Non-linear structure formation with N-body simulations

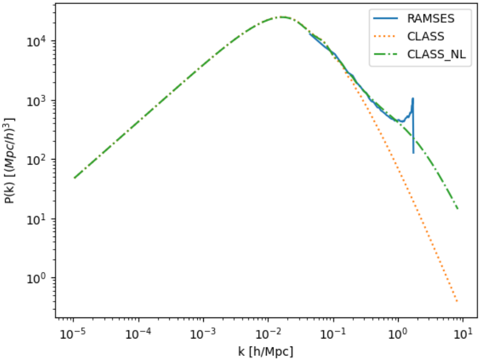
Clément Stahl

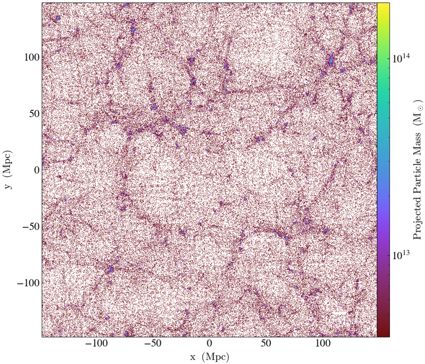
Observatoire Astronomique de Strasbourg, Université de Strasbourg

7 June 2023

Outline


linear
CLASS
evolution
Monofonic
realization
non-linear
RAMSES
evolution
visualization
YT
summary statistics


RAMSES
CLASS
CLASS_NL
P(k) [(Mpc/h)^3]
k [h/Mpc]


y (Mpc)
x (Mpc)
Projected Particle Mass (M$_\odot$)

Why a (super-)computer?

- Linear perturbation theory: a (very good) tool when

$$\frac{\rho - \bar{\rho}}{\bar{\rho}} \equiv \delta \ll 1 \tag{1}$$

- Quasi linear scales: $\delta \sim 1$: various analytical approach [NOT the topic today]
  - Lagrangian perturbation theory (Buchert 92, Bernardeau & Valageas 08)
  - renormalized perturbation theory (Crocce & Scoccimarro 05)
  - regularized perturbation theory (RegPT, Bernardeau 15)
  - path integral formalism (Valageas 03)
  - coarse grained perturbation theory (Matarrese 07)
  - renormalization group flow (Pietroni 08)
  - kinetic field theory (Bartelmann 19)
  - effective field theories (Baumann 10)
    - `CLASS-PT`
    - `PyBird` (Pierre Zhang)

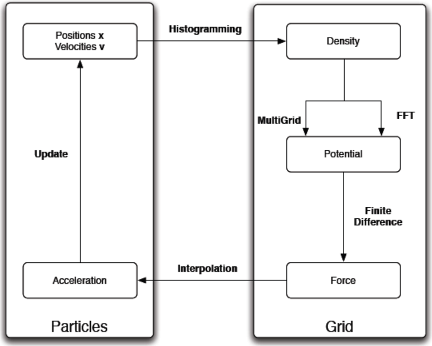
  $\rightarrow k \sim 0.3h/\mathrm{Mpc}$
- Strong non-linear regime: $\delta \gg 1$, no analytical approximations for gravitational evolution of matter density perturbations $\rightarrow$ **numerical approach** [topic of today :)]

Numerical approach

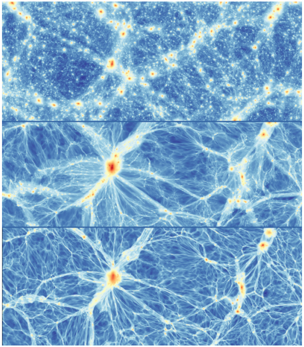
- Large number of elements (stars, dark matter particles) $\rightarrow$ statistical approach: **distribution function in phase space**: f_{m}
- Dark matter assumed to be composed of $\mathcal{O}(10^6 - 10^{12})$ particles gravitationally interacting.
- Vlasov-Poisson system : $(\Phi \sim 10^{-5}, v \sim 10^{-2})$

