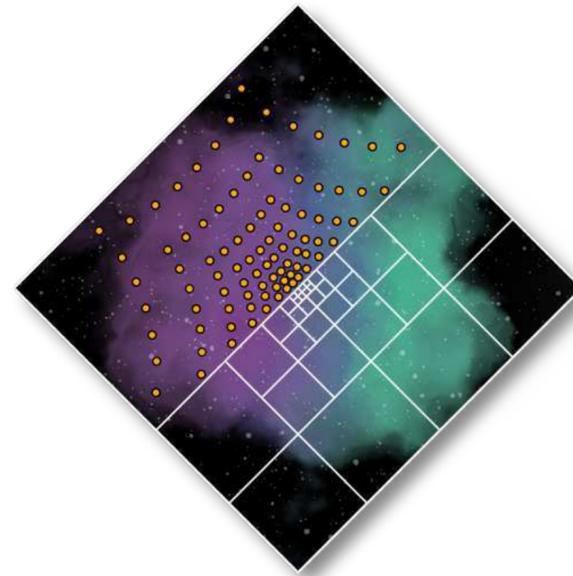




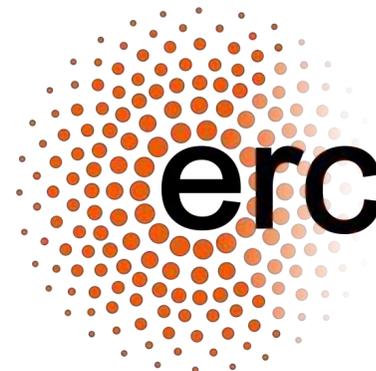
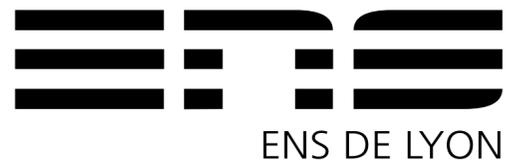
Leave a star
It helps packaging !



Shamrock



Beyond a PhD Code: The Rise of the Shamrock



No. 101053020 (Dust2Planets)

Last year !



Storms knock out power to 174,000 homes in Australia
 Published: 14 Feb 2024 - 08:25 am | Last Updated: 14 Feb 2024 - 08:32 am

Pictures of last year talk 😬

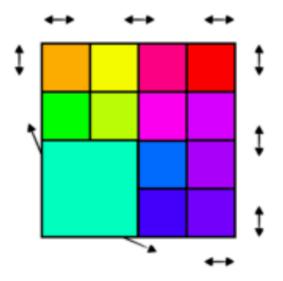


Domain decomposition

The global simulation is divided into multiple patches (abstract decomposition)

We only have to manage the interface to iterate

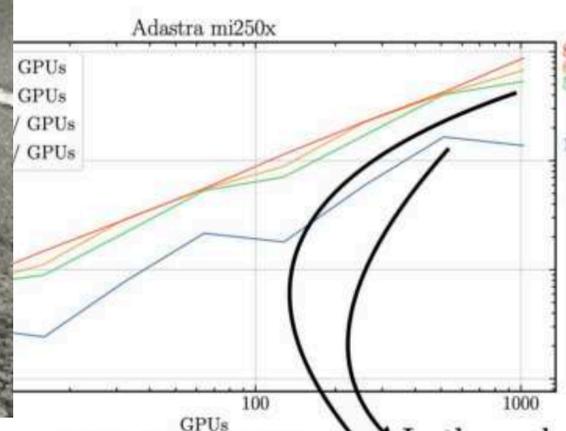
🔒 • >1G part in a simulation



Patches are scattered across the nodes

If patches are too large they split
If they are too small they merge

You take a big cluster



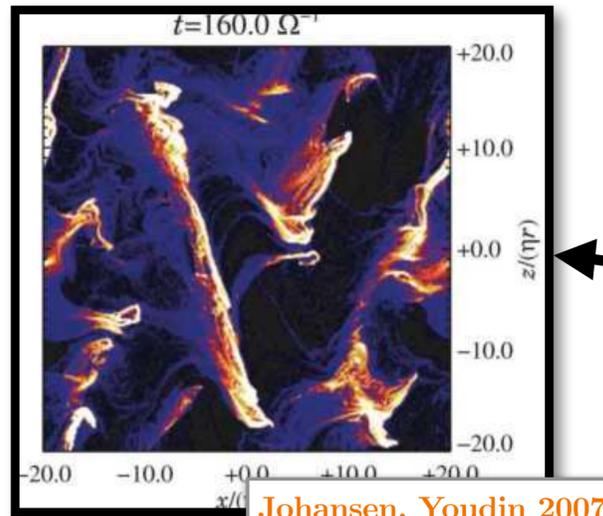
- 9G part / seconds
- 65.000.000.000 particles (4000³)
- 7 sec / iterations

Global time steps : (Estimate 10k steps at 1M part)
 Isothermal disc : 1G part = 1 orbit/16h
 Isothermal disc : 16G part = 1 orbit/4day



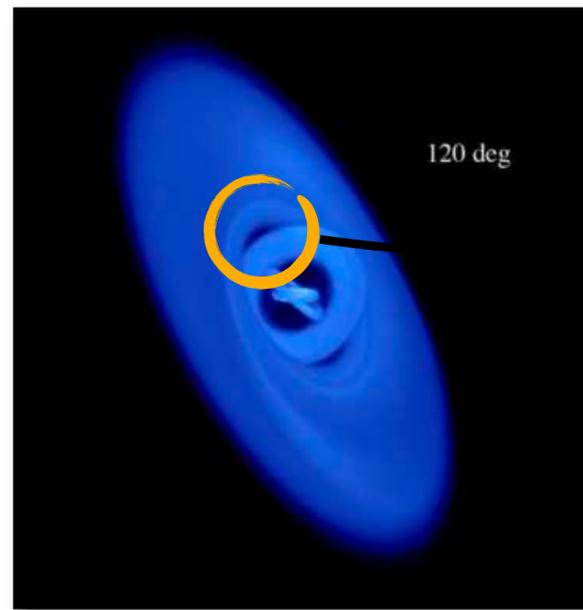
Sometimes we need a lot of resolution ...

Global instabilities ?
(1-10 Gpart)



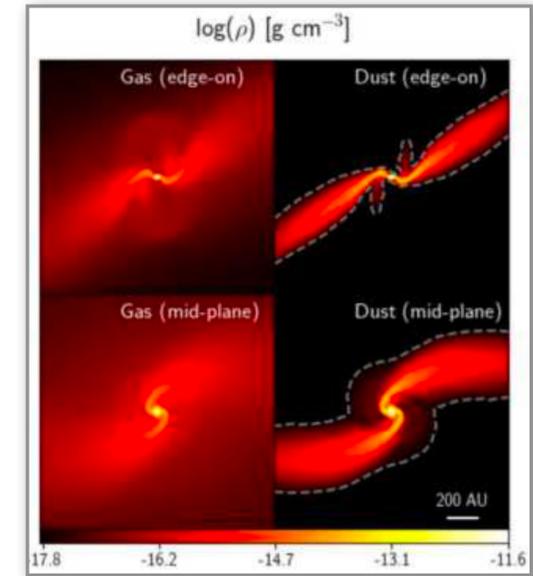
Johansen, Youdin 2007

What happens here ?

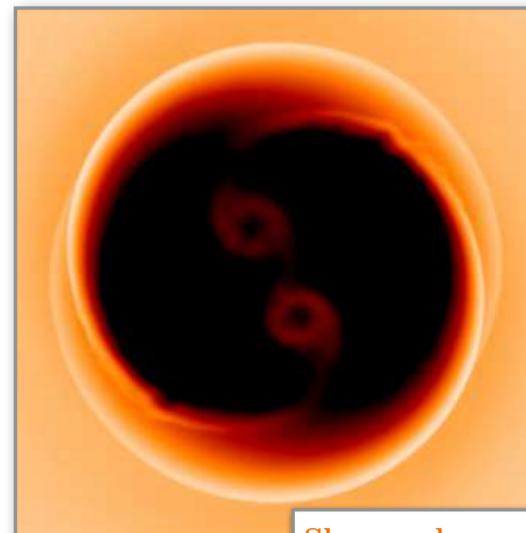


(1 Gpart)

Envelope, disc, low α_{num}
(1 Gpart)

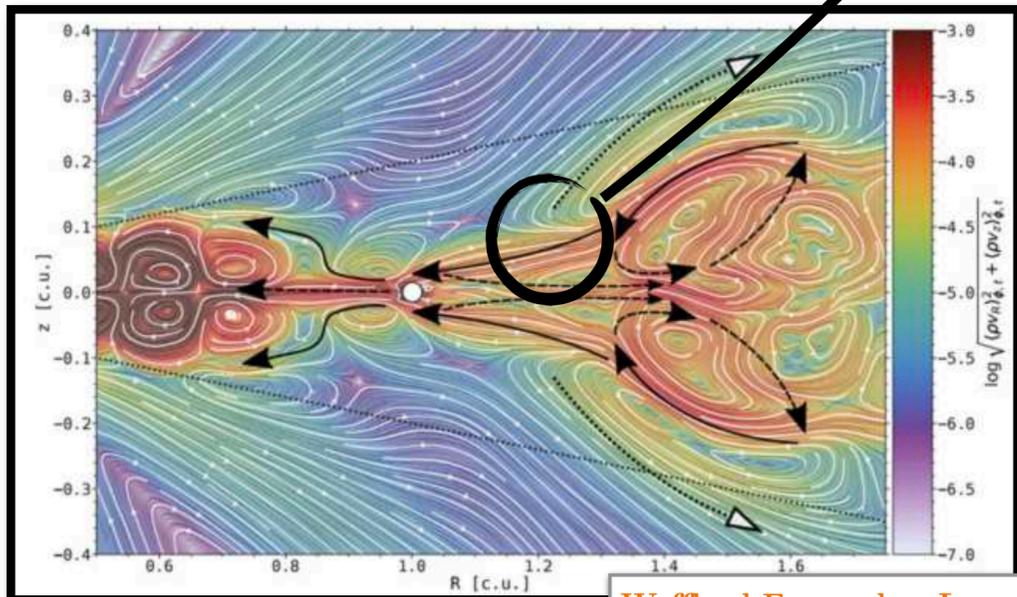


Circum-stellar accretion ?



(>1 Gpart)

Shamrock paper



Wafflard-Fernandez, Lesur 2023

$1\text{Gpart} \Leftrightarrow \alpha_{\text{num}} \simeq 10^{-3}, 10^{-4}$



1 Exaflop
= 10^{18} float operations per second

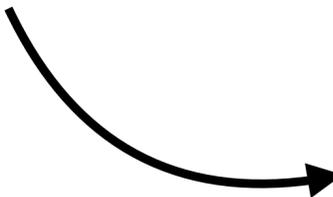
Phantom SPH code :



Limitations :

- CPU only
- No MPI scalability (limited to few nodes)
- Soft limit around 10-100M particles

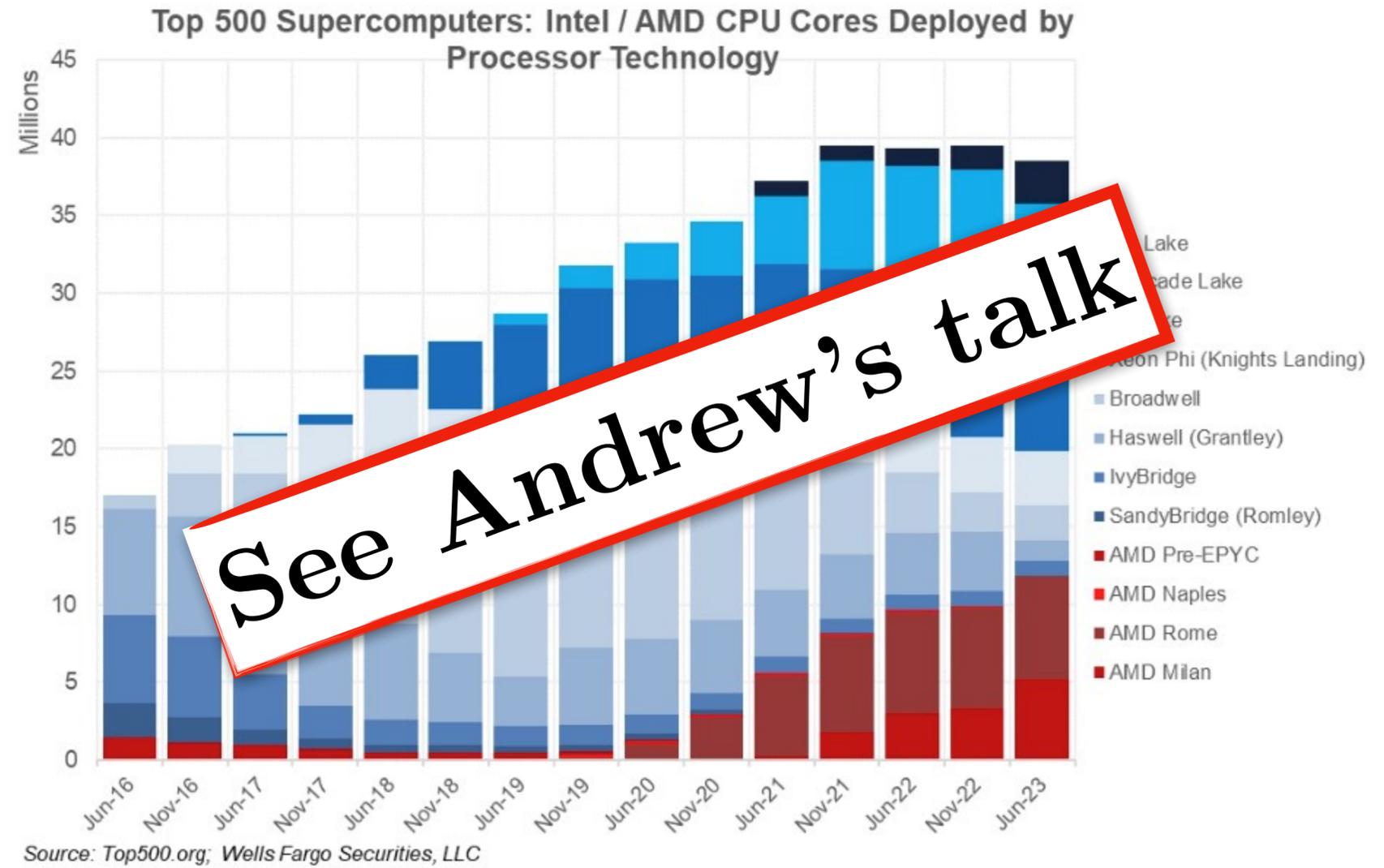
Phantom can not be ported to GPUs \Rightarrow New approach

- 
- Kd-tree build
 - Domain decomposition
 - Single loop approach



Top 500 list :

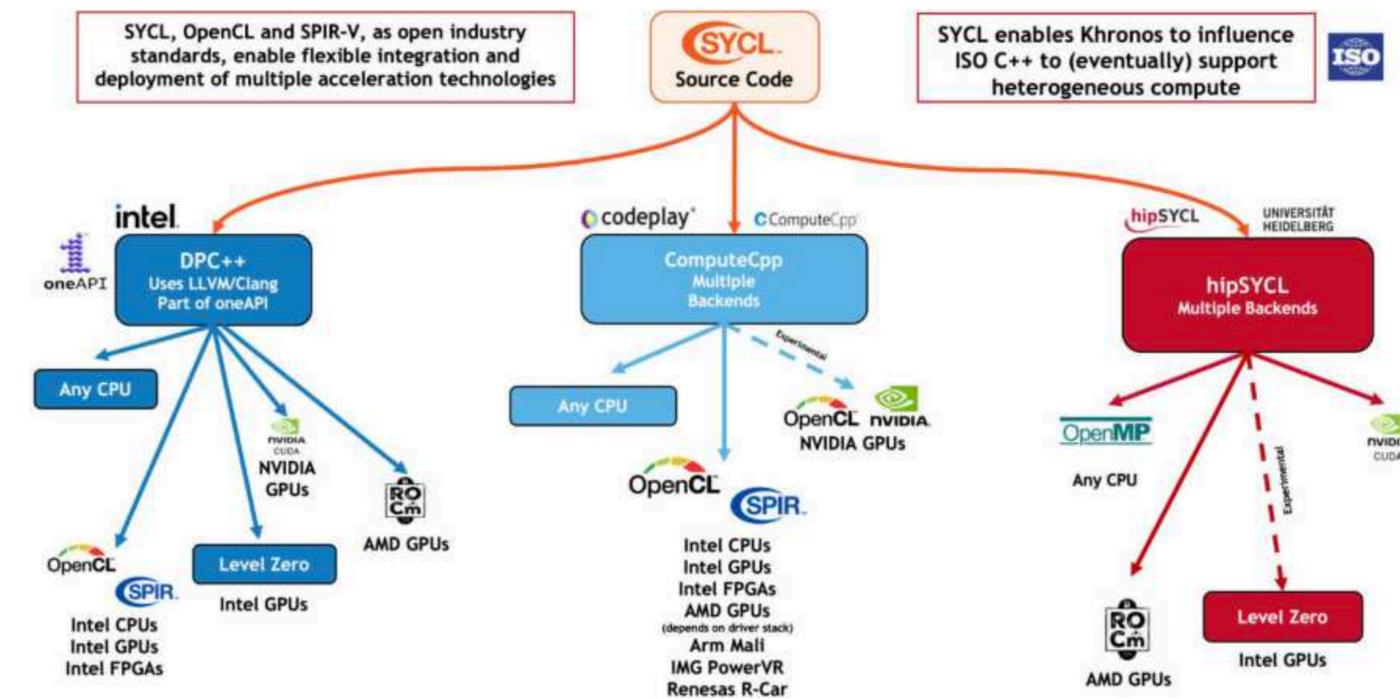
Rank	System	Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
1	El Capitan - HPE Cray EX255a, AMD Opti... AMD Instinct MI300A, Slingshot-11, TOSS, HPE DOE/NNSL/LLNL United States	11,039,616	1,742.00	2,746.38	29,581
2	Frontier - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE Cray OS, HPE DOE/SC/Oak Ridge National Laboratory United States	9,066,176	1,353.00	2,055.72	24,607
3	Aurora - HPE Cray EX - Intel Exascale Compute Blade, Xeon CPU Max 9470 52C 2.4GHz, Intel Data Center GPU Max, Slingshot-11, Intel DOE/SC/Argonne National Laboratory United States	9,264,128	1,012.00	1,980.01	38,698
4	Farlo - Microsoft Nov5 Xeon Platinum 8400C, NVIDIA NVIDIA H100, NVIDIA Infiniband NDR, Microsoft Azure Microsoft Azure United States	2,073,600	561.20	846.84	
5	HPC6 - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, RHEL 8.9, HPE Eni S.p.A. Italy	3,143,520	477.90	606.97	8,461



⇒ Supercomputers move to GPUs
 (Efficiency, density, AI, ...)
 (But all vendors involved)

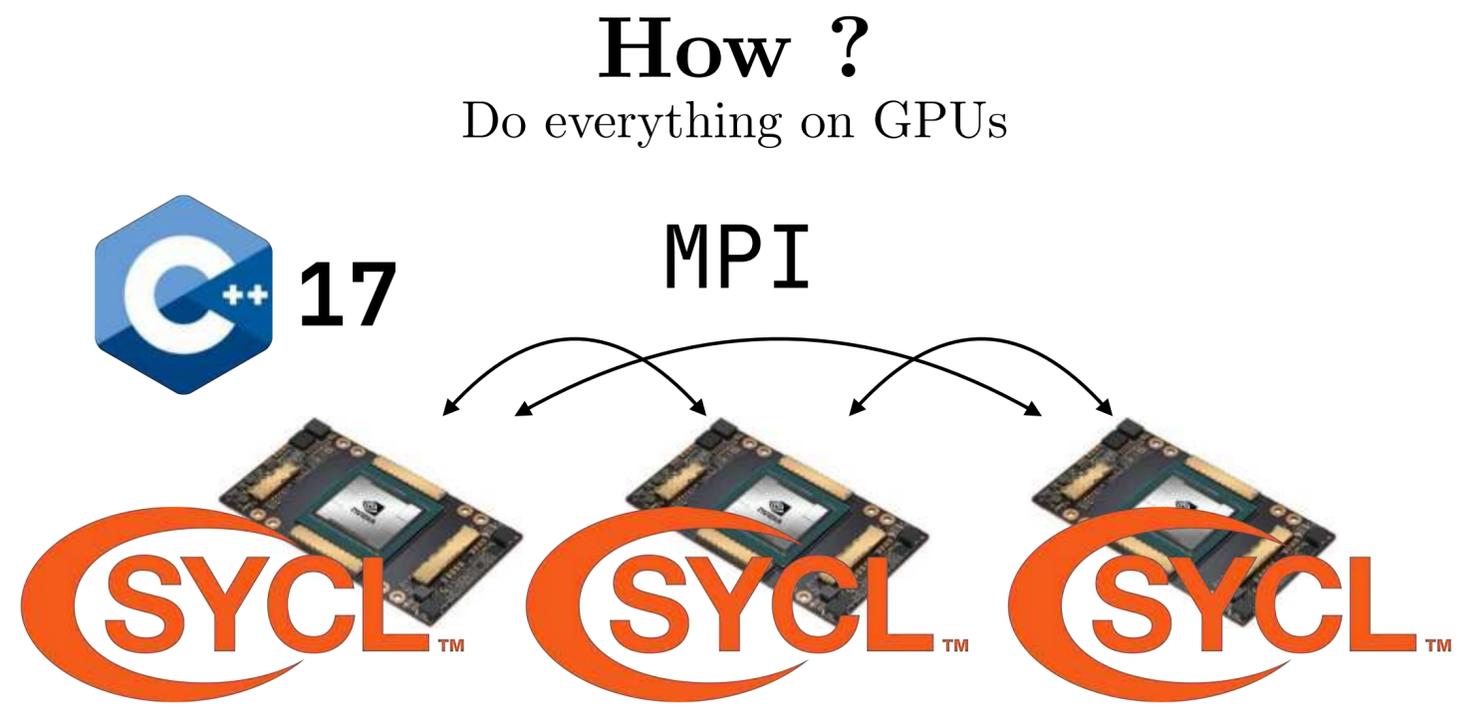
Could also happen with NPUs soon-ish

Our choice :



- C++17 extension
- Programming standard
- Directly compiles through CUDA, Hip, OpenMP, ...
- Supports all major CPU & GPUs (AMD, Nvidia, Intel, ARM)

The goal: Exascale



13th International Workshop on OpenCL and SYCL

IWOCL 2025

Shamrock: Exascale Hydrodynamics for Astrophysics Using SYCL

Timothée David Cléris, University of Grenoble.

April 7-11, 2025 | Heidelberg, Germany | iwocl.org

SHAMROCK: Exascale Hydrodynamics for Astrophysics Using SYCL

IWOCL 827 subscribers

SHAMROCK: Exascale hydrodynamics for astrophysics using SYCL.

Timothée David-Cléris
Univ. Grenoble Alpes, CNRS, IRAG
38000 Grenoble, France
david@iposy.grenoble-alpes.fr

Figure 1: Snapshot of an MPI simulation of a circum-binary disc using Shamrock with one billion particles, made using 10 AMD MI300X on the Ahabus supercomputer.

Abstract

We present Shamrock, a native SYCL framework for astrophysics, designed to implement various numerical methods for modelling hydrodynamic flows, in particular Smoothed Particle Hydrodynamics (SPH). As the case of Shamrock is a fast multi-scale building algorithm that allows the user to be explicit at each timestep with minimal cost, obtaining the need for low communication or updates. Additionally, a stream interpolation method is used on top of the naive one, allowing for a nearly three-fold GPU weak scaling, resulting in 92% weak scaling efficiency on 100 MI300X AMD graphics accelerators for large SPH simulations.

CCS Concepts

- Applied computing → Physics & Computing mathematics
- Massively parallel and high-performance simulation.

Keywords

High performance computing, Astrophysics, SYCL, MPI, GPU, exascale, Smoothed Particle Hydrodynamics, SPH.

ACM Reference Format

Timothée David-Cléris, 2025. Shamrock: Exascale hydrodynamics for astrophysics using SYCL. In Proceedings of International Workshop on OpenCL and SYCL.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted by ACM, provided that the fee code and the full name of the conference are printed on the front cover. For all other uses, permission should be sought from ACM. Copyright © 2025, ACM. Copyright held by the owner/authors. Publication rights licensed to ACM. ACM is not responsible for the content or the accuracy of any information published in this work. Request permission for reprints from ACM.

1 Introduction

The study of astrophysical flows in the context of structure, stars, or planet formation involves multi-scale, multi-physics, out-of-equilibrium processes, which require the use of numerical simulation. Comprehensive studies of all the involved scales in such processes require a large amount of data and long integration times, necessitating extreme computational power. Although there are existing codes for simulating astrophysical processes, most still rely exclusively on CPU-based computation and are not optimized to reach the petascale from being ported to multi-GPU architectures. This implies that new existing codes need to be rewritten, often from scratch, with hybrid architectures in mind.

In this advantage of exascale architectures, new codes have been developed. However, modern exascale clusters come with hardware from different vendors (e.g., Nvidia, AMD, and Intel), each with a specific backend for GPU acceleration. As a result, directly targeting all of them would require intricate customizations. This involves the use of a portability layer, such as OpenCL based in the aforementioned context or SYCL, which we employ in our case. Known in a library, these making such porting dependent on a single implementation. To avoid such a possible single point of failure and ease our migration toward a unified source language for our multi-accelerator system, we primarily decided using SYCL standard. This allows our code to work with any implementation supporting it. As of now, Shamrock supports both OpenCL and AMDGPU [1]. Additionally, all our numerical astrophysics codes focus solely on grid-based methods, which can be limiting for those involving complex, evolving geometries in astrophysics, for such

The SHAMROCK code: 1 – smoothed particle hydrodynamics on GPUs

T. David-Cléris^{1,2,3*}, G. Laibe^{1,3} and Y. Lapeyrie¹

¹Univ. Grenoble Alpes, CNRS, IRAG, 38000 Grenoble, France
²CNRS, IRAG, Université Grenoble Alpes, 7, 38000 Grenoble, France
³Univ. Grenoble Alpes, France

Accepted 2023 March 13. Received 2023 March 11; in original form 2024 May 23.

ABSTRACT

We present SHAMROCK, a performance portable framework developed in C++ 17 with the SYCL programming standard, tailored for numerical astrophysics on Exascale architectures. The core of SHAMROCK is an accelerated parallel code with multiple communication layers, whose efficiency is based on binary algebra. The smoothed particle hydrodynamics algorithm of the SHAMROCK code is implemented in SHAMROCK. On-the-fly tree construction circumvents the necessity for extensive data communication. It is built displaying a uniform density with global time-stepping with one of billions of particles. SHAMROCK completes a single time-step in a few seconds using over the thousand of GPUs of a supercomputer. This corresponds to processing billions of particles per second, with less of millions of particles per GPU. The parallel efficiency across the entire cluster is larger than ~90 per cent.

Key words: methods: numerical.

1 INTRODUCTION

The study of the formation of structure in the Universe is a field in which new frontiers are opening. The physical processes involved in these different scales, requiring the use of numerical simulation to simulate them, right up to Exascale (one quadrillion operations per second). To increase energy efficiency with acceptable CO₂ emissions, recent supercomputers have been designed with specialized hardware such as ARM central processing units (CPUs) or graphics processing units (GPUs). GPUs involve multiple computational units that perform the same operations on multiple data simultaneously through specific instructions (Single Instruction Multiple Data, or SIMD parallel processing). This type of hardware offers radically from standard CPU CPUs, requiring a complete rewrite of CPU-based codes.

Consequently, efforts have recently been invested into developing codes adapted to the new hybrid architectures based at Exascale (e.g. KERRI, Lauer et al. 2023; PARTHENON, Grez et al. 2023; QUINTANA, Wilkins & Krumholz 2023). Most of the codes in the community targeting Exascale are grid-based. However, these methods can be inadequate for systems with complex geometries or unstructured flows, highlighting the need for smoothed particle hydrodynamics (SPH) methods. However, traditional SPH methods call for a different design from the state-of-the-art methods developed for CPUs (e.g. Laibe & Krumholz 2015). The simulation domain undergoes an initial partitioning into subdomains, featuring communication and interface exchange through an coarse layer of MPI parallelism. The core of simulation is in the layer of particles, which consists in processing each subdomain on GPUs using the SYCL standard. This includes a new algorithm that delivers the performance required for handling any number of objects compatible with various GPU capabilities. It combines state-of-the-art algorithms based on Morton codes (Salmon 1986; Warren & Salmon 1995; Luthardt et al. 2009)



1
= 10¹⁸ float o

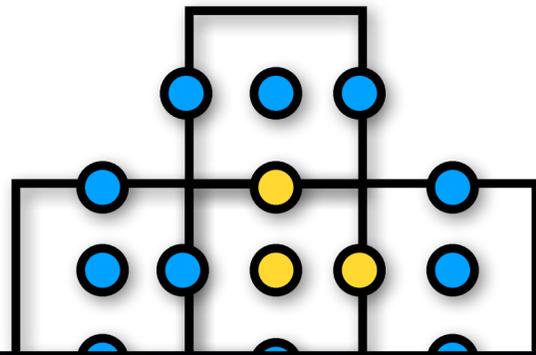
Technical talk at IWOCL

Accepted

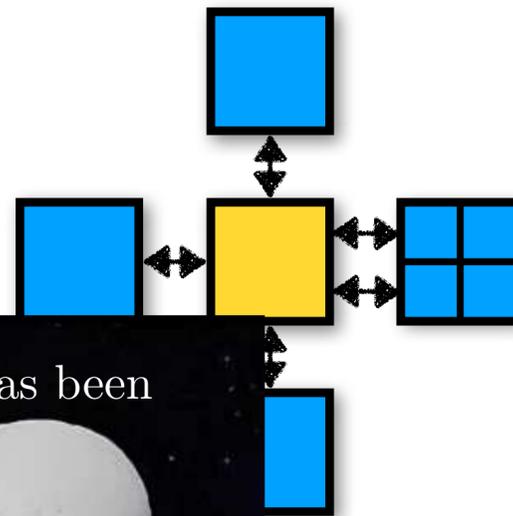
Published

2-slide summary

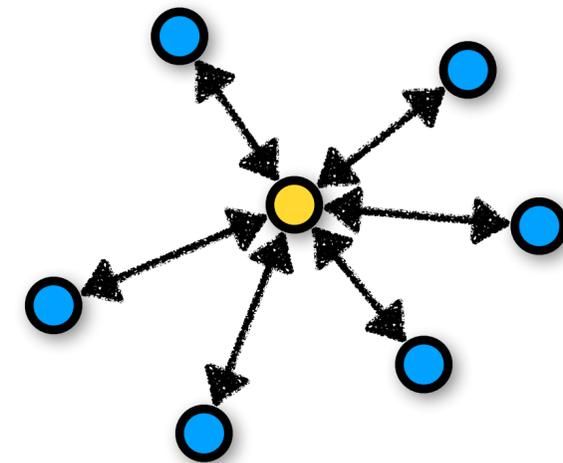
Finite elements



Finite Volumes



SPH

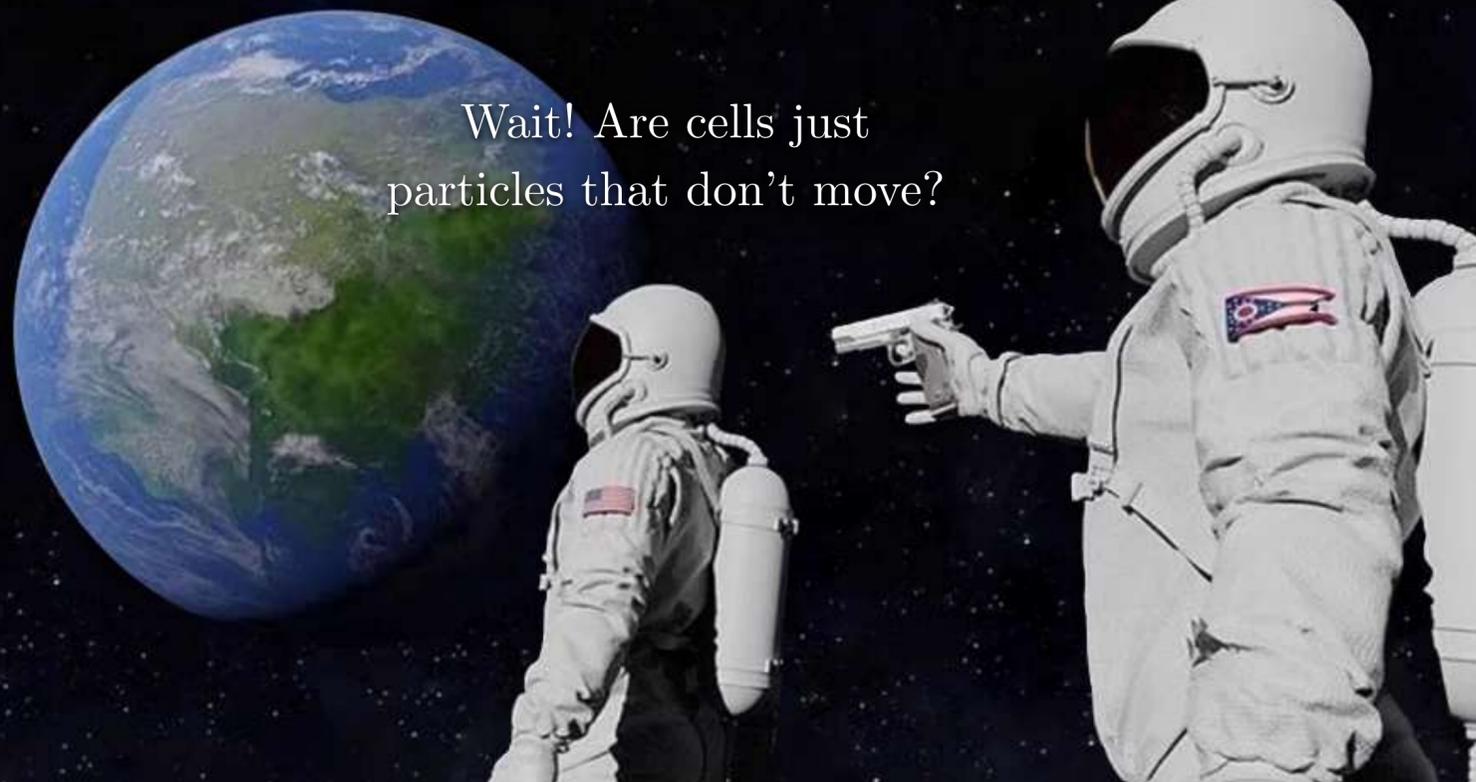


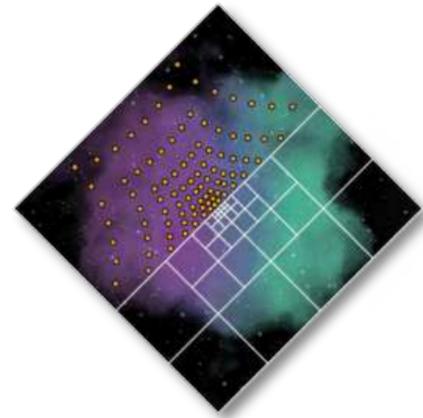
Always has been

Wait! Are cells just particles that don't move?

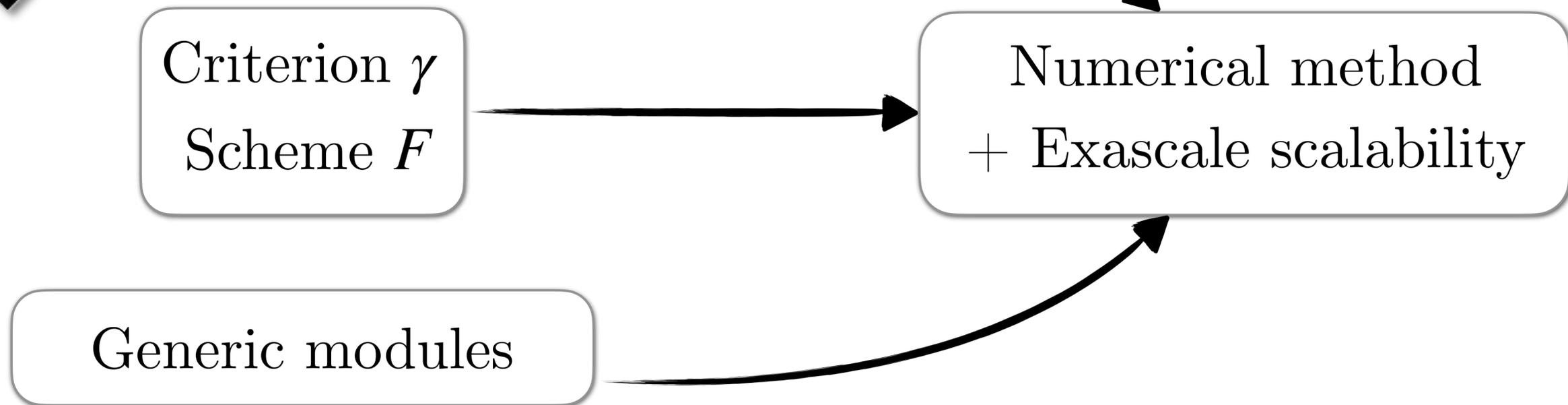
$$U_i^{t+1} = F \left(\{U_i^t\}_{i \in \{\bullet, \square\}} \right)$$

Numerical scheme





Shamrock

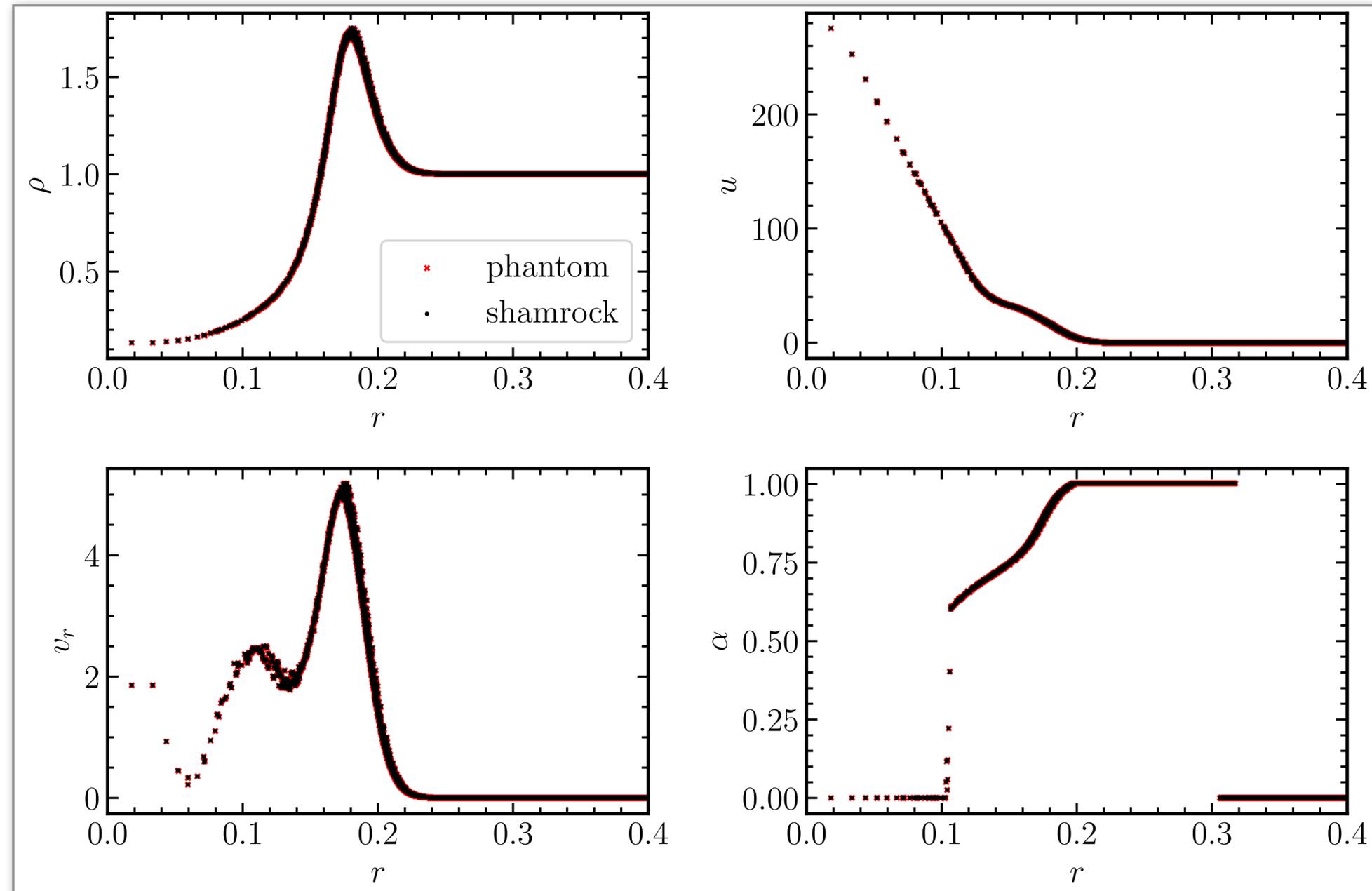


Optimising SPH \Leftrightarrow Optimising AMR

Generic modules are coded **once** for all schemes

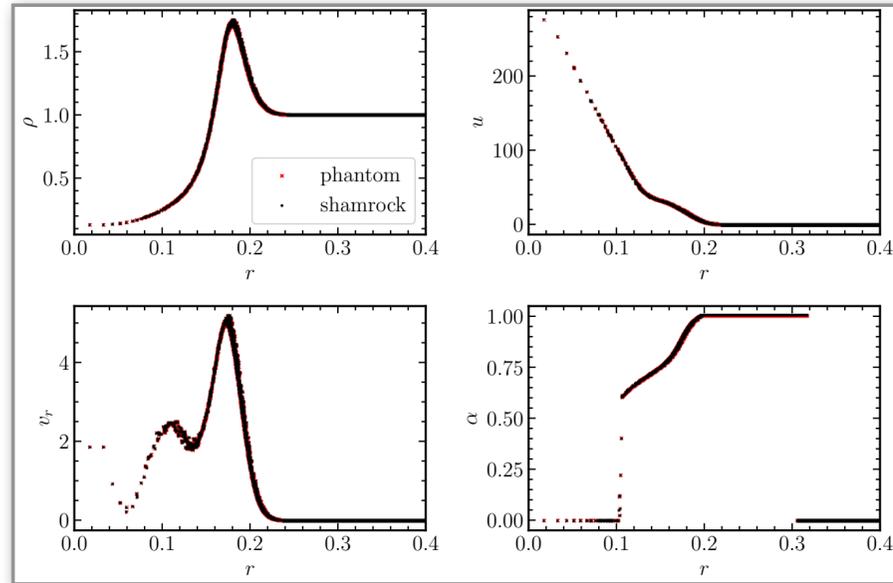
Reality is a bit more nuanced but that's the goal

Sedov-blast wave:

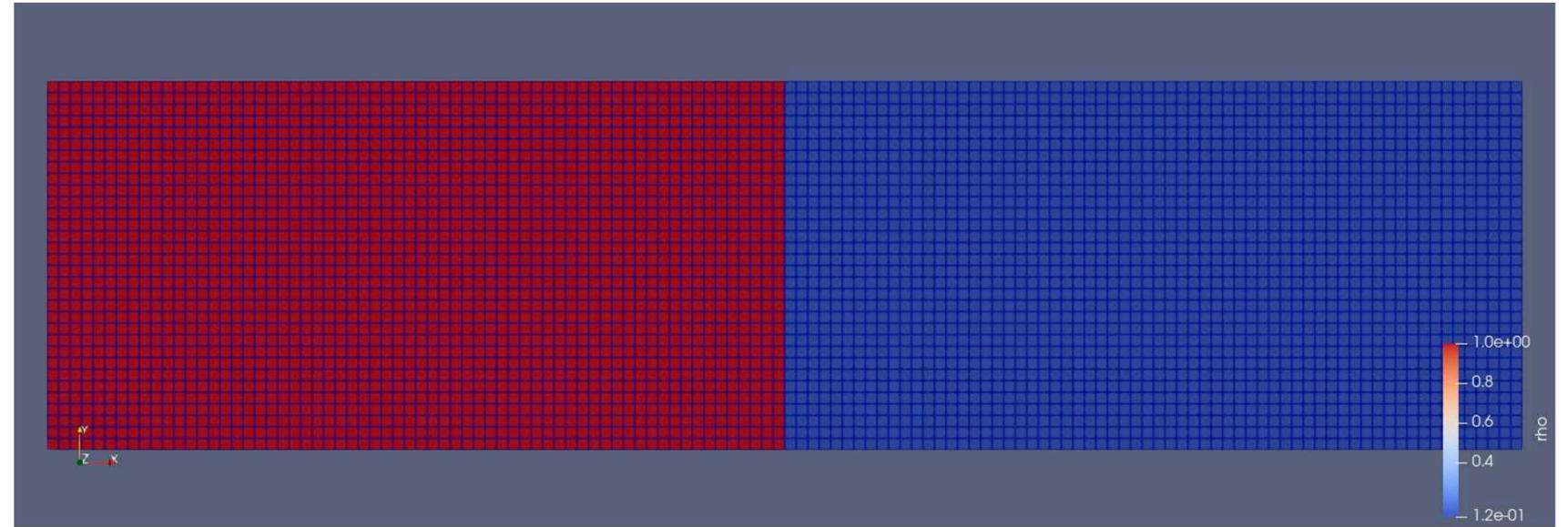


Reproduce Phantom solver !

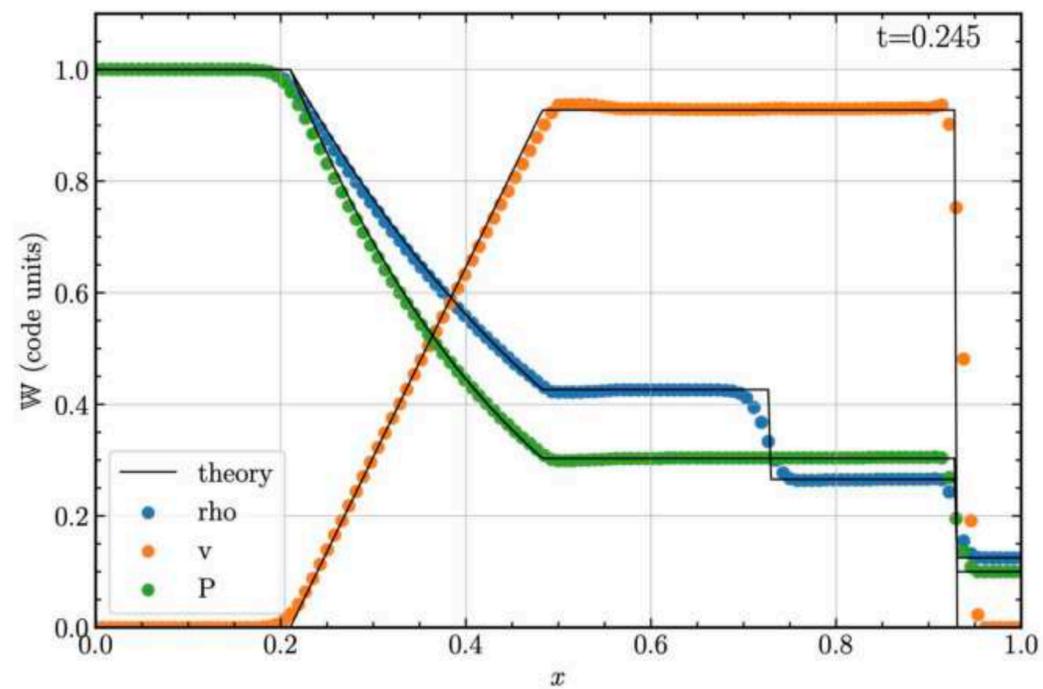
SPH (Phantom kind)



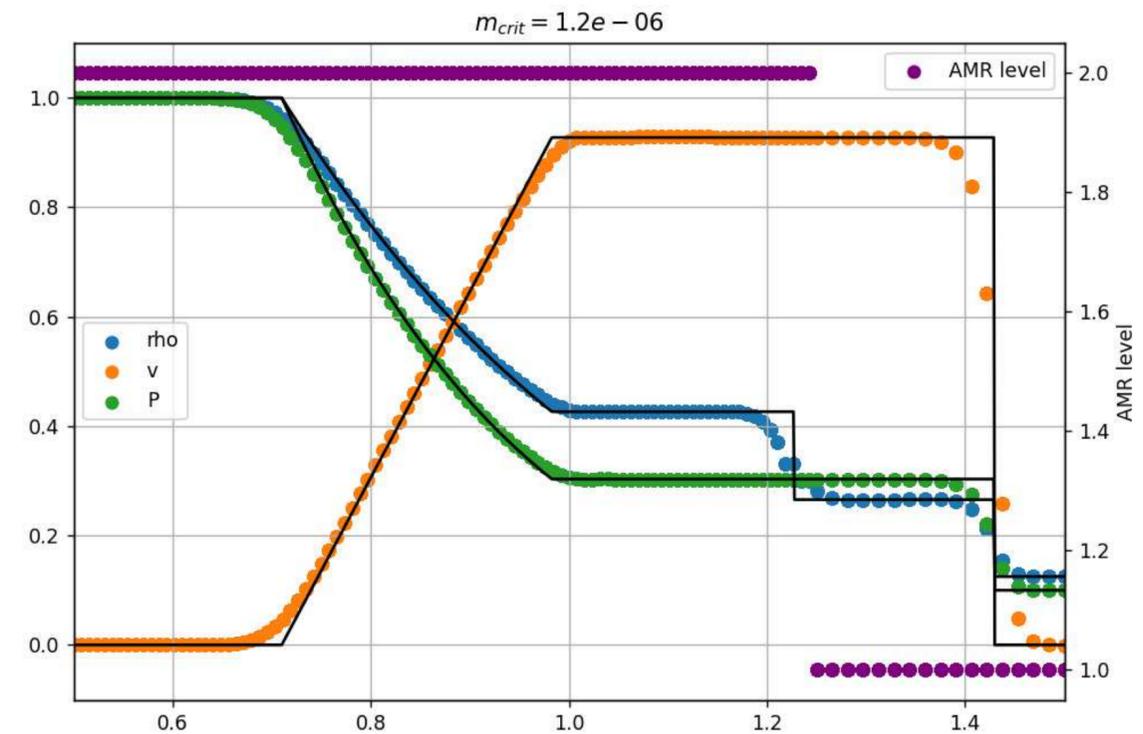
Finite volume (RAMSES kind)



Finite elements (Zeus kind)



Mass based refinement

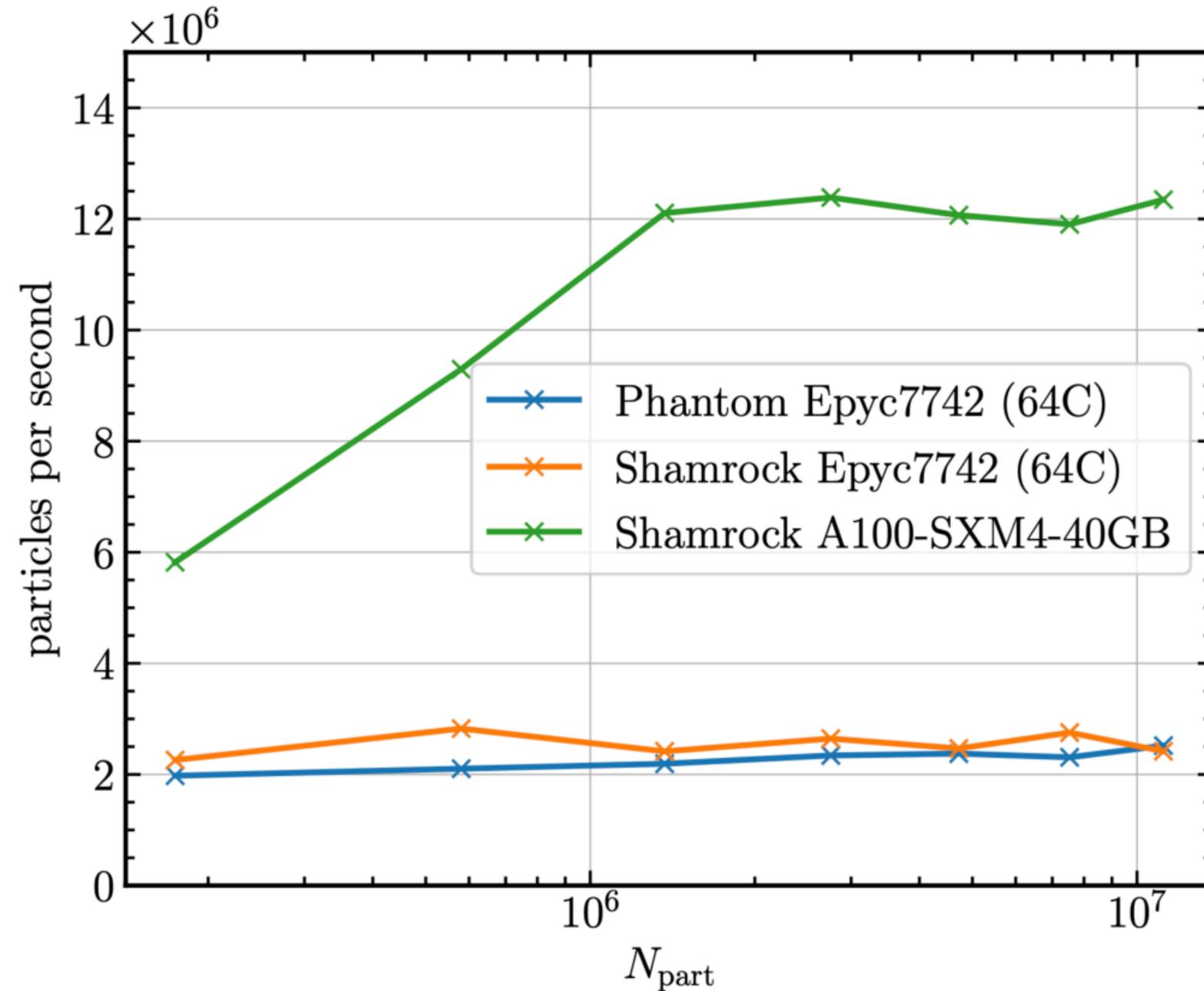


Performance !!!!

(Measured in part/sec)



Sedov-blast wave:

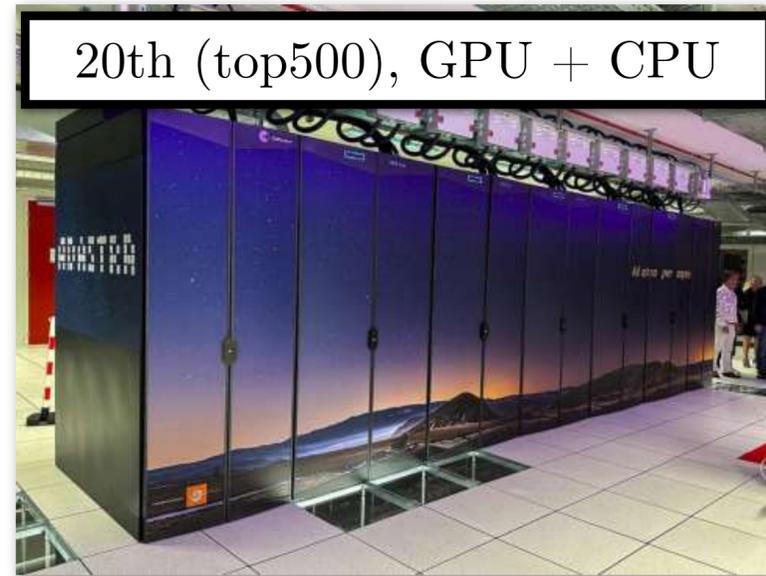
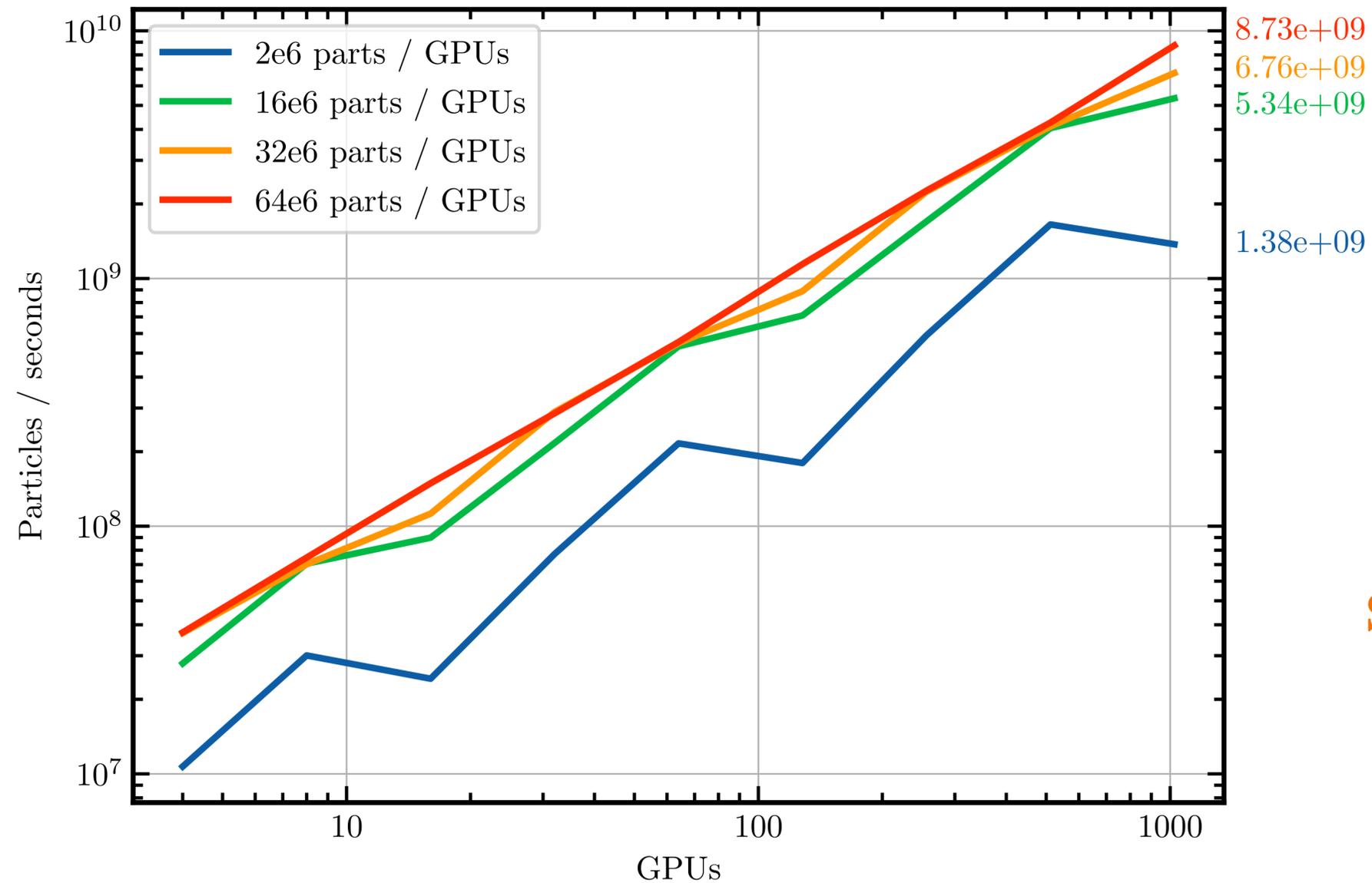


Latest benchmarks

$\approx 15\text{Mpart/sec}$

⇒ Acceleration by a factor 7/8 CPU ⇒ GPU

⇒ Same power consumption !!!



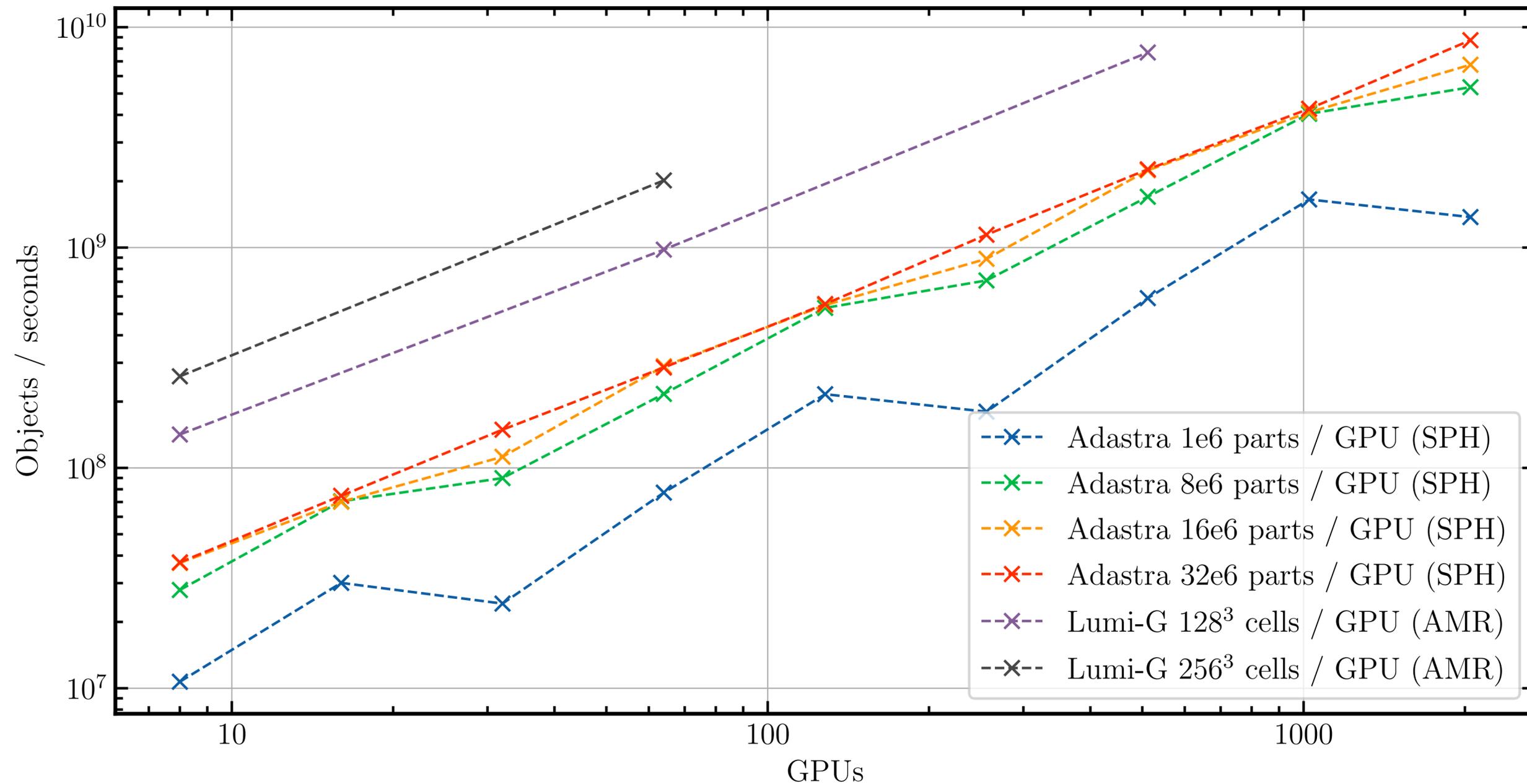
Shamrock :

- 65G particles (4000^3)
- 9G part / seconds
- 7 sec / iterations
- 1024 GPU (MI250x)
- 92% parallel efficiency

Reference :

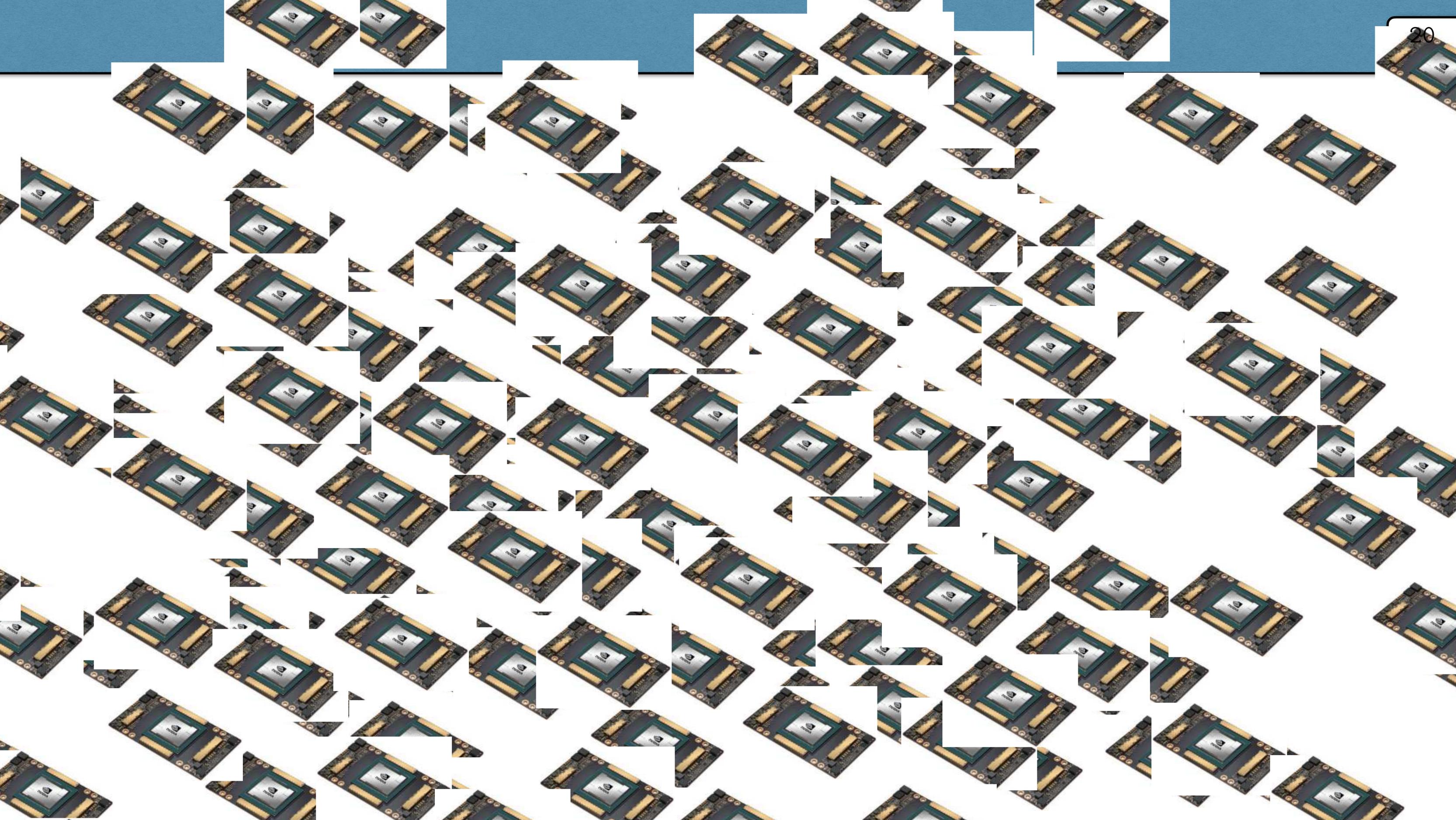
Same test with Phantom SPH = 2e6 part / seconds
→ x4500 speedup !





84% parallel efficiency on LUMI before optimisation

Next: Optimizing the solver to get at least a factor of 10





Top 500 list :

Rank	System	Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
1	El Capitan - HPE Cray EX255a, AMD Gen 5 EPYC 24C 1.8GHz, AMD Instinct MI300A, Slingshot-11, HPE DOE/NNSA/LLNL United States	11,039,616	1,742.00	2,746.38	29,581
2	Frontier - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE Cray OS, HPE DOE/SC/Oak Ridge National Laboratory United States	9,066,176	1,353.00	2,055.72	24,607
3	Aurora - HPE Cray EX - Intel Exascale Compute Blade, Xeon CPU Max 9470 52C 2.4GHz, Intel Data Center GPU Max, Slingshot-11, Intel DOE/SC/Argonne National Laboratory United States	9,264,128	1,012.00	1,980.01	38,698
4	Eagle - Microsoft NDv5, Xeon Platinum 8480C 48C 2GHz, NVIDIA H100, NVIDIA Infiniband NDR, Microsoft Azure Microsoft Azure United States	2,073,600	561.20	846.84	
5	HPC6 - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, RHEL 8.9, HPE Eni S.p.A. Italy	3,143,520	477.90	606.97	8,461

X Military

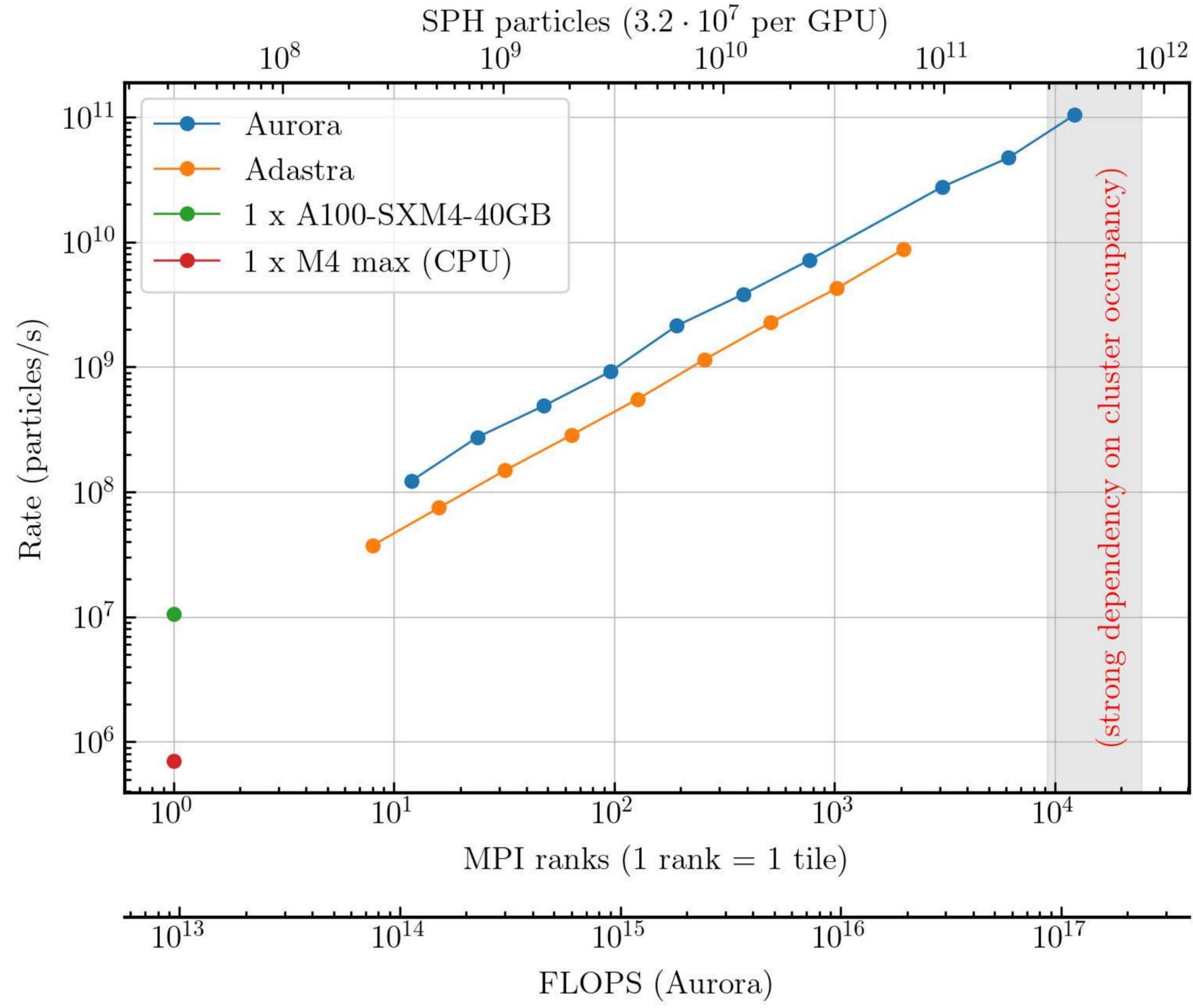
We have access to this one



120 000 GPUs

1 Exaflop

Note: getting GPU time is really easy



→ x50000 speedup !

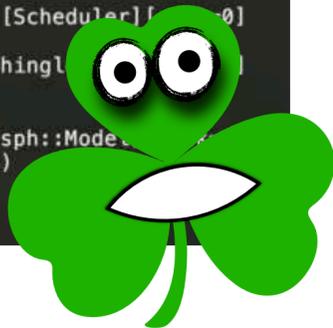
Shamrock :

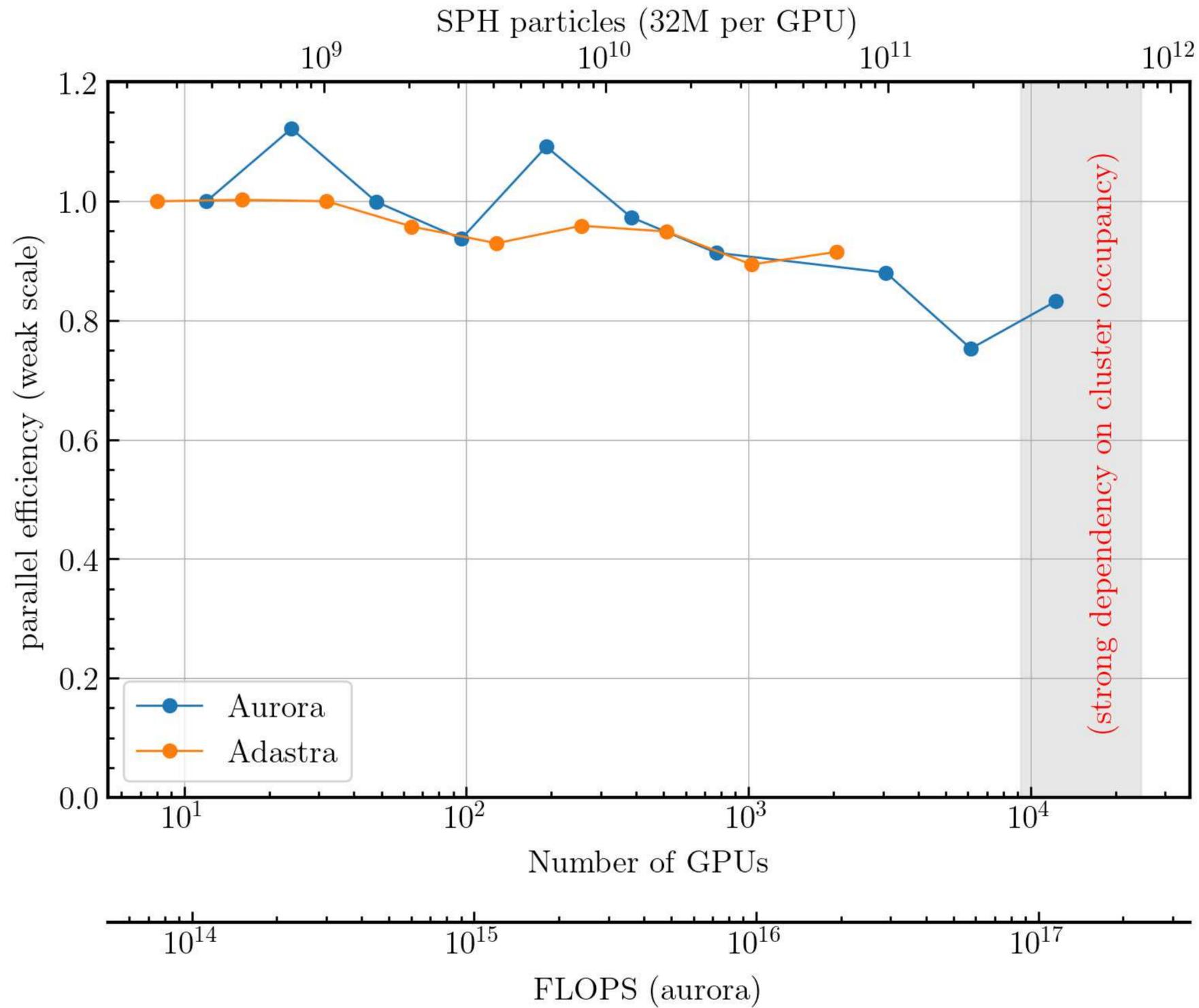
- 393.692.655.200 particles (7324^3)
- 10^{11} part / seconds
- 4 sec / iterations
- 12288 GPU (Intel PVC tiles)

```

12283 | 9.5624e+06 | 36057000 | 3.771e+00 | 28 % | 3 % | 32.00 GB
12284 | 9.5758e+06 | 36108000 | 3.771e+00 | 28 % | 3 % | 32.00 GB
12285 | 6.3777e+06 | 24046500 | 3.770e+00 | 28 % | 2 % | 32.00 GB
12286 | 9.5692e+06 | 36082500 | 3.771e+00 | 28 % | 3 % | 32.00 GB
12287 | 9.5609e+06 | 36057000 | 3.771e+00 | 28 % | 3 % | 32.00 GB
-----<sum N>/<max t> -----<sum> -----<max> -----<avg> -----<avg> -----<sum>
| all | 1.0438e+11 | 393692655200 | 3.772e+00 | 28 % | 2 % | 384.00 TB

Info: estimated rate : 2.6815576783253906e-09 (tsim/hr) [sph::Model][rank=0]
----- t = 5.59052345397294e-12, dt = 2.836977696765609e-12 -----
Info: summary : [LoadBalance][rank=0]
Info: - strategy "psweep" : max = 36196500 min = 23874120 [LoadBalance][rank=0]
Info: - strategy "round robin" : max = 36171000 min = 23874120 [LoadBalance][rank=0]
Info: Loadbalance stats : [LoadBalance][rank=0]
  npatch = 32768
  min = 23874120
  max = 36171000
  avg = 32038790.299479168
  efficiency = 66.00%
Info: Scheduler step timings : [Scheduler][rank=0]
  metadata sync : 126.96 ms (83.2%)
  patch tree reduce : 4.43 ms (2.9%)
  gen split merge : 1232.84 us (0.8%)
  split / merge op : 0/0
  apply split merge : 1523.00 ns (0.0%)
  LB compute : 8.33 ms (5.5%)
  LB move op cnt : 0
  LB apply : 3.17 us (0.0%)
Info: Scheduler step timings : [Scheduler][rank=0]
  metadata sync : 123.01 ms (98.3%)
Info: smoothing length iteration converged [Smoothing][rank=0]
  eps min = 0, max = 5.511837432562038e-16
  iterations = 0
Info: conservation infos : [sph::Model][rank=0]
  sum v = (6.240937604346087e-26,-6.240937604346087e-26,1.5602344010865217e-26)
  sum a = (0,0,-2.1443698325634424e-15)
  sum e = 0.9999999999692291
  sum de = -5.551115123125783e-15
    
```



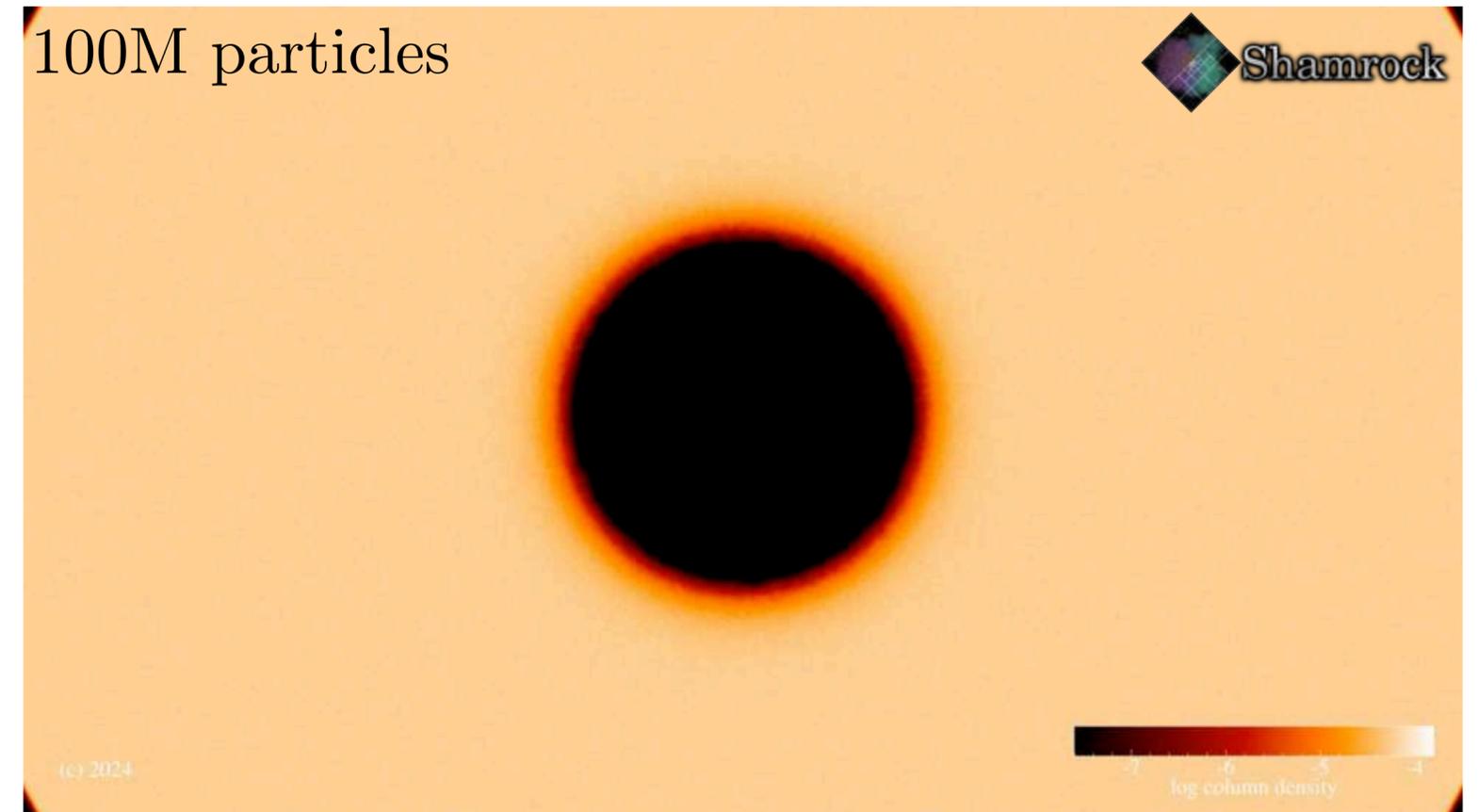
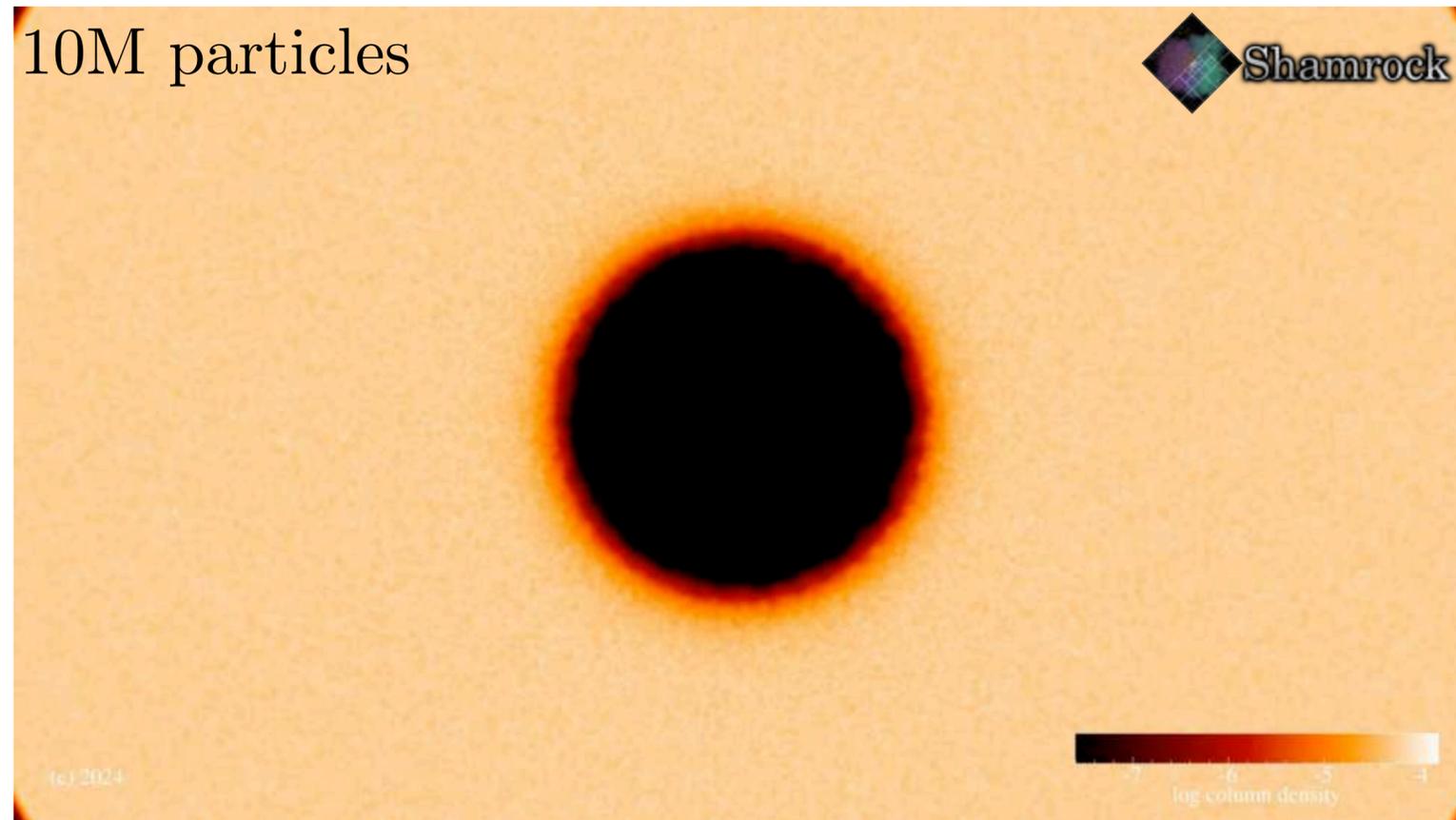


Next step is Exascale !

Exemple of a simulation

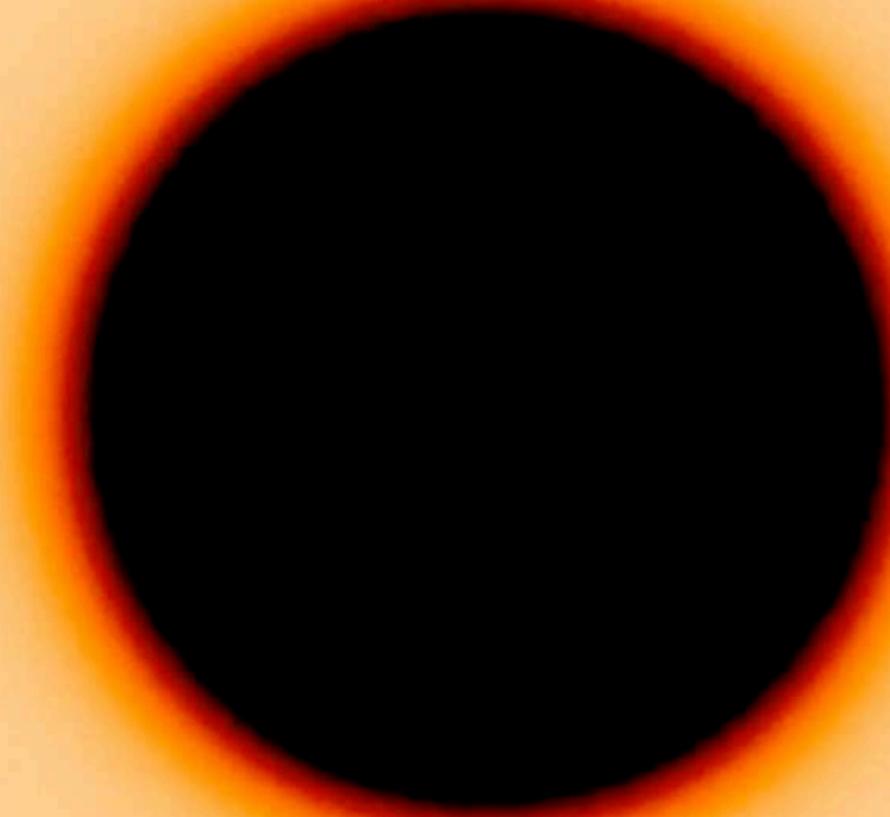


- Soft limit around 10-100M particles
But new physics at 1G particles

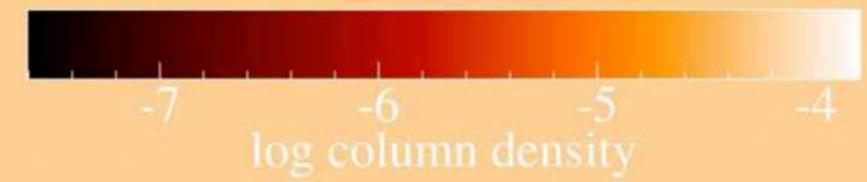


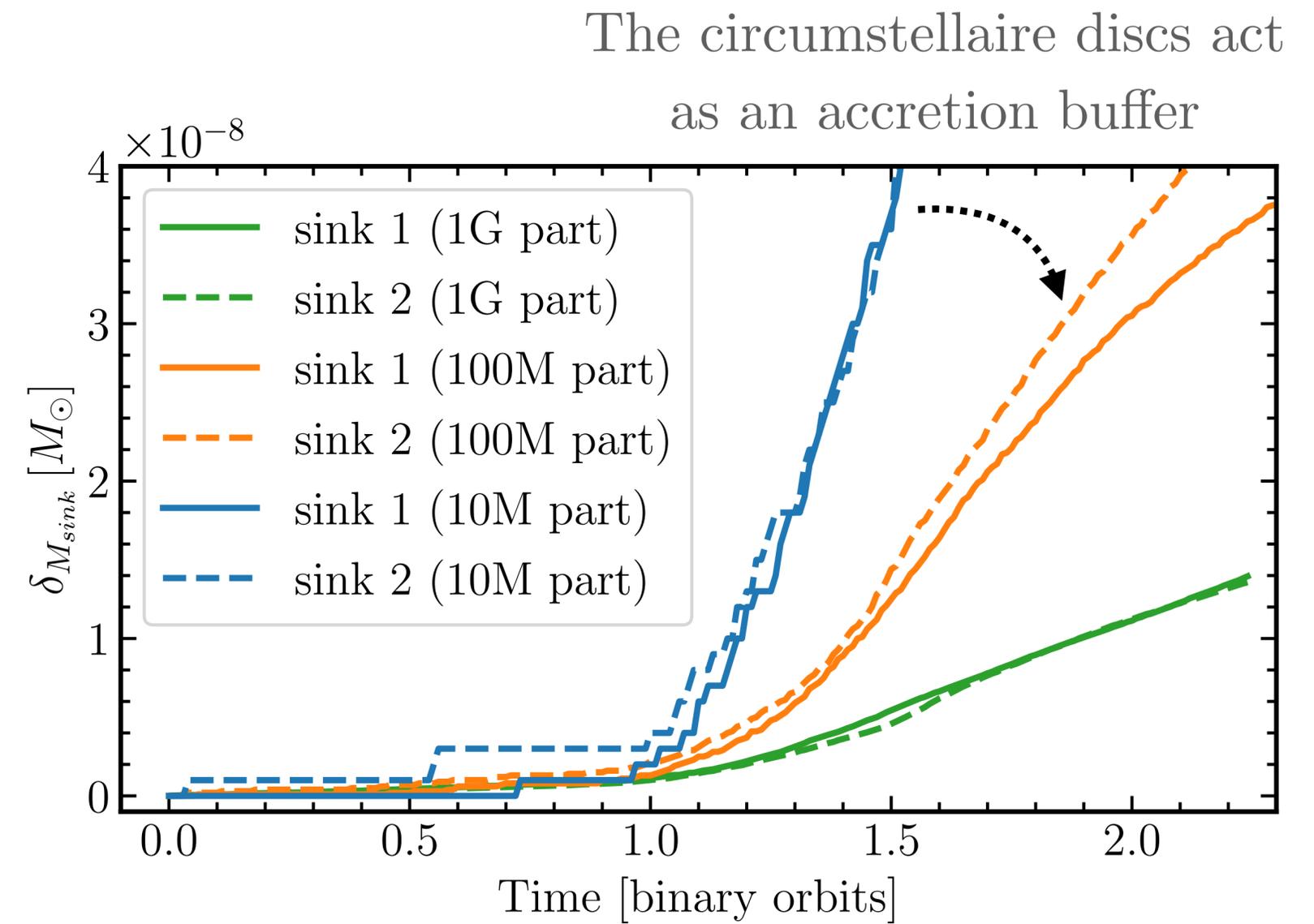
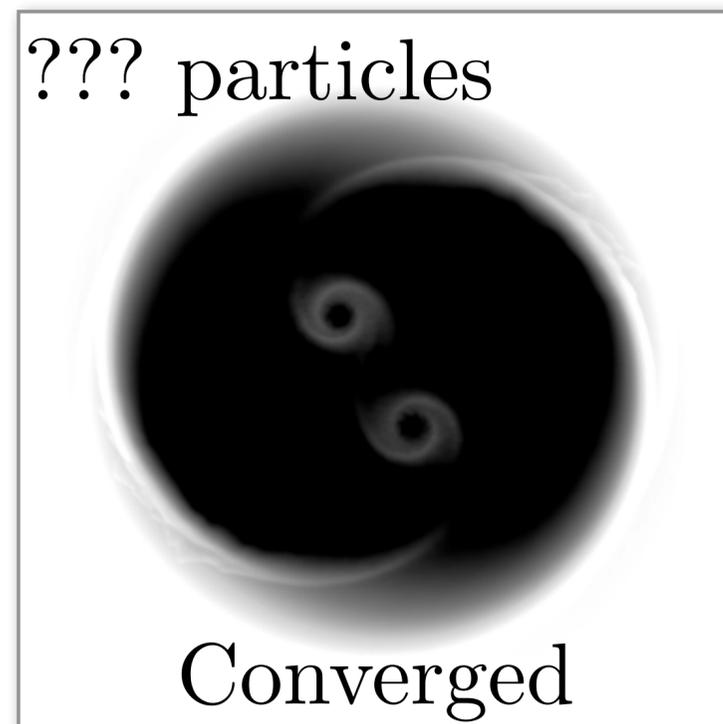
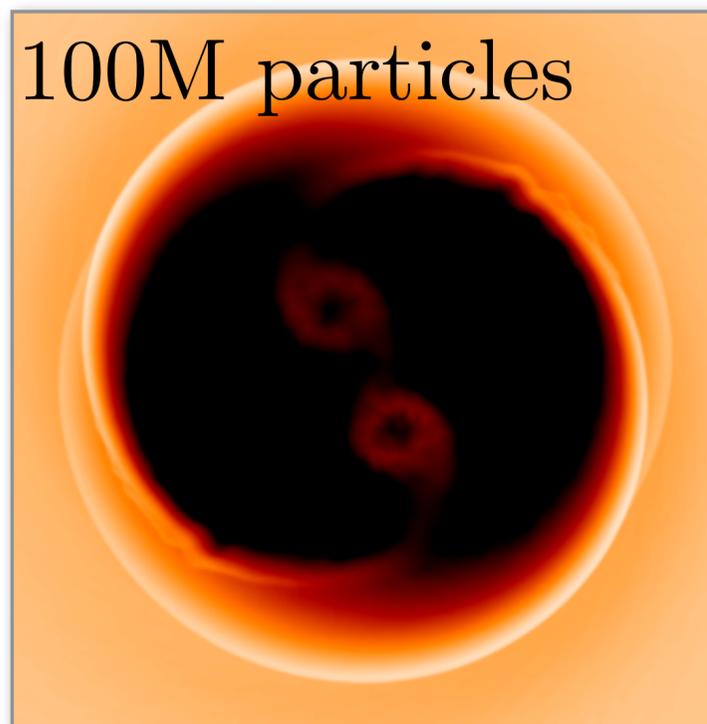
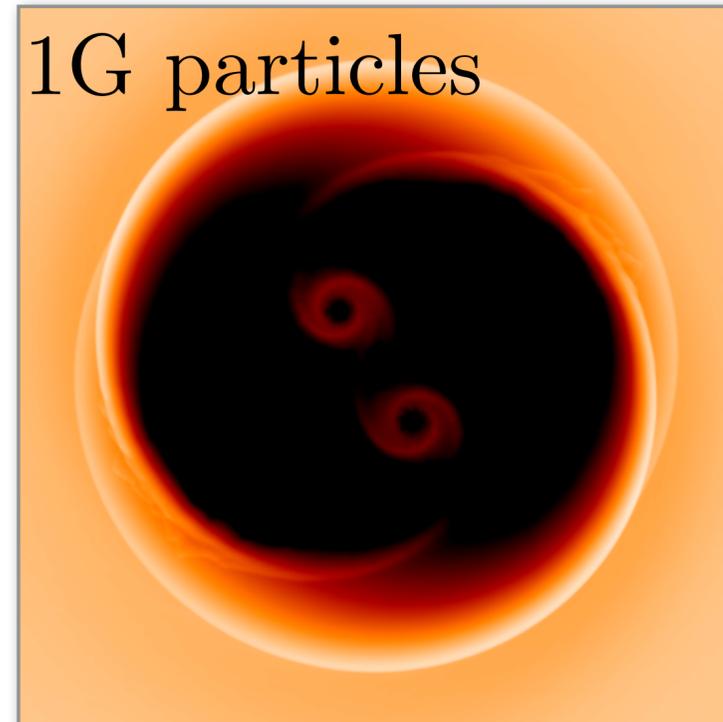
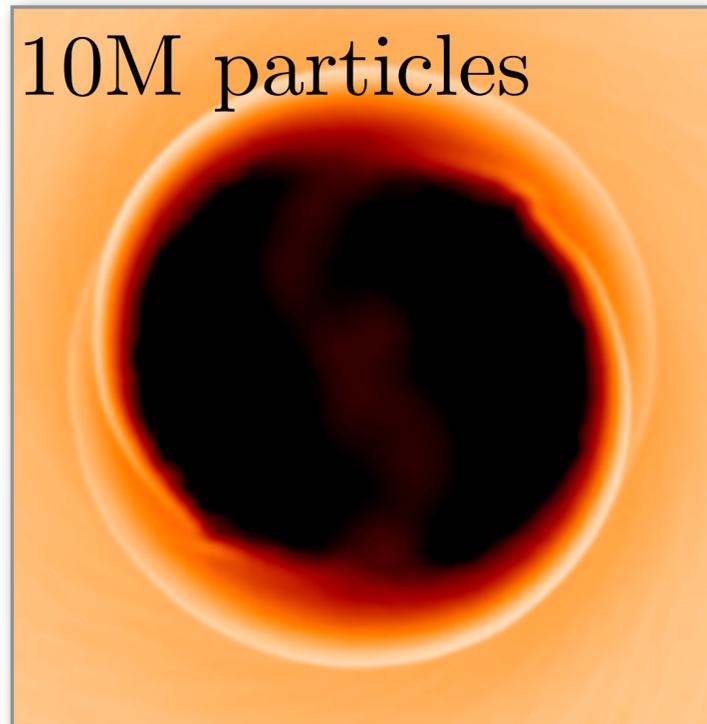
Need more resolution !

1G particles

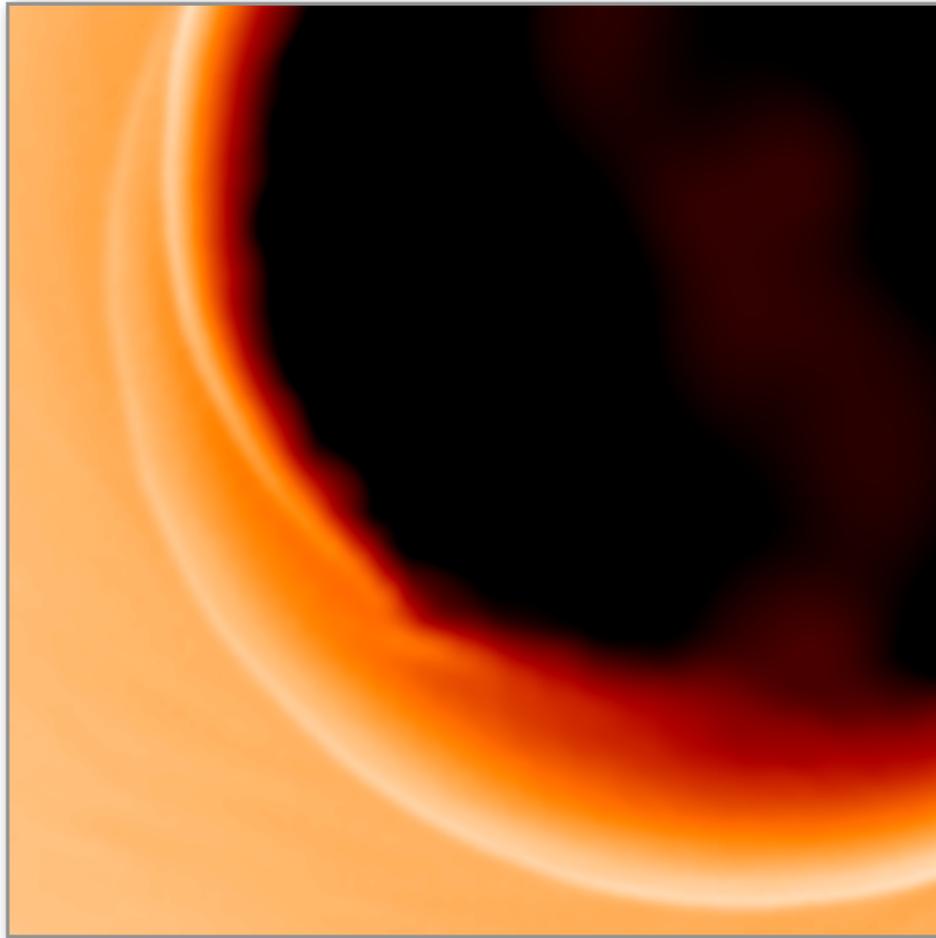


(c) 2024

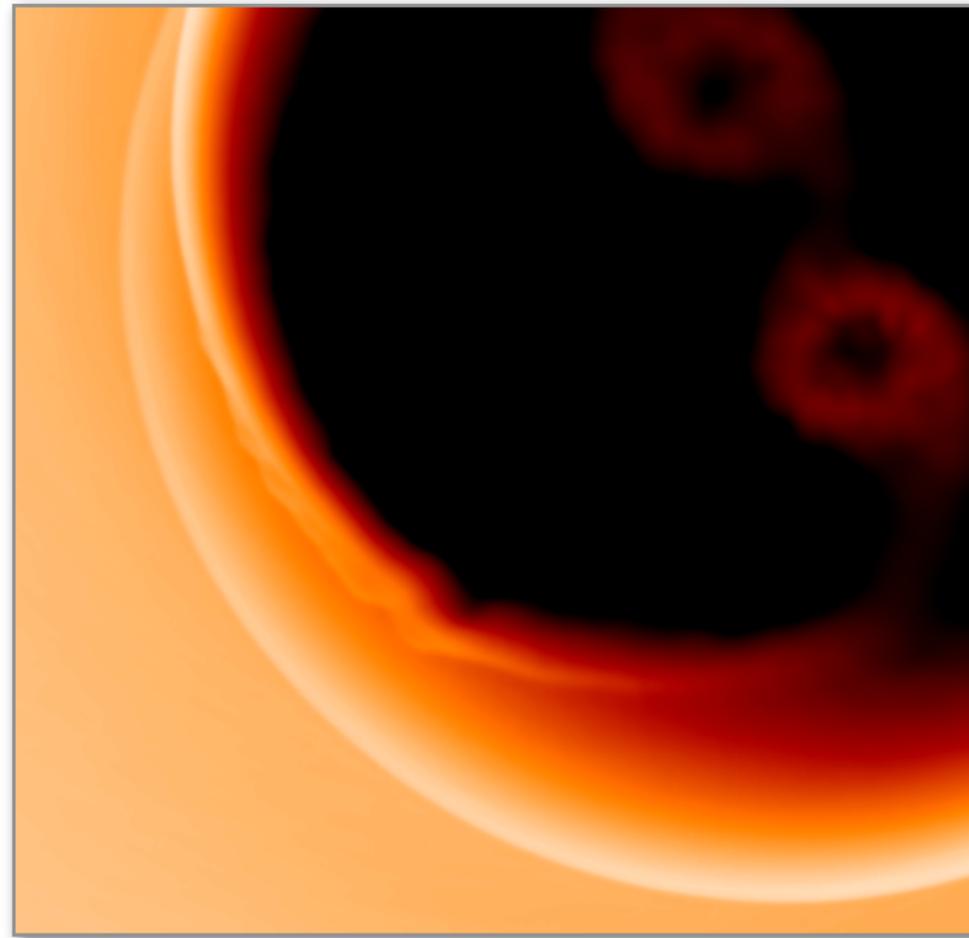




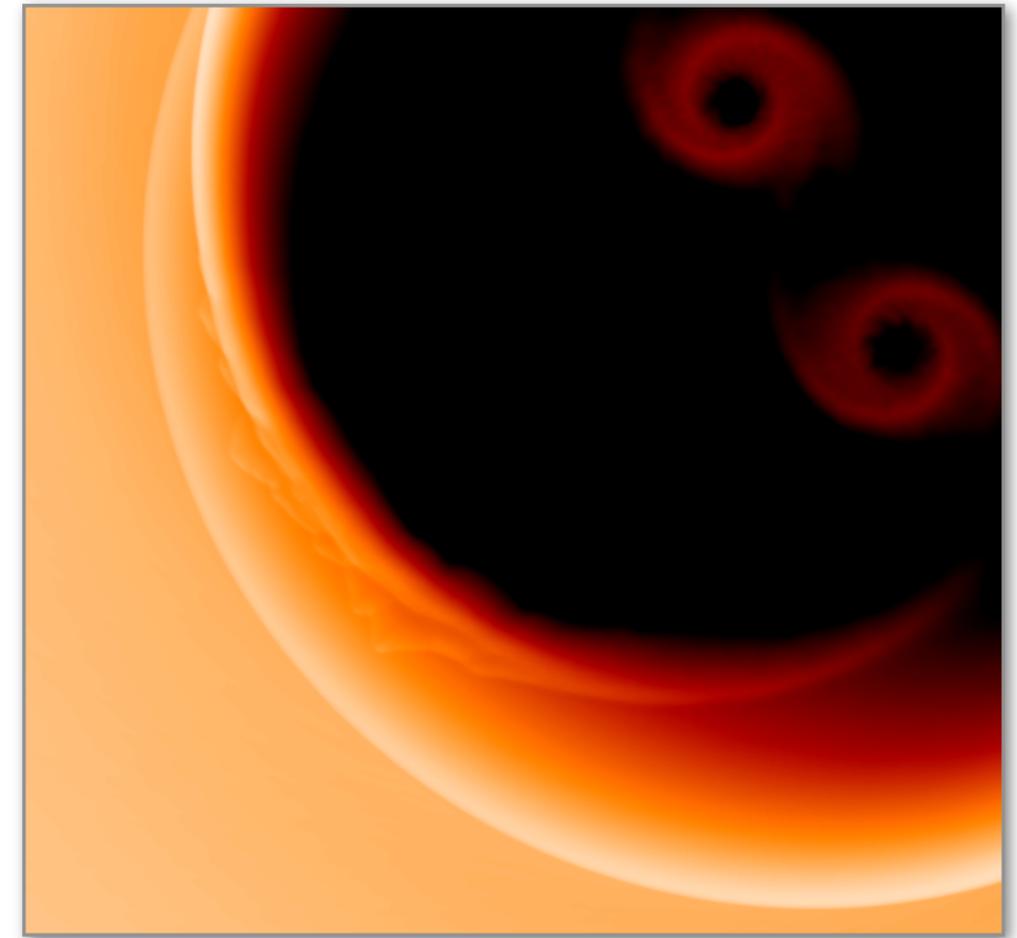
10M particles



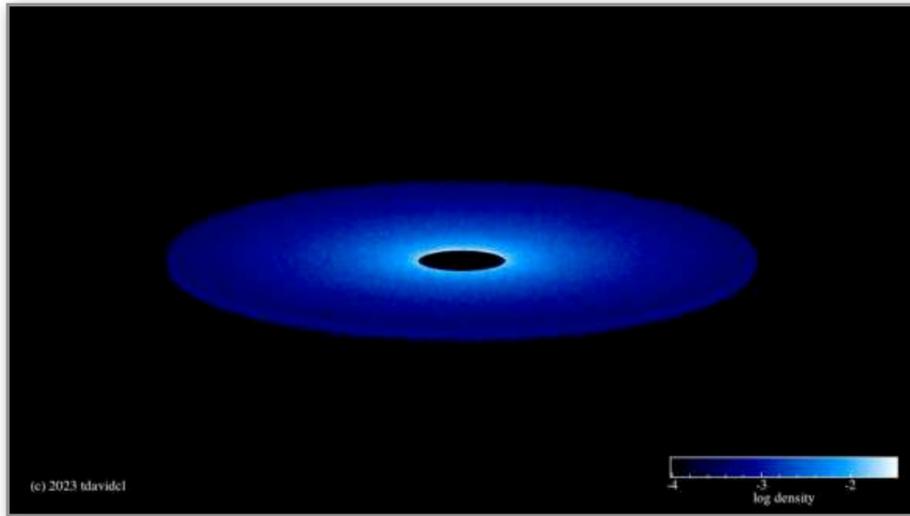
100M particles



1G particles



1G particules : 125 particules / H \Rightarrow Enough for VSI, SI, parametric instability, ...



See Yona's talk
for discs



Where do we go from there ????

See guillaume talk



SPH solver (production ready) :

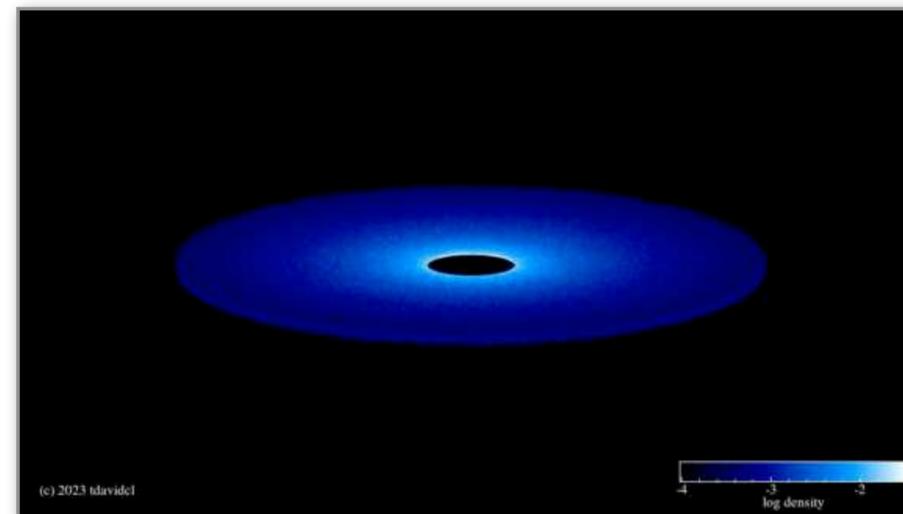
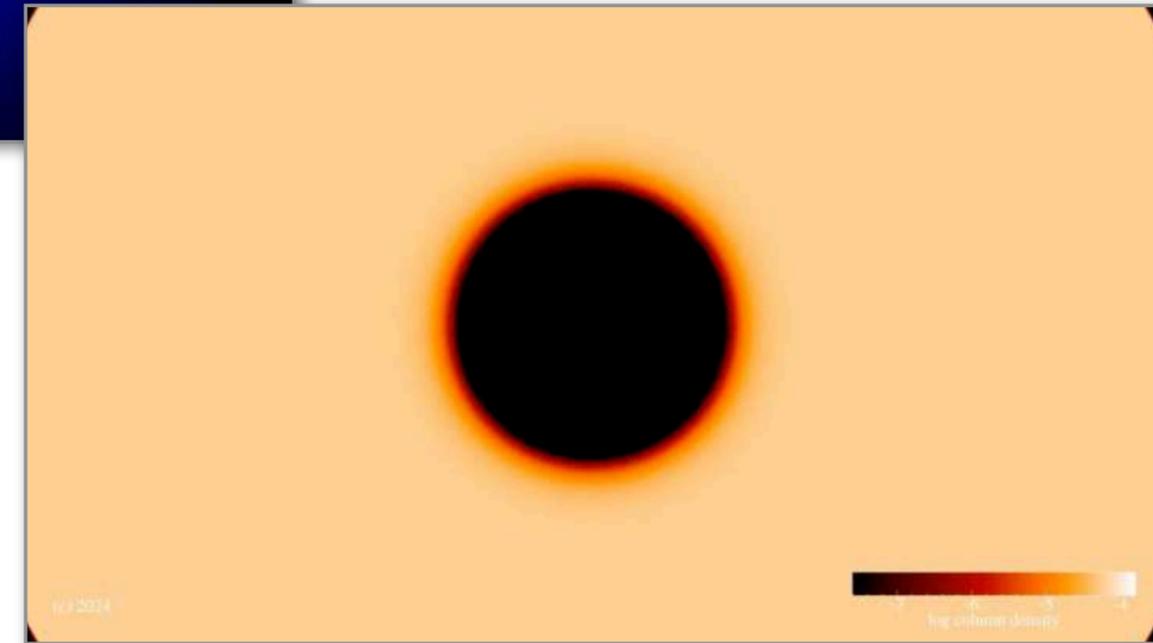
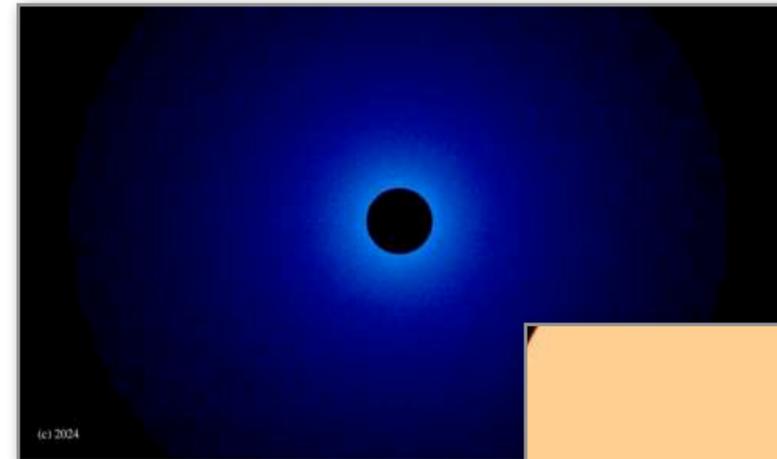
- Hydro ✓
- Discs ✓
- MHD ✓ (Y. Lapeyre)
- Black hole + warp ✓

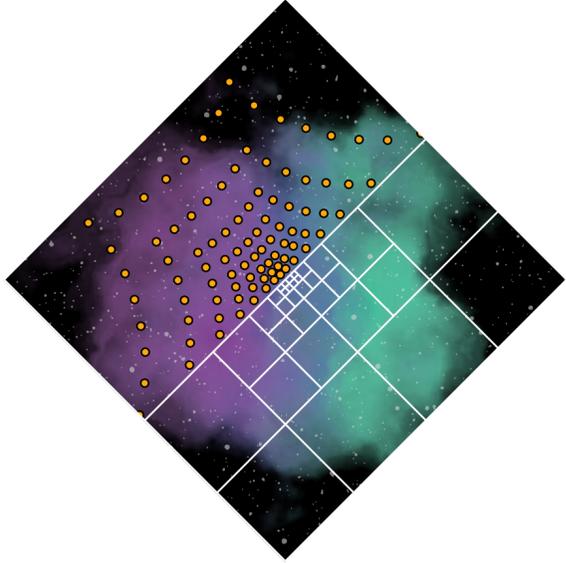
AMR solver (WIP) :

- Hydro ✓
- Dust ✓
- Self-gravity (WIP)

Next :

- Exascale disc simulations
- Non local forces (self-gravity)
- Live MCMC radiative transfert





Shamrock

