

Parallelizing Multi-Physics SPH in Phantom

Terrence Tricco

Memorial University of Newfoundland



Do you use MPI in Phantom?

Need for Larger Simulations

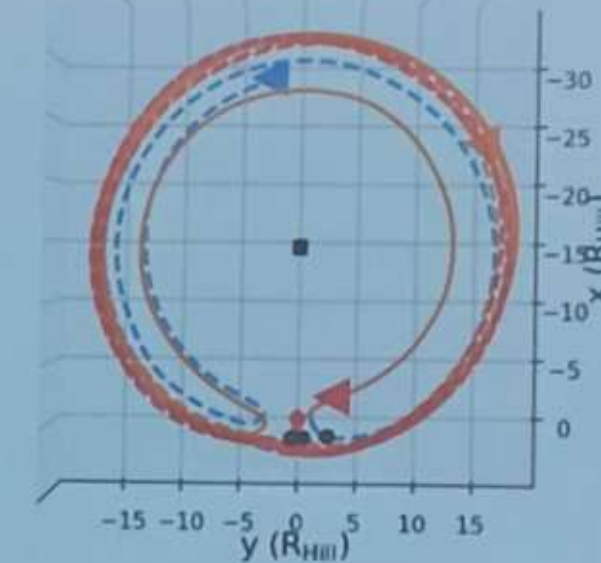
George Blaylock-Squibbs

What's next?

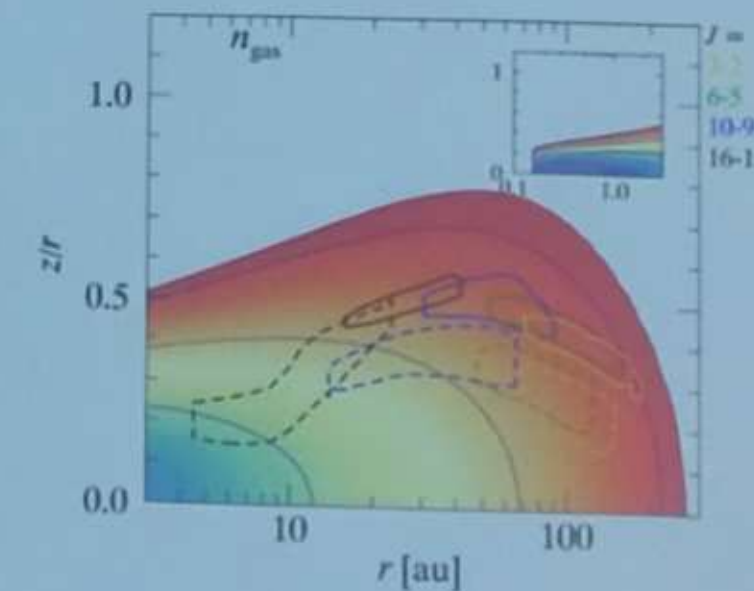
The pipeline works, now to make it "right"
i.e. Realistic chemical network

Use APR to get better resolution around dense clumps of particles.

Compare to current work looking at the chemical processing around circumplanetary disks in CA formation.



Cridland + (2025)



Fedele + (2016)

Need for Larger Simulations

Ethan Carter

Future Work

- Perform a statistical analysis of the protoplanet properties (i.e. aspect ratios, masses).
- Investigate the effect of varying the metallicity on the properties of disc-instability protoplanets.
- Need a high number of particles to study the fragments in detail – highly interested in adaptive particle refinement and how it may be applied!

Need for Larger Simulations

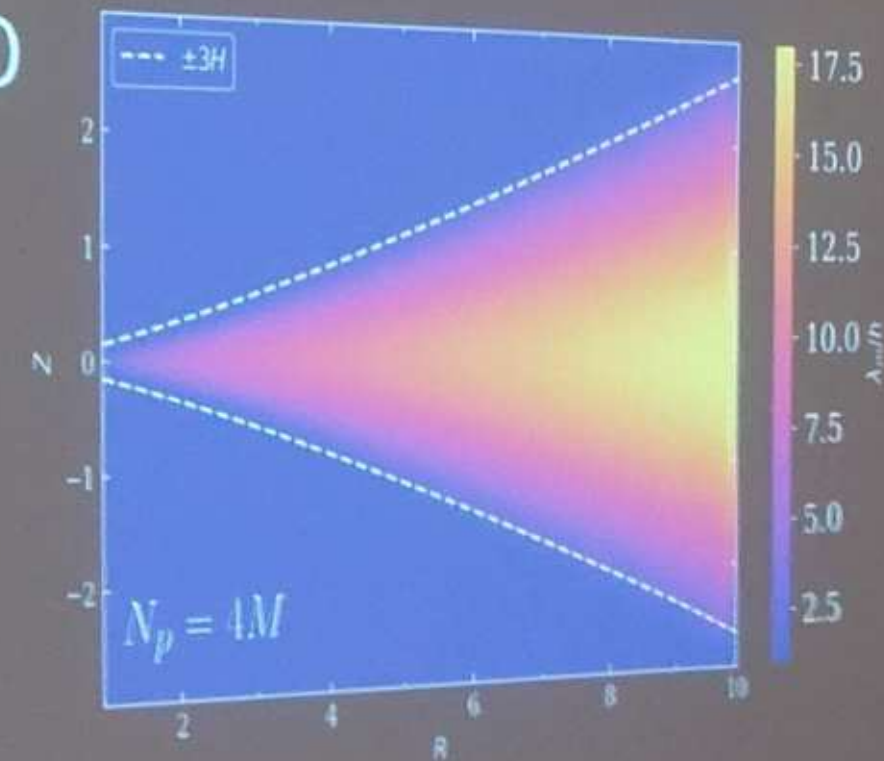
Jackson Narvaez-Coral

- In SPH, MRI has not been activated yet
- Critical challenges for Global Disc MHD Simulations:

- Dissipation
- Divergence cleaning

- Future Work:

- Study energy loss due to artificial resistivity, divergence cleaning,
- Improve resolution: Include adaptive particle refinement (APR)



Need for Larger Simulations

Claudia Toci

Future perspectives

- Increase the resolution: SHAMROCK and APR
(e.g., testing GG Tau with increased resolution)
- Accurate temperature layers: PHANTOM + MCFOST
(e.g., testing GG Tau with different temperature structures)
- Chemical complexity: PHANTOM + KROME
(e.g., testing molecular emission of GG Tau)

Need for Larger Simulations

Mike Lau

Wrapping up: Room for improvement

Recombination energy

- Add recombination energy, making substantial progress towards settling the debate on its relevance

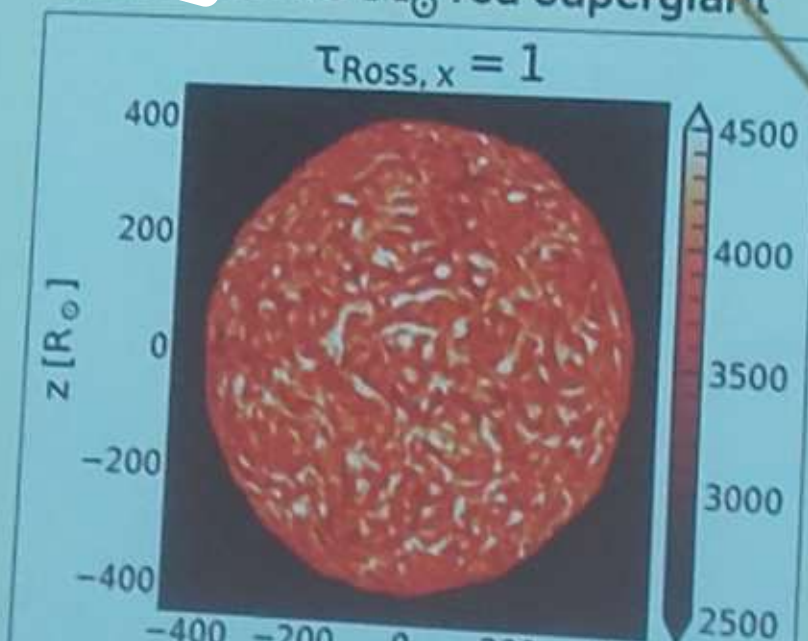
Sabach+ 2017, Grichener+ 2018, Ivanova 2018, Soker+18

- Cannot use MESA EoS tables directly, because radiation energy must be separated out from the total internal energy
- Current progress: Use Ryo Hirai's fits analytical treatment of ionisation physics (gas + radiation + recombination EoS, $i_{\text{eos}}=20$), including new fits of c_V accounting for rotational and vibrational degrees of freedom of H_2

Convection and optically thin radiation transport

- A realistic 3D giant star must have convection driven by photospheric cooling
- Photospheric cooling is not correctly captured due to the unresolved photosphere and an initial lack of SPH particles above the photosphere to radiate into