$$\frac{\partial f_{\mathrm{m}}}{\partial t} + \frac{v_i}{a^2}\frac{\partial f_{\mathrm{m}}}{\partial x^i} - \frac{\partial \Phi}{\partial x^i}\frac{\partial f_{\mathrm{m}}}{\partial v_i} = 0 \qquad (2)$$

$$\Delta \Phi = 4\pi G \bar{\rho} a^2 \delta \qquad (3)$$



Credits : Aubert 19



Credits : Angulo 21

Review: Angulo & Hahn 21

Different gravity solvers

- Particle-Particle (Aarseth03)
- Hierarchical Tree (Barnes & Hut 86).
  Exemples of (public) tree codes: PKDGRAV3 (Potter 16) GADGET4 (Springel 20), ChaNGa (Quinn 15).
- Particle-Mesh (Hockney & Eastwood 81).
  Exemples of (public) mesh codes: FASTPM (Feng 16), ENZO (Bryan 97), **RAMSES** (Teyssier 02), AMIGA (Knebe01)

Time evolution

- Once you have calculated the gravitational force for every particle, needs to move the particle according to its dynamics (Newton's law)
- Second order ODE: Runge-Kutta4, or LeapFrog
- Symplectic structure: Kick-Drift-Kick or Drift-Kick-Drift.

Supercalculators

- HPC infrastructure = paralellism. MPI library to communicate among different nodes. OpenMP to speed up directives in nodes computations.
- Computational domain (particles or particles+grid) decomposed into smaller pieces distributed over the nodes.
- Hilbert curve decomposition (<u>note</u>: for very cluttered situation `MULTIPLEDOMAINS` helps in GADGET4.)

Jean Zay @ IDRIS

Joliot Curie @ TGCC

Jean Zay (IDRIS, Saclay)

- <u>CSL:</u> 86 344 CPU cores
- <u>V100:</u> 2448 V100 GPUs
- <u>A100:</u> 416 A100 GPUs

Joliot Curie (TGCC, Bruyères-le-Châtel)

- <u>Irene SKL:</u> 79 488 CPU cores
- <u>Irene Rome:</u> 293 376 CPU cores
- <u>Irene V100:</u> 128 V100 GPUs

State of the art simulations

Year	Simulation	Code, Algorithm	Supercomputer, Location	Cores $[10^3]$	N_p $[10^{12}]$	Box $[h^{-1}\text{Gpc}]$	ϵ $[h^{-1}\text{kpc}]$
2014	Dark Sky (Skillman et al. 2014)	2HOT FMM	Titan USA	20	1.1	8	36.8
2017	TianNu (Emberson et al. 2017)	CUBEP^{3}M PM-PM-PP	Tianhe-2 China	331	2.97	1.2	13
2017	Euclid Flagship (Potter et al. 2017)	PKDGRAV3 Tree-FMM	PizDaint Switzerland	4	2.0	3.	4.8
2019	Outer Rim (Heitmann et al. 2019)	HACC Tree-PM	Mira USA	524	1.07	3.0	2.84
2019	Cosmo-π (Cheng et al. 2020)	CUBE PM-PM	π 2.0 China	20	4.39	3.2	195
2020	Uchuu (Ishiyama et al. 2021)	GreeM Tree-PM	ATERUI-II Japan	$<$40	2.0	2.0	4.3
2020	Last Journey (Heitmann et al. 2021)	HACC Tree-PM	Mira USA	524	1.24	3.4	3.14
2021	Far Point (Frontiere et al. 2021)	HACC Tree-PM	Summit USA	?	1.86	1	0.8

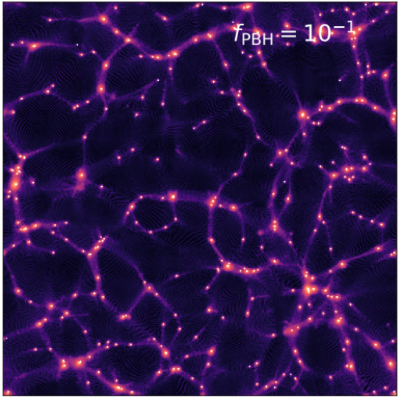
Table 1 List of cosmological simulations with a particle number in excess of 1 trillion (10^{12})

