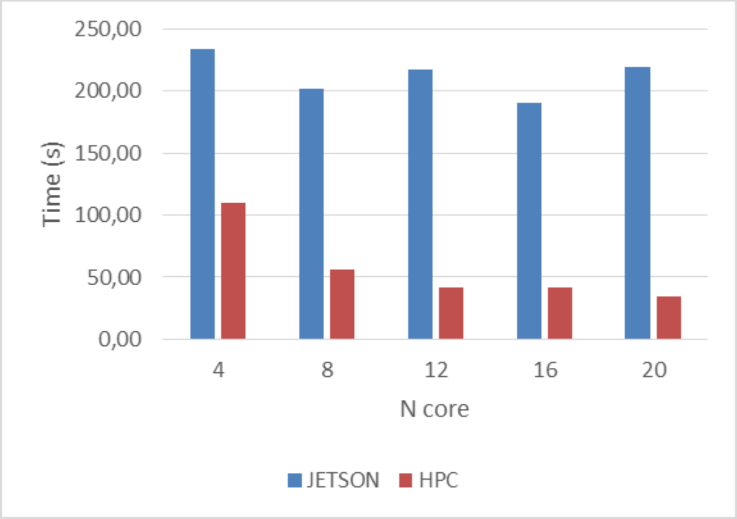
CPU ONLY

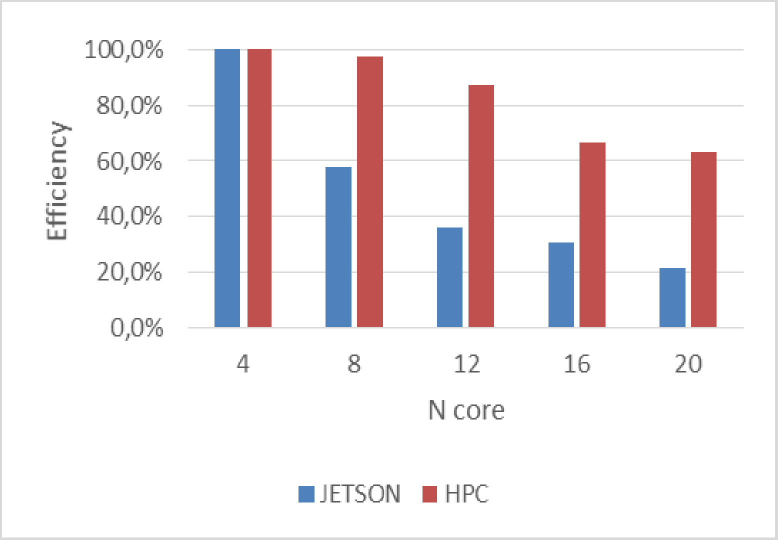
- 32^3 dark matter particles, 32^3 gas particles in a box 50 Mpc size, following structure formation in an evolving Universe, with adiabatic gas physics