Ma 2025: $10 M_{\odot}$ red supergiant

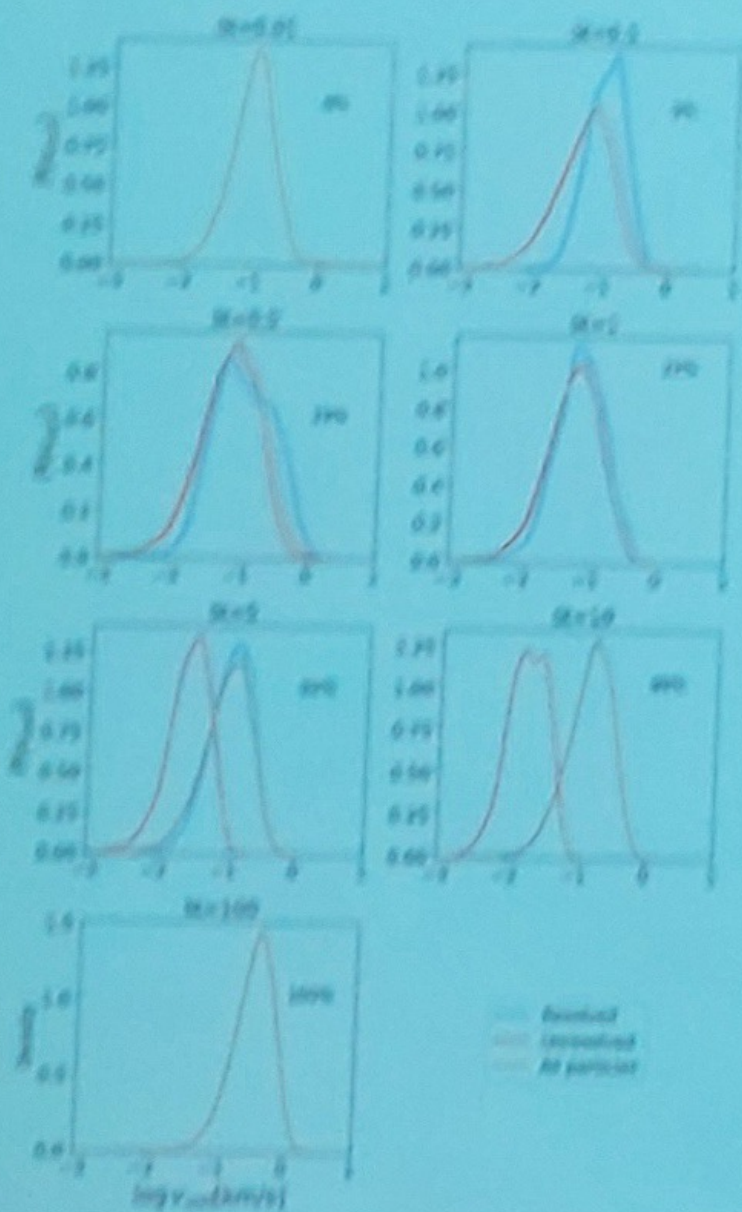


Full 4π er
convection
resolved
in AREPO
Implicit D

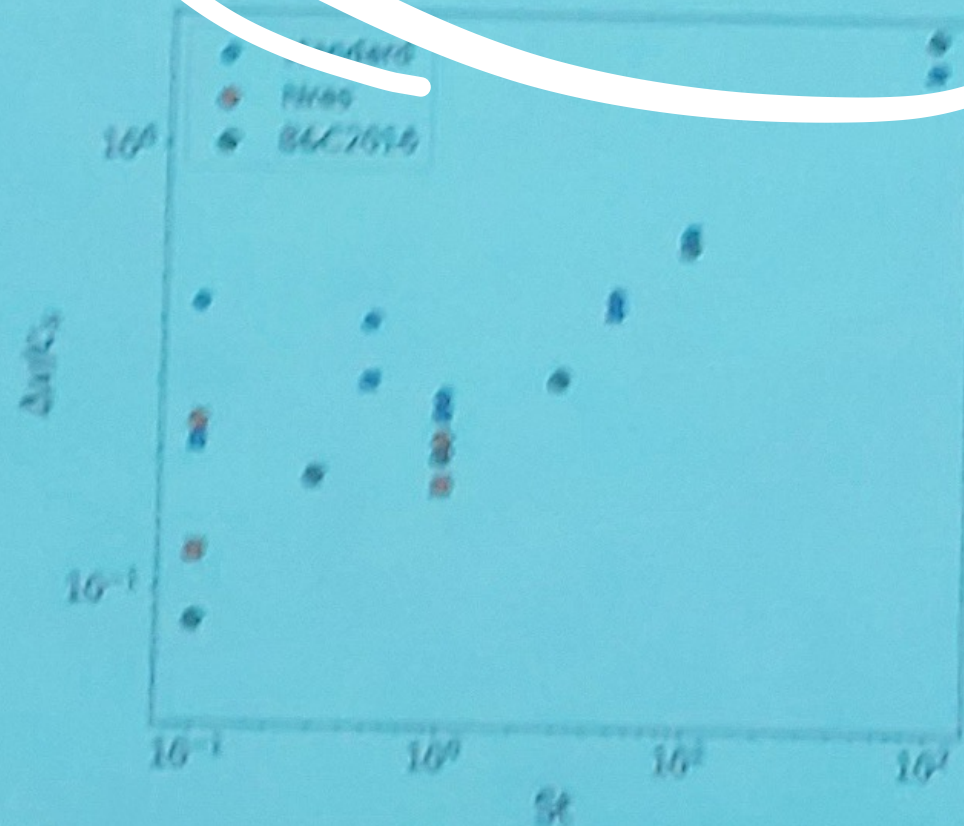
Need for Larger Simulations

Cristiano Longarini

GI and cooling: planet formation



$\Delta R_{1,2} < t_s \Delta v_{1,2}$
Very high resolution
is required to resolve
collisions



Longarini et al. in prep

There is a clear need to run higher resolution simulations.

What is possible with Exascale

Guillaume Laibe

Resolution

e.g. resolve the radius of a planet in a disc simulations

Integration

e.g. form a planet after the protostar

Statistics

e.g. 1 simulation = all discs ever generated with RAMSES

Environnement

e.g. run tens of objets in interaction

To Higher Resolution

- How can we reach higher resolutions?
 1. Algorithmic optimizations.
 2. More compute.