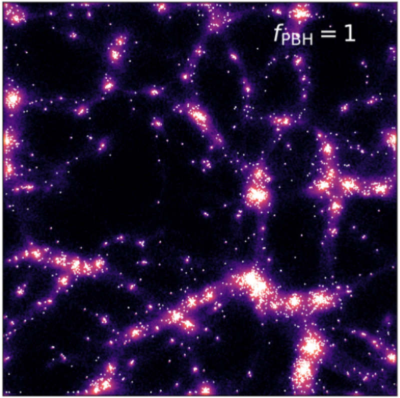
Gordon Bell Prize

'Modified' gravity

Some selected codes (often written on the top of RAMSES or GADGET)

- Axion/Fuzzy Dark matter: `SCALAR` (Mina 19), May 21, `axionCAMB`
- $f(R)$ (Zhao 10), `ECOSMOG` (Li 11, Brax 12), Modified Gravity-GADGET (Puchwein 13), (Arnold 19), `MG-GLAM` (Ruan 21 & Hernández-Aguayo 21)
- `ISIS` (Llinares 13), Scalar dark matter: (Hopkins 18)
- MOND: Phantom of RAMSES, Dipolar dark matter, `Bi-Poisson`
- ΛPBH (Inmar 19)





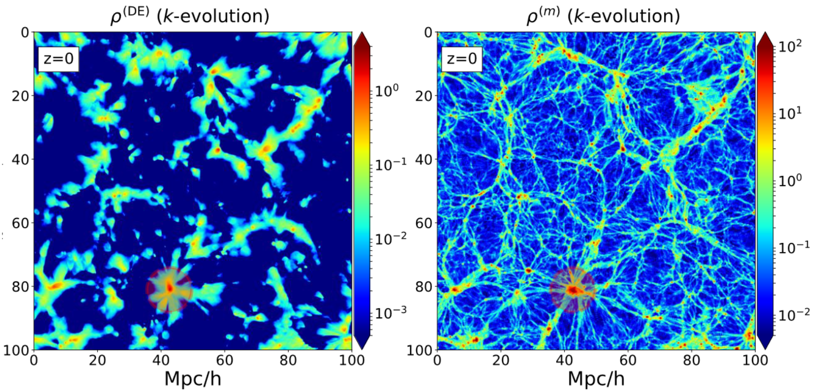
General Relativity

Instead of Poisson, solve

$$-3\mathcal{H}(\phi' + \mathcal{H}\psi) + \Delta\phi = 4\pi G\bar{\rho}a^2\delta \quad (4)$$

+ 5 other degrees of freedom of GR

- RayGal (Breton 21)
- GRAMSES (Barrera-Hinojosa 19)
- `RELIC` (Adamek 20), `gevolution` (Adamek 16)
- `k-evolution` (Hassani 19), Asevolution (Christiansen 23)



1. Motivation and general considerations
2. Setting up initial condition: `Monofonic`
3. Non-linear dynamics: `RAMSES`
4. Diagnostic out the output: `YT, Pylians`

General description

Goal

Convert the (linear) power spectrum into a list of particles' positions + velocities

Draw a realisation of a 3D Gaussian Random field

$$\delta(\vec{x}) \propto \int \frac{d^3\vec{k}}{(2\pi)^3} \sqrt{P_L(k)} \mathcal{N}_k(0,1) e^{i\vec{k}\cdot\vec{x}}, \quad (5)$$

where $\mathcal{N}(0,1)$ is drawn from a Gaussian random field and P_L is the linear power spectrum (CLASS, CAMB)
$\rightarrow$ velocities obtained with linear perturbation theory

Example of initial condition generators

- mpgrafic
- 2LPTIC
- FASTPM
- N-GENICS (with GADGET)
- **MUSIC** and **Monofonic**
- ginnungagap
- genetIC
- Panphasia

```
#####################################################################################
# Example config file for MUSIC2 - monofonIC single resolution simulation ICs
#                                                              version 1 from 2020/08/23
#####################################################################################

#####################################################################################
[setup]

GridRes         = 128      # number of grid cells per linear dimension for calculations
                           #    = particles for sc initial load
BoxLength       = 300      # length of the box in Mpc/h
zstart          = 24.0     # starting redshift
```