Large-scale structure formation with gas



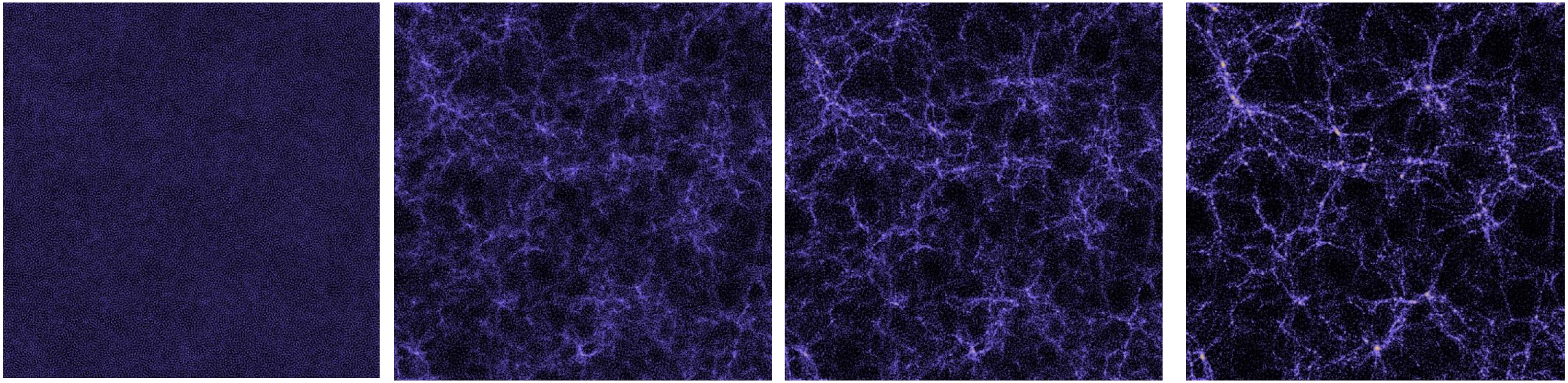






Real use-case: evolving the Universe

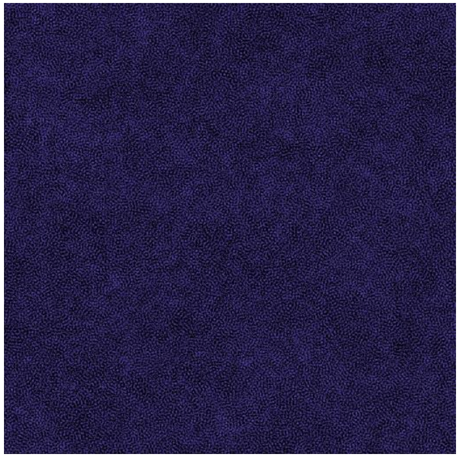
128^3 particles in a cube of 250 Mpc side, 13.6 Gyr

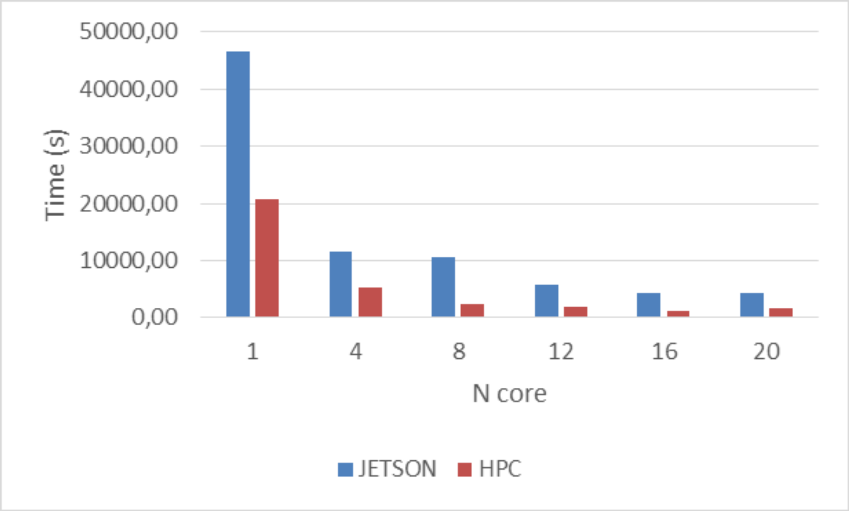


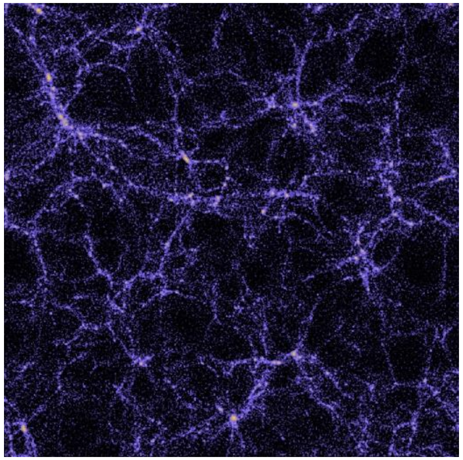
- Jetson-K1 about 2.6X slower than Xeon using 4 CPU cores, and about 4.4X slower using 8 CPU cores
  - Jetson-K1 **12.1 W**
  - Xeon **~220 W**