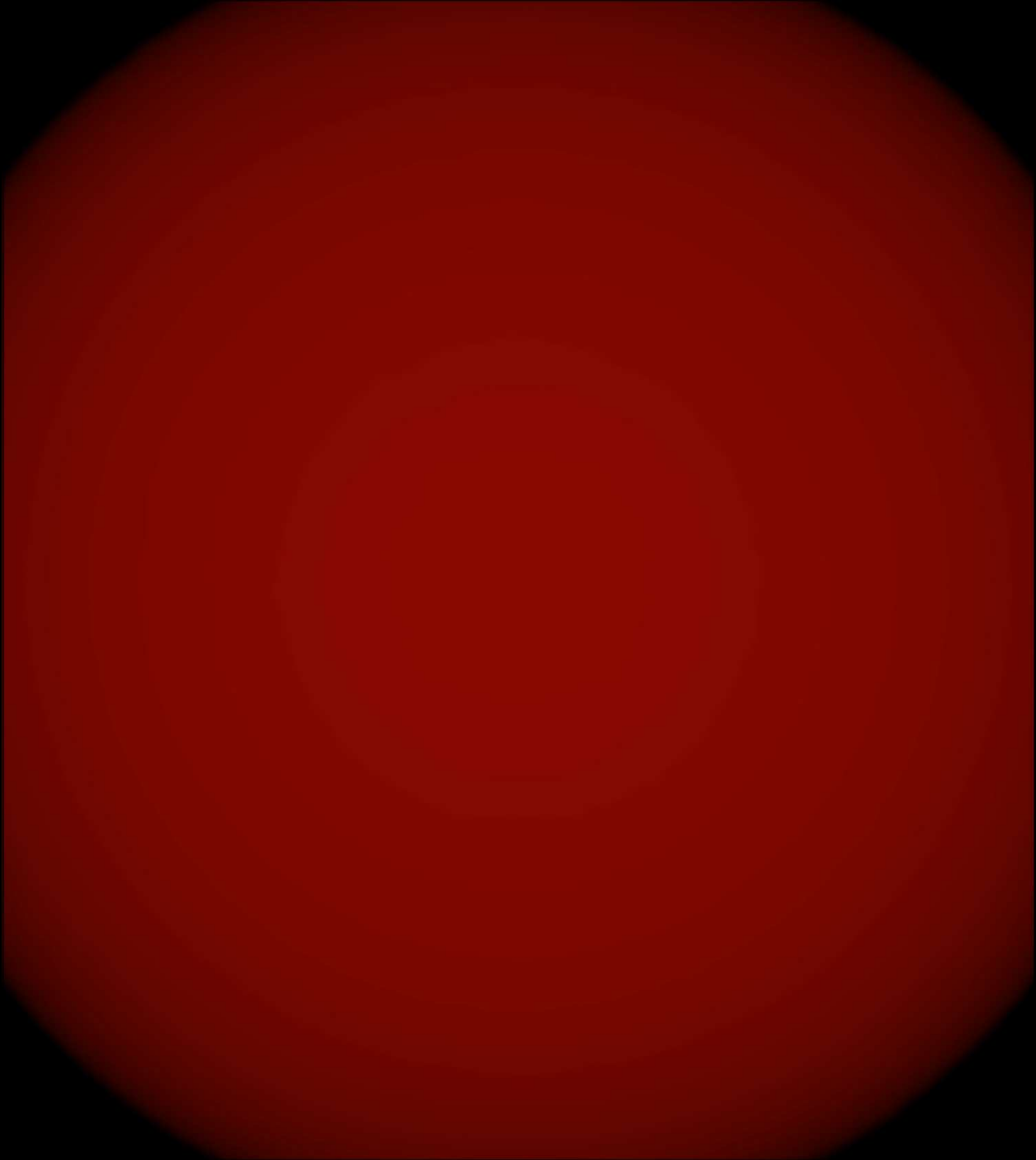
To Higher Resolution

- How can we reach higher resolutions?
 1. Algorithmic optimizations.
 - Adaptive particle refinement (APR) – ~6x speedup.
 - Individual particle timesteps – 100-1000x speedup.
 2. More compute.

To Higher Resolution

- How can we reach higher resolutions?
 1. Algorithmic optimizations.
 - Adaptive particle refinement (APR) – ~6x speedup.
 - Individual particle timesteps – 100-1000x speedup.
 2. More compute.
 - OpenMP – ~48-64x speedup.
 - MPI – ~2-1000x speedup?
 - GPUs – ~10x speedup?

How good is Phantom's MPI scaling?



MPI in Phantom

64 cores (OpenMP)

```
----->  TIME =      8.275      : full dump written to file jet1m_00097  <-----
```

input file jet1m.in written successfully.
Since code start: 746 timesteps, wall: 956s cpu: 5.09E+04s cpu/wall: 53
Since last dump : 8 timesteps, wall: 12s cpu: 692s cpu/wall: 60

		wall	cpu	cpu/wall	load bal	frac
└step	:	11.62s	691.92s	59.52	100.00%	92.08%
└└tree	:	2.88s	135.36s	47.08	100.00%	22.77%
└└└density	:	2.50s	163.60s	65.44	100.00%	19.80%
└└└└local	:	2.50s	161.29s	64.51	100.00%	19.80%
└└└force	:	5.75s	370.66s	64.46	100.00%	45.54%
└└└└local	:	4.62s	292.27s	63.19	100.00%	36.63%
└└cons2prim	:	0.12s	5.43s	43.41	100.00%	0.99%
└write_dump	:	1.00s	0.98s	0.98	100.00%	7.92%

```
** Number of steps since last summary:      8      **|
** Wall time since last summary:      11.625 seconds  **|
```

```
* particles woken *|
| #steps | mean # part/step | max # part/step |
|      4 |          13.00 |          17      |
```

bin	dt	npart	frac	cpufrac
0	8.886E-02	0	0.00%	20.38%
1	4.443E-02	572589	37.45%	19.05%
2	2.221E-02	666016	43.56%	26.99%
3	1.111E-02	290346	18.99%	33.58%

0.51M particles per second, IND TIMESTEPS efficiency: 75.48%

128 cores (2 MPI nodes + OpenMP)

```
----->  TIME =      8.264      : full dump written to file jet1m-2n_00093  <-----
```

input file jet1m-2n.in written successfully.
Since code start: 744 timesteps, wall: 3640s cpu: 3.61E+05s cpu/wall: 99
Since last dump : 8 timesteps, wall: 50s cpu: 4983s cpu/wall: 100

		wall	cpu	cpu/wall	load bal	frac
└step	:	50.00s	4982.64s	99.65	99.84%	96.15%
└└tree	:	9.88s	224.62s	22.75	88.00%	18.99%
└└└balance	:	7.25s	14.16s	1.95	100.00%	13.94%
└└└density	:	27.25s	3259.12s	119.60	99.94%	52.40%
└└└└local	:	7.12s	814.16s	114.27	95.52%	13.70%
└└└└remote	:	20.12s	2434.36s	120.96	98.39%	38.70%
└└└force	:	12.38s	1464.86s	118.37	99.29%	23.80%
└└└└local	:	6.88s	815.92s	118.68	89.08%	13.22%
└└└└remote	:	4.62s	573.53s	124.01	86.59%	8.89%
└└cons2prim	:	0.25s	6.25s	25.00	81.82%	0.48%
└write_dump	:	2.00s	2.31s	1.16	43.69%	3.85%

```
** Number of steps since last summary:      8      **|
** Wall time since last summary:      50.000 seconds  **|
```

```
* particles woken *|
| #steps | mean # part/step | max # part/step |
|      4 |          42.25 |          55      |
```

bin	dt	npart	frac	cpufrac
0	8.886E-02	0	0.00%	14.87%
1	4.443E-02	638476	41.76%	13.64%
2	2.221E-02	643124	42.06%	27.32%
3	1.111E-02	247351	16.18%	44.17%