Comments: on a laptop, GridRes ~ 128
good to start at early enough time for linear theory to be valid

```
[cosmology]
## transfer = ... specifies the Einstein-Boltzmann plugin module

ParameterSet    = Planck2018EE+BAO+SN  # specify a pre-defined parameter set,
                                       # or set to 'none' and set manually below

## cosmological parameters, to set, choose ParameterSet = none,
## default values (those not specified) are set to the values
## from 'Planck2018EE+BAO+SN', we currently assume flatness
# Omega_m         = 0.3158
# Omega_b         = 0.0494
# Omega_L         = 0.6842
# H0              = 67.321
# n_s             = 0.9661
# sigma_8         = 0.8102
# A_s             = 2.148752e-09  # can use A_s instead of sigma_8 when using CLASS
# Tcmb            = 2.7255
# k_p             = 0.05
# N_ur            = 2.046
# m_nu1           = 0.06
# m_nu2           = 0.0
```

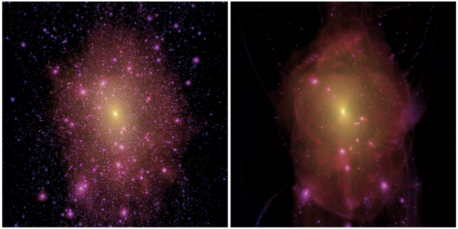
Physics beyond ΛCDM in Monofonic

Dark Matter

- Warm Dark Matter: Vanilla (public) monofonic
- ETHOS-like parametrization (Cyr-Racine 15, Murgia 17):

$$P_X/P_{\rm CDM} = \left(1 + (\alpha k)^{\beta}\right)^{-2\gamma} \quad (6)$$

(WDM: $\beta = \gamma = 2.23$)
describe decay of the power spectrum of fuzzy dark matter, sterile ν...
Implemented in a separate branch of monofonic (available upon request)



Credits : Angulo 21

Dark Energy

Scalar dark energy and w_0, w_a in branch of monofonic (available upon request)

Physics beyond ΛCDM in Monofonic

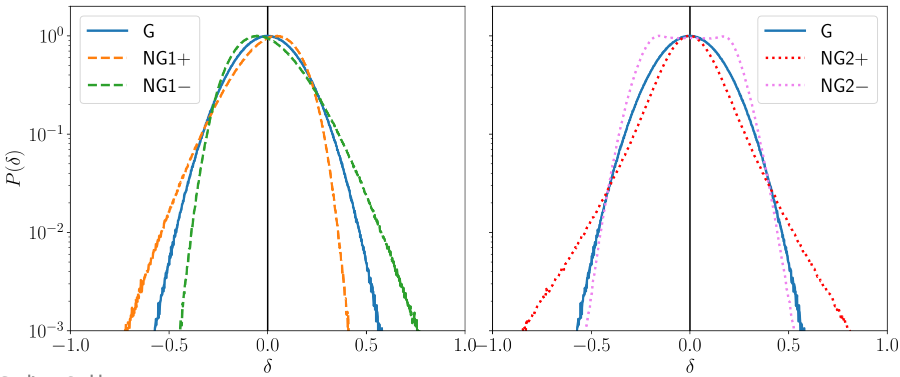
Primordial non-Gaussianities (PNG)

$$\Phi = \Phi_G + f_{\rm NL}\Phi_G^2 + g_{\rm NL}\Phi_G^3 \quad (7)$$

Scale dependant PNG:

$$f_{\rm NL}(k) = f_{\rm NL}^0 \left(\frac{k}{k_0}\right)^{n_{f_{\rm NL}}} \quad (8)$$

(available upon request)

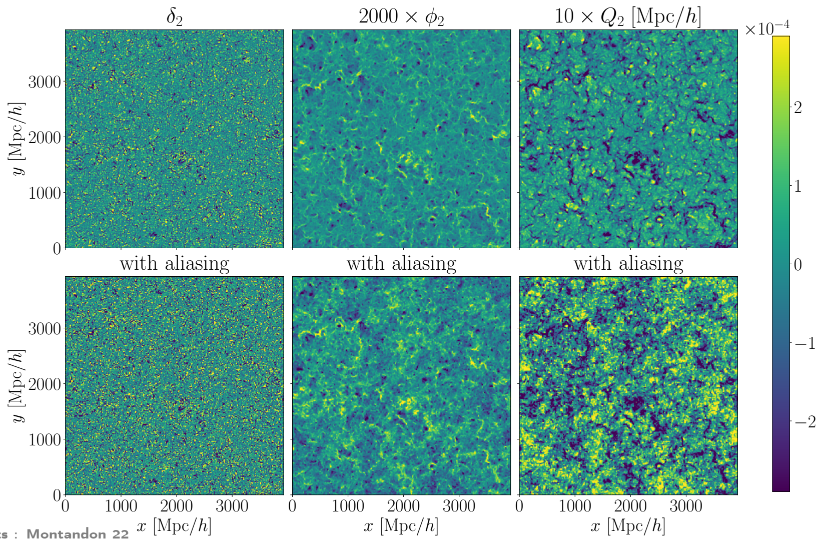


Credits : Stahl 22

Nice features of Monofonic: De-aliasing (*)

Aliasing:

numerical error due to sampling of the fields
→ most important when multiplying two fields (PNG, nLPT...)

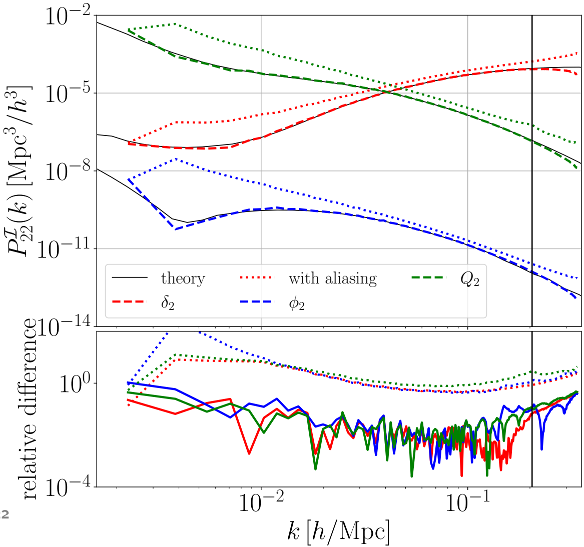


Credits : Montandon 22

Nice features of Monofonic: De-aliasing (*)

Orszag's 3/2 rule

implemented in monofonic (Orszag 71)



Credits : Montandon 22

Nice features of Monofonic (*)

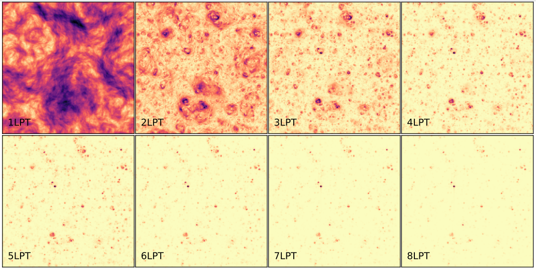
Transiants

Linear perturbation theory has two solutions: the growing modes ($\propto a(t)$ in matter domination) and the decaying mode ($\propto a^{-3/2}$).
Numerical error to neglect decaying mode at early time → transiant Crocce 06.

Push perturbation theory to higher order, to start simulation later, so that decaying modes had time to decay.

- 2LPTIC code
- 3LPT (start simulation at $z = 7$) in vanilla Monofonic and recursion relations nLPT implemented in a branch of Monofonic (Rampf 20)

```
LPTorder        = 3         # order of the LPT to be used (1,2 or 3)
```



Nice features of Monofonic (*)

```
DoFixing        = no      # do mode fixing à la Angulo&Pontzen (https://arxiv.org/abs/1603.05253)
DoInversion     = no       # invert phases (for paired simulations)
```

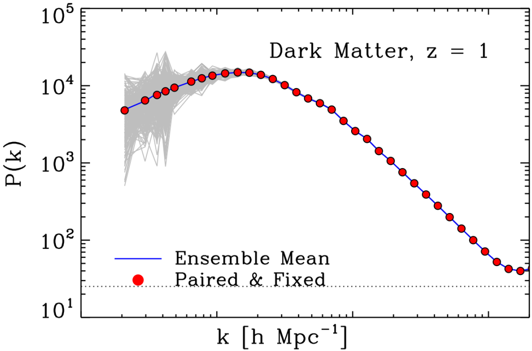
Fixing: draw the amplitude from a Dirac-delta

the phase θ_k is drawn from $[0, 2\pi]$

$$\delta(\vec{x}) \propto \int \frac{d^3\vec{k}}{(2\pi)^3} \sqrt{P_L(k)} e^{i\theta_k} e^{i\vec{k}\cdot\vec{x}}, \quad (9)$$

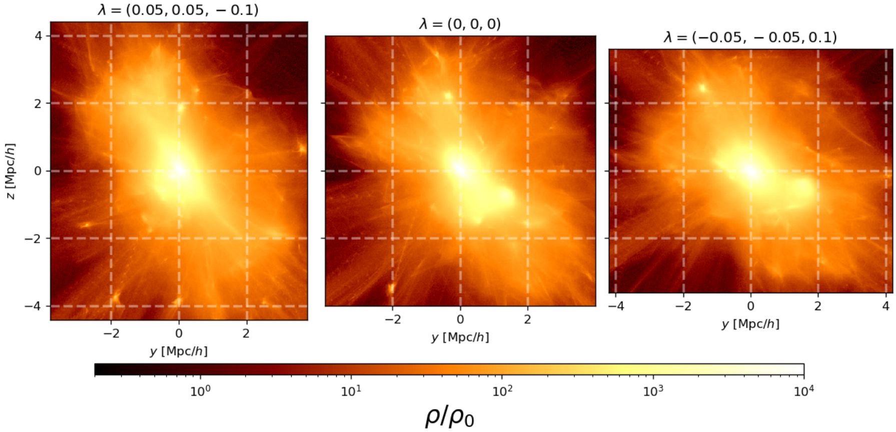
no more Gaussian random field realisation!

Paired = take the same simulation with $\theta_k + \pi$



Anisotropic simulations (*)

```
## Use below for anisotropic large scale tidal field ICs up to 2LPT
## see Stuecker+2020 (https://arxiv.org/abs/2003.06427)
# LSS_aniso_lx    = +0.1
# LSS_aniso_ly    = +0.1
# LSS_aniso_lz    = -0.2
```



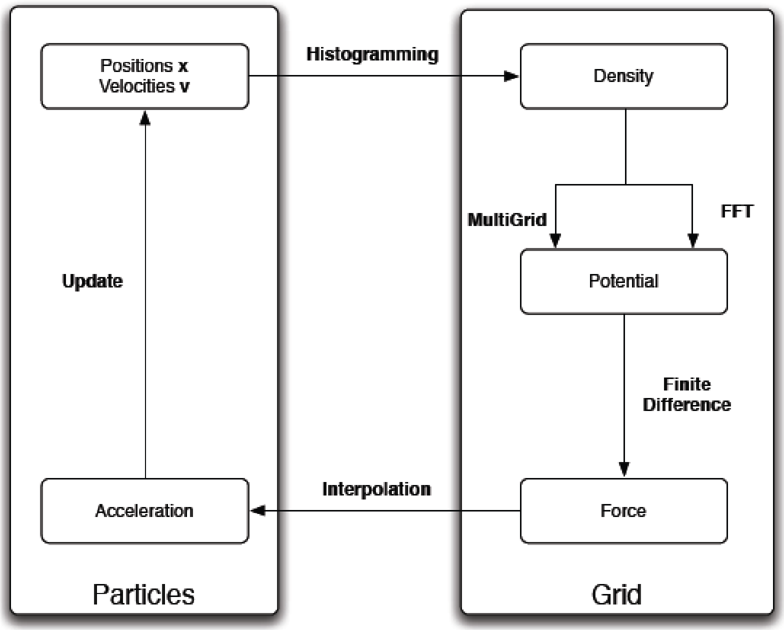
Possible outputs

```
[output]
## format = .... specifies the output plugin module

##> RAMSES / GRAFIC2 compatible format
# format                    = grafic2
# filename          = ics_ramses
# grafic_use_SPT    = no # if no then uses PPT, otherwise linear SPT

##> Gadget-2/3 'fortran unformatted binary'-style format
# format            = gadget2
# filename          = ics_gadget.dat
# UseLongids        = false

##> Gadget-2/3 HDF5 format
# format            = gadget_hdf5
# filename          = ics_gadget.hdf5

##> Arepo HDF5 format (virtually identical to gadget_hdf5)
# format            = AREPO
# filename          = ics_arepo.hdf5

##> HACC compatible generic-io format
# format            = genericio
# filename          = ics_hacc

##> SWIFT compatible HDF5 format. Format broadly similar to gadget_hdf5 but in a single
##> file even when using MPI. No h-factors for position and masses and no sqrt(a)-factor for the velocities.
##> IDs are stored using 64-bits unless UseLongids is set to false.
# format            = SWIFT
# filename          = ics_swift.hdf5
# UseLongids        = true

##> Generic HDF5 output format for testing or PT-based calculations
# format            = generic
# filename          = debug.hdf5
# generic_out_eulerian = yes  # if yes then uses PPT for output
```

don't forget to uncomment the desired output (RAMSES, for this tutorial)

1. Motivation and general considerations
2. Setting up initial condition: Monofonic
3. Non-linear dynamics: RAMSES
4. Diagnostic out the output: YT, Pylians

RAMSES in one slide



Credits : Aubert 19

Example of input file for RAMSES

```
&RUN_PARAMS
cosmo=.true.
pic=.true.
poisson=.true.
hydro=.false.
nrestart=0
nremap=1
nsubcycle=1,2
ncontrol=1
verbose=.false.
/

&AMR_PARAMS
levelmin=7
levelmax=12
nexpand=1
ngridmax=2000000
npartmax=3000000
/
```

```
&INIT_PARAMS
filetype='grafic'
initfile(1)='../music/ics_ramses/level_007/'
/

&OUTPUT_PARAMS
foutput=1000
noutput=1
/

&REFINE_PARAMS
m_refine=6*8.,
/
```

`levelmin` and `levelmax` control the minimal and maximal level of refinement of the RAMSES grid.

Hydrodynamics in RAMSES (*)

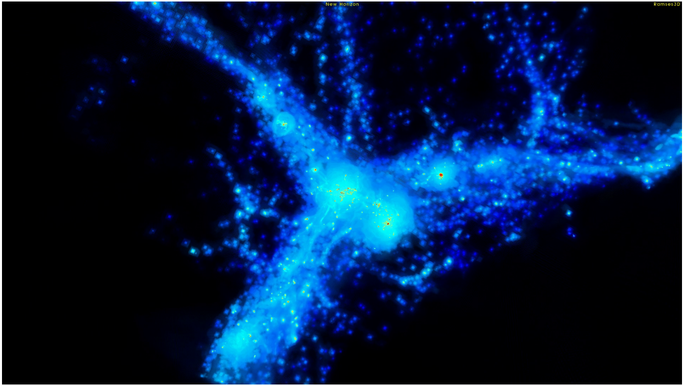
```
&RUN_PARAMS
cosmo=.true.
pic=.true.
poisson=.true.
ordering='hilbert'
nremap=20
nsubcycle=1,30*2
nrestart=55
hydro=.true.
ncontrol=1
sink=.true.
sinkprops=.true.
/

&HYDRO_PARAMS
gamma=1.6666667
courant_factor=0.8
slope_type=1
pressure_fix=.true.
scheme='muscl'
riemann='hllc'
/
```

NewHorizon
El Gordo cluster

```
&PHYSICS_PARAMS
haardt_madau=.true.
cooling=.true.
z_reion=10.0
z_ave=1d-3
n_star=0.1d0
eps_star=0.02
del_star=50d0
T2_star=1d4
g_star=1.3333334
f_ek=1d0
Mseed=1d5
sink_AGN=.true.
bondi=.true.
drag=.true.
X_floor=1d-2
eAGN_K=1d0
eAGN_T=0.15d0
TAGN=