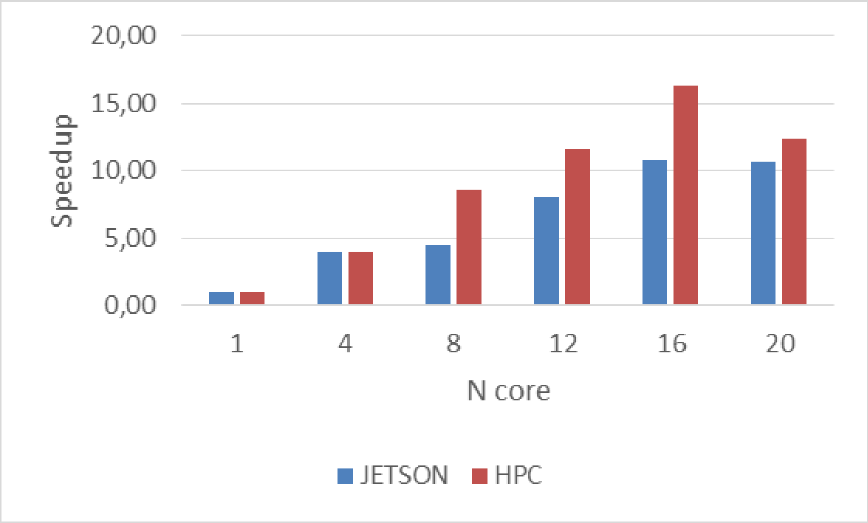
CPU ONLY

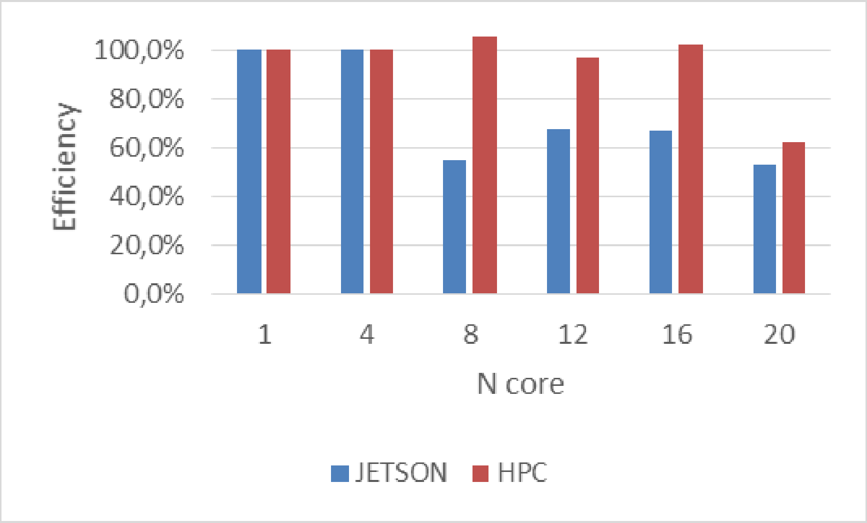
Real use-case: evolving the Universe





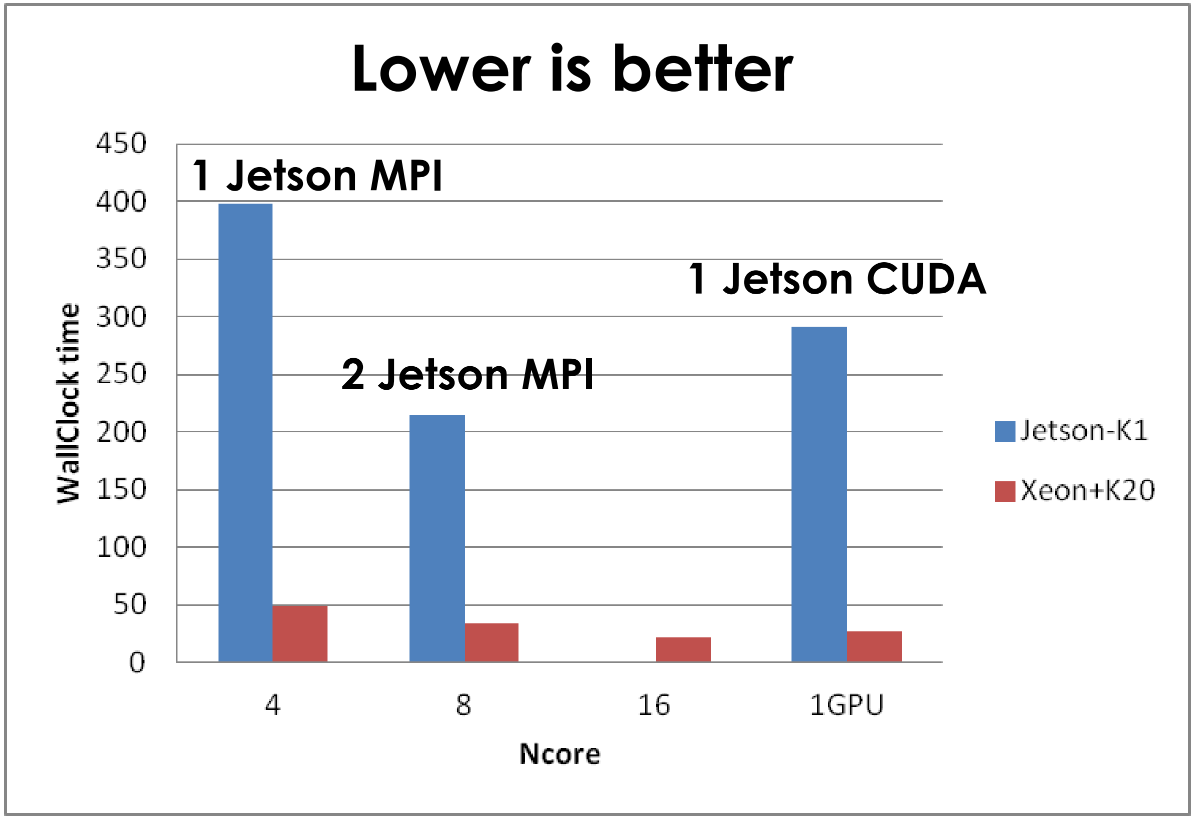


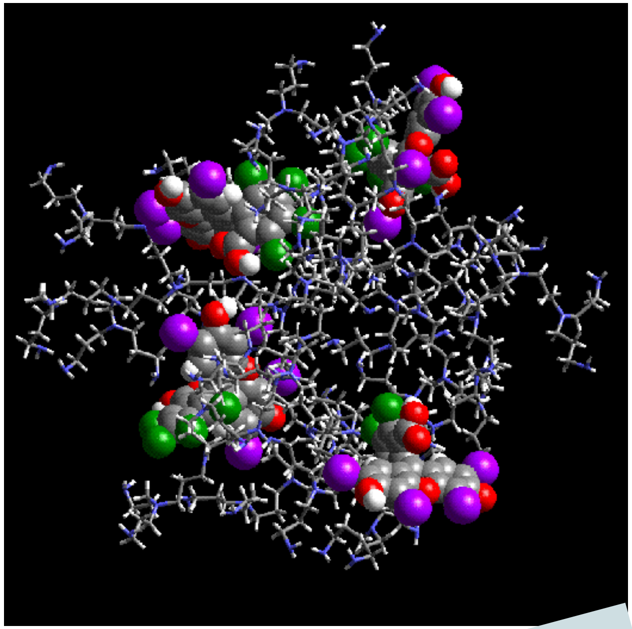




Molecular dynamics

Real life use case with GROMACS





CPU & GPU

- Jetson-K1 about 10X slower using the same number of CPU cores
- Jetson-K1 about 10X slower using the GPU (vs. an NVIDIA K20)