0.11M particles per second, IND TIMESTEPS efficiency: 56.51%

MPI in Phantom

64 cores – 53 cpu/wall time

```
----->  TIME =      8.275      : full dump written to file jet1m_00097  <-----
```

input file jet1m.in written successfully.
Since code start: 746 timesteps, wall: 95s cpu: 5.09E+04s cpu/wall: 53
Since last dump : 8 timesteps, wall: 12s cpu: 692s cpu/wall: 60

	wall	cpu	cpu/wall	load bal	frac
└step	: 11.62s	691.92s	59.52	100.00%	92.08%
└└tree	: 2.88s	135.36s	47.08	100.00%	22.77%
└└└density	: 2.50s	163.60s	65.44	100.00%	19.80%
└└└└local	: 2.50s	161.29s	64.51	100.00%	19.80%
└└└force	: 5.75s	370.66s	64.46	100.00%	45.54%
└└└└local	: 4.62s	292.27s	63.19	100.00%	36.63%
└└cons2prim	: 0.12s	5.43s	43.41	100.00%	0.99%
└write_dump	: 1.00s	0.98s	0.98	100.00%	7.92%

```
-----
```

** Number of steps since last summary:	8	**
** Wall time since last summary:	11.625 seconds	**

```
-----
```

* particles woken		*
#steps mean # part/step max # part/step		
4 13.00 17		

```
-----
```

bin	dt	npart	frac	cpufrac
0	8.886E-02	0	0.00%	20.38%
1	4.443E-02	572589	37.45%	19.05%
2	2.221E-02	666016	43.56%	26.99%
3	1.111E-02	290346	18.99%	33.58%

```
-----
```

0.51M particles per second, IND TIMESTEPS efficiency: 75.48%

128 cores – 99 cpu/wall time

```
----->  TIME =      8.264      : full dump written to file jet1m-2n_00093  <-----
```

input file jet1m-2n.in written successfully.
Since code start: 744 timesteps, wall: 3040s cpu: 3.61E+05s cpu/wall: 99
Since last dump : 8 timesteps, wall: 50s cpu: 4983s cpu/wall: 100

	wall	cpu	cpu/wall	load bal	frac
└step	: 50.00s	4982.64s	99.65	99.84%	96.15%
└└tree	: 9.88s	224.62s	22.75	88.00%	18.99%
└└└balance	: 7.25s	14.16s	1.95	100.00%	13.94%
└└└density	: 27.25s	3259.12s	119.60	99.94%	52.40%
└└└└local	: 7.12s	814.16s	114.27	95.52%	13.70%
└└└└remote	: 20.12s	2434.36s	120.96	98.39%	38.70%
└└└force	: 12.38s	1464.86s	118.37	99.29%	23.80%
└└└└local	: 6.88s	815.92s	118.68	89.08%	13.22%
└└└└remote	: 4.62s	573.53s	124.01	86.59%	8.89%
└└cons2prim	: 0.25s	6.25s	25.00	81.82%	0.48%
└write_dump	: 2.00s	2.31s	1.16	43.69%	3.85%

```
-----
```

** Number of steps since last summary:	8	**
** Wall time since last summary:	50.000 seconds	**

```
-----
```

* particles woken		*
#steps mean # part/step max # part/step		
4 42.25 55		

```
-----
```

bin	dt	npart	frac	cpufrac
0	8.886E-02	0	0.00%	14.87%
1	4.443E-02	638476	41.76%	13.64%
2	2.221E-02	643124	42.06%	27.32%
3	1.111E-02	247351	16.18%	44.17%

```
-----
```

0.11M particles per second, IND TIMESTEPS efficiency: 56.51%

MPI in Phantom

64 cores – 0.51M particles per second

```
----->  TIME =      8.275      : full dump written to file jet1m_00097  <-----
```

input file jet1m.in written successfully.
Since code start: 746 timesteps, wall: 956s cpu: 5.09E+04s cpu/wall: 53
Since last dump : 8 timesteps, wall: 12s cpu: 692s cpu/wall: 60

		wall	cpu	cpu/wall	load bal	frac
└step	:	11.62s	691.92s	59.52	100.00%	92.08%
└tree	:	2.88s	135.36s	47.08	100.00%	22.77%
└density	:	2.50s	163.60s	65.44	100.00%	19.80%
└└local	:	2.50s	161.29s	64.51	100.00%	19.80%
└force	:	5.75s	370.66s	64.46	100.00%	45.54%
└└local	:	4.62s	292.27s	63.19	100.00%	36.63%
└cons2prim	:	0.12s	5.43s	43.41	100.00%	0.99%
└write_dump	:	1.00s	0.98s	0.98	100.00%	7.92%

```
** Number of steps since last summary:      8      **|
** Wall time since last summary:      11.625 seconds  **|
```

```
* particles woken *|
| #steps | mean # part/step | max # part/step |
|      4 |      13.00 |      17 |
```

bin	dt	npart	frac	cpufrac
0	8.886E-02	0	0.00%	20.38%
1	4.443E-02	572589	37.45%	19.05%
2	2.221E-02	666016	43.56%	26.99%
3	1.111E-02	290346	18.99%	33.58%

0.51M particles per second, IN TIMESTEPS efficiency: 75.48%

128 cores – 0.11M particles per second

```
----->  TIME =      8.264      : full dump written to file jet1m-2n_00093  <-----
```

input file jet1m-2n.in written successfully.
Since code start: 744 timesteps, wall: 3640s cpu: 3.61E+05s cpu/wall: 99
Since last dump : 8 timesteps, wall: 50s cpu: 4983s cpu/wall: 100

		wall	cpu	cpu/wall	load bal	frac
└step	:	50.00s	4982.64s	99.65	99.84%	96.15%
└tree	:	9.88s	224.62s	22.75	88.00%	18.99%
└└balance	:	7.25s	14.16s	1.95	100.00%	13.94%
└density	:	27.25s	3259.12s	119.60	99.94%	52.40%
└└local	:	7.12s	814.16s	114.27	95.52%	13.70%
└└remote	:	20.12s	2434.36s	120.96	98.39%	38.70%
└force	:	12.38s	1464.86s	118.37	99.29%	23.80%
└└local	:	6.88s	815.92s	118.68	89.08%	13.22%
└└remote	:	4.62s	573.53s	124.01	86.59%	8.89%
└cons2prim	:	0.25s	6.25s	25.00	81.82%	0.48%
└write_dump	:	2.00s	2.31s	1.16	43.69%	3.85%

```
** Number of steps since last summary:      8      **|
** Wall time since last summary:      50.000 seconds  **|
```

```
* particles woken *|
| #steps | mean # part/step | max # part/step |
|      4 |      42.25 |      55 |
```

bin	dt	npart	frac	cpufrac
0	8.886E-02	0	0.00%	14.87%
1	4.443E-02	638476	41.76%	13.64%
2	2.221E-02	643124	42.06%	27.32%
3	1.111E-02	247351	16.18%	44.17%

0.11M particles per second, AND TIMESTEPS efficiency: 56.51%

MPI in Phantom

64 cores – 0.51M particles per second

```
----->  TIME =      8.275      : full dump written to file jet1m_00097  <-----
```

input file jet1m.in written successfully.
Since code start: 746 timesteps, wall: 956s cpu: 5.09E+04s cpu/wall: 53
Since last dump : 8 timesteps, wall: 12s cpu: 692s cpu/wall: 60

	step	tree	density	local	force	local	cons2prim	write_dump
wall	11.62s	2.88s	2.50s	2.50s	5.75s	4.62s	0.12s	1.00s
cpu	691.92s	135.36s	163.60s	161.29s	370.66s	292.27s	5.43s	0.98s
cpu/wall	59.52	47.08	65.44	64.51	64.46	63.19	43.41	0.98
load bal	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
frac	92.08%	22.77%	19.80%	19.80%	7.92%	7.92%	7.92%	7.92%

```
** Number of steps since last summary:
** Wall time since last summary:      11.625 seconds **
```

* particles woken *
#steps mean # part/step max # part/step
4 13.00 17

bin	dt	npart	frac	cpufrac
0	8.886E-02	0	0.00%	20.38%
1	4.443E-02	572589	37.45%	19.05%
2	2.221E-02	666016	43.56%	26.99%
3	1.111E-02	290346	18.99%	33.58%

0.51M particles per second, IN TIME STEPS efficiency: 75.48%

128 cores – 0.11M particles per second

```
----->  TIME =      8.264      : full dump written to file jet1m-2n_00093  <-----
```

input file jet1m-2n.in written successfully.
Since code start: 744 timesteps, wall: 3640s cpu: 3.61E+05s cpu/wall: 99
Since last dump : 8 timesteps, wall: 50s cpu: 4983s cpu/wall: 100

	step	tree	balance	density	local	force	local	cons2prim	write_dump
wall	50.00s	9.88s	7.25s	27.25s	7.12s	12.38s	6.88s	4.62s	2.00s
cpu	4982.64s	224.62s	14.16s	3259.12s	814.16s	2434.36s	1464.86s	815.92s	573.53s
cpu/wall	99.65	22.75	1.95	119.60	114.27	120.96	118.37	118.68	124.01
load bal	99.84%	88.00%	100.00%	99.94%	95.52%	98.39%	99.29%	89.08%	86.59%
frac	96.15%	18.99%	13.94%	52.40%	13.70%	38.70%	23.80%	13.22%	8.89%

```
** Number of steps since last summary:      8
** Wall time since last summary:      50.000 seconds **
```

* particles woken *
#steps mean # part/step max # part/step
4 42.25 55

bin	dt	npart	frac	cpufrac
0	8.886E-02	0	0.00%	14.87%
1	4.443E-02	638476	41.76%	13.64%
2	2.221E-02	643124	42.06%	27.32%
3	1.111E-02	247351	16.18%	44.17%

0.11M particles per second, AND TIME STEPS efficiency: 56.51%

**MPI is 5x slower!!
(for this problem)**

Multi-Physics SPH

- One challenge is including **multiple types of physics**.
- Shamrock has demonstrated that **hydrodynamics** can be scaled to **500 billion particles**.
- But how do we do that for gravity, dust (1-fluid, 2-fluid, multigrain, growth, etc), magnetic fields, relativity, radiation (FLD), sink particles, winds, chemistry evolution, etc etc?

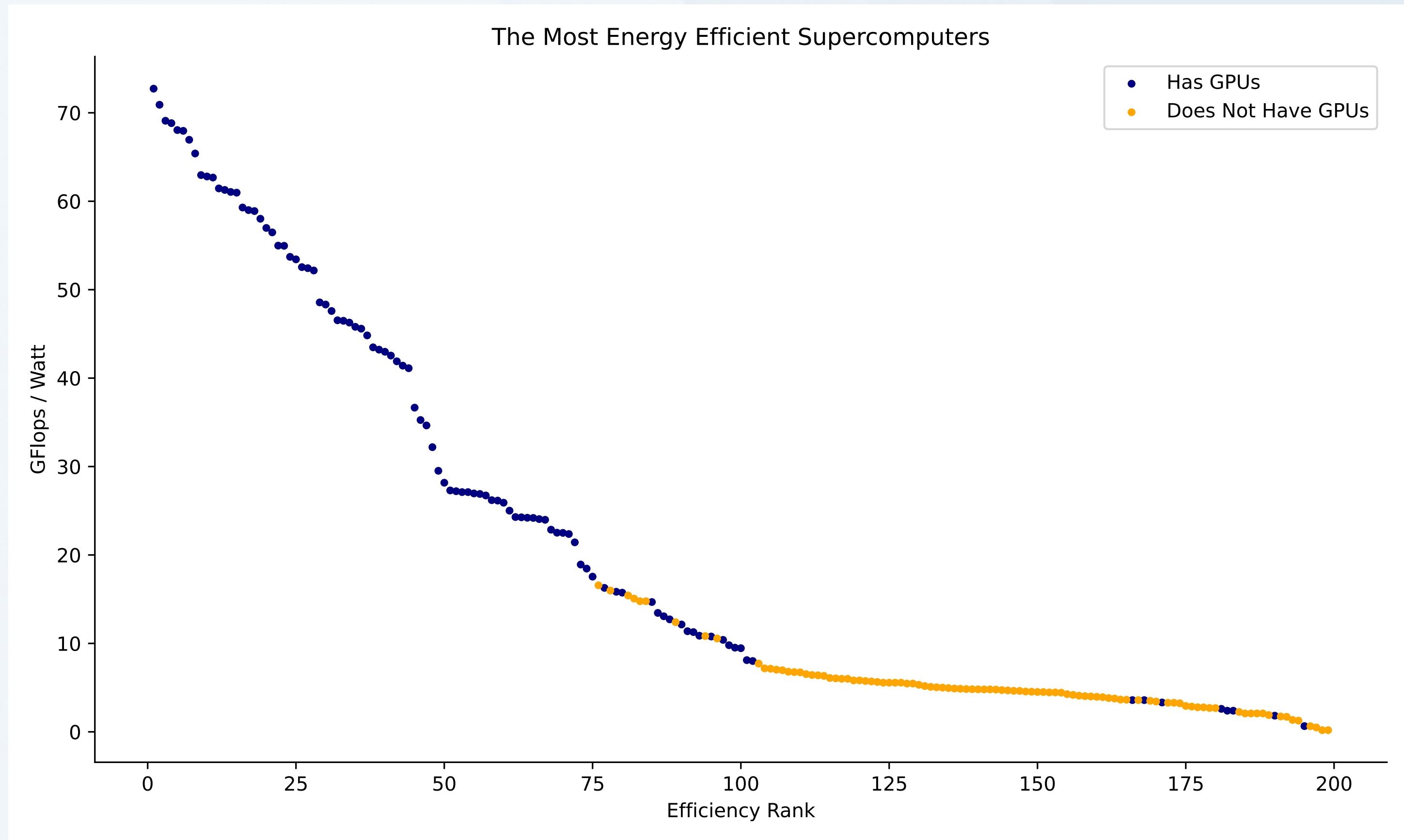
Scaling Multi-Physics SPH

- I am very interested in **parallelizing multi-physics SPH** simulations.

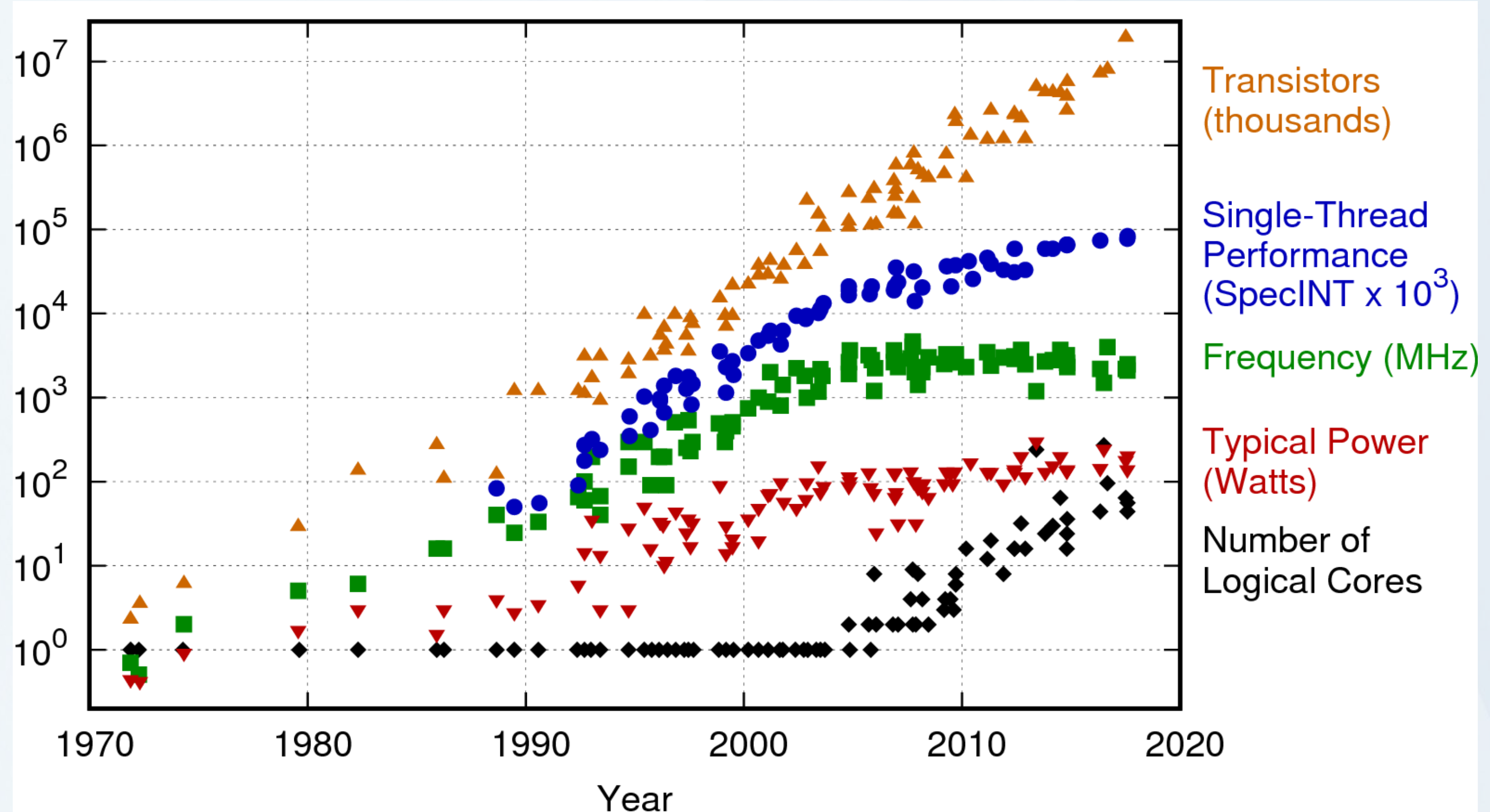
Scaling Multi-Physics SPH

- I am very interested in **parallelizing multi-physics SPH** simulations.
- Algorithmic optimizations are important!
- But we need to **combine** existing algorithmic optimizations with the ability to add more compute.
- One very important area will be GPUs.
(see talks, e.g., by Timothée, Andrew and Tom)

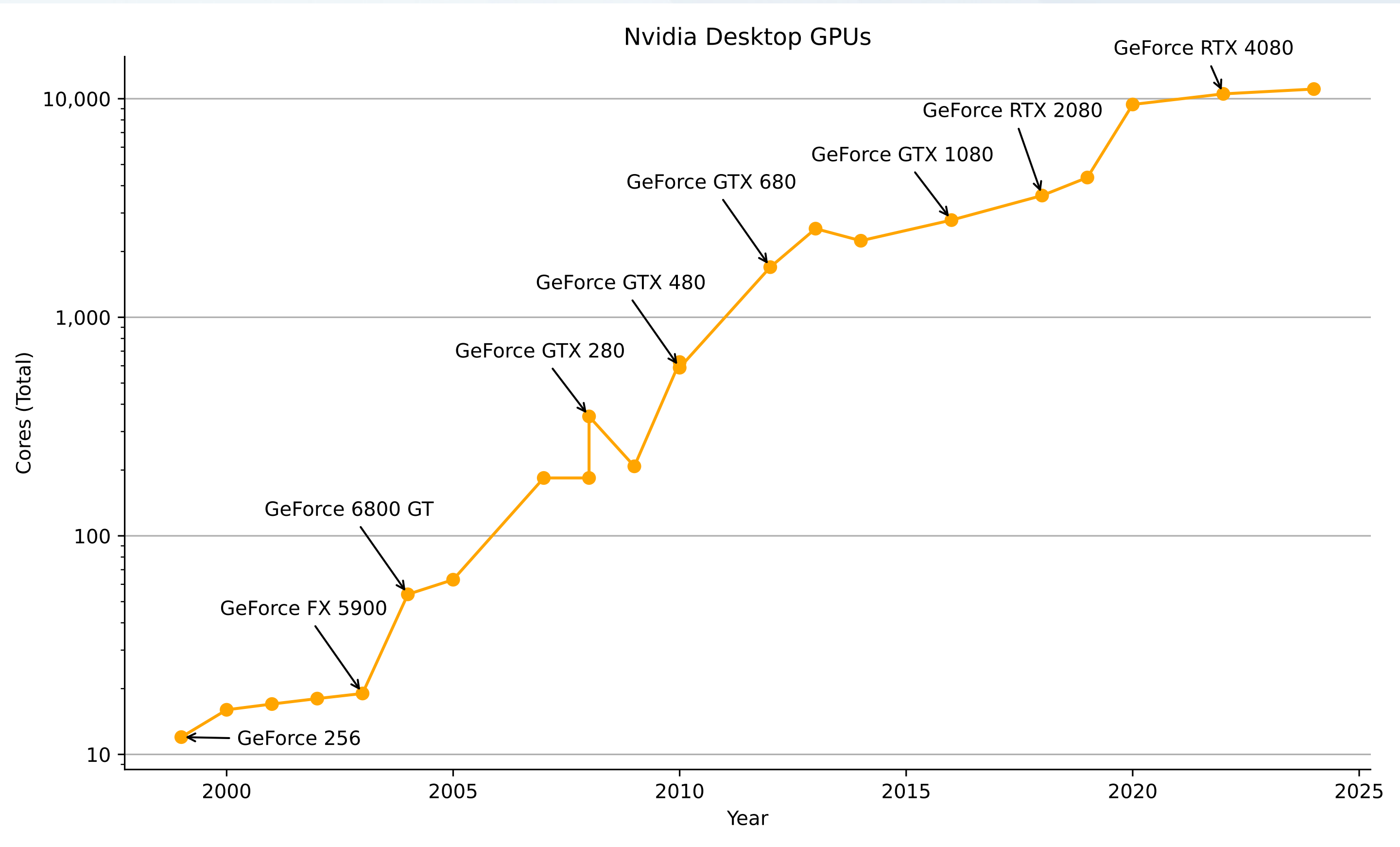
Supercomputer Energy Performance



42 Years of CPU Trends



30 Years of Nvidia GPUs



Do I have answers?

Do I have answers?

No. But we are looking for solutions.

Phantom Parallelization Roadmap

- Short term goals for improving the parallel performance of Phantom:
 1. Use 1-4 GPUs to increase compute.
 - Most HPC nodes have an attached GPU (or 4).
 - Needs to work with individual particle timesteps (Andrew Harris).
 - Key is for your simulation to not go slower when GPU=yes.
 2. MPI scaling to 4-8 nodes.
 - Will be engaging with the HPC consortium in Canada to improve Phantom's current MPI implementation.

Summary

- We are working on **MPI** and **GPU** angles to increase **Phantom** parallelization.
- Important to retain algorithmic optimizations + all the physics.
- I did not speak about this, but am very interested in **MHD everything**.
 - MRI in global discs (Jackson Narvaez).
 - Magnetized white dwarf mergers.