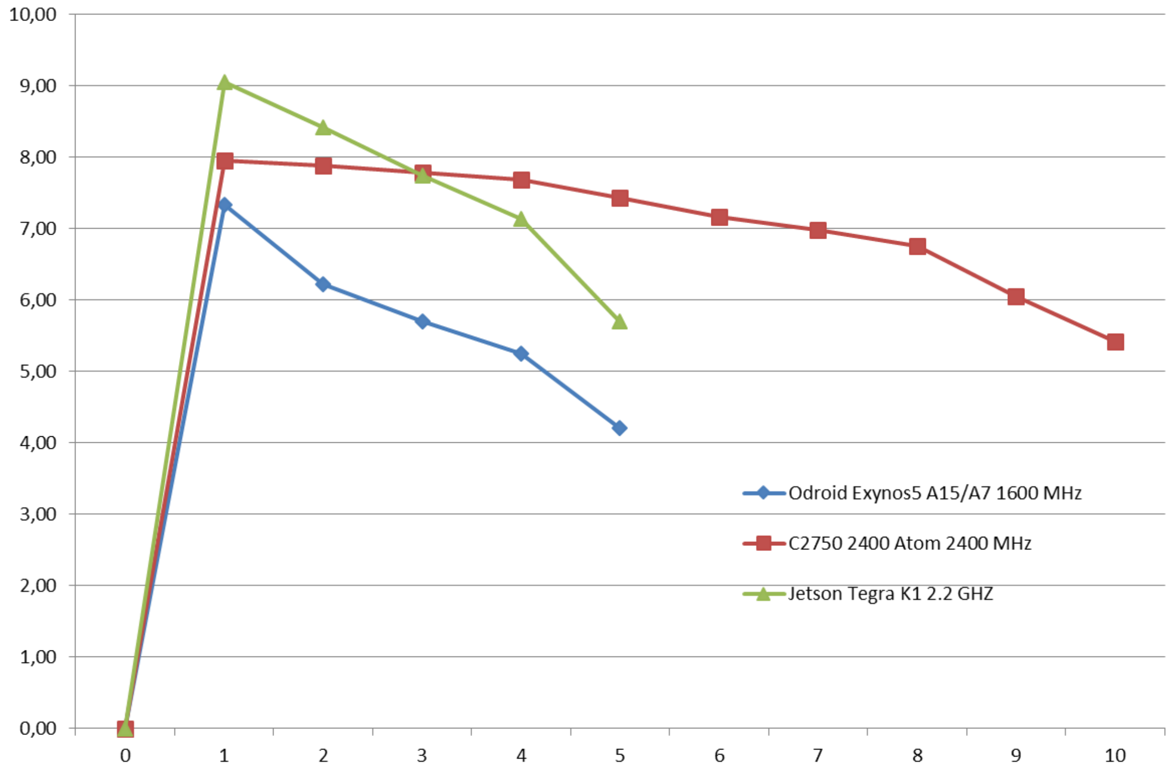
Jetson-K1 **13.5** W

Xeon+K20 **~320** W

HS

HS06

HS06 on Exynos5, TegraK1 and Atom C2750 – Per core loaded



(data from M.Michelotto@HEPIX 2014 https://indico.cern.ch/event/320819/session/3/contribution/30/material/slides/0.pptx)

A few comments on measuring HS

- Tests were ok on the Jetson, for 4 cores
- NFS doesn't work, eMMC works
- For 8 core boards (Odroid & Cubieboard), 1.5 GB of RAM are not enough for 8 processes
- So, for the Odroid, we put swap on eMMC, faster than SSD
- Now testing the Cubieboard, seems ok

The end (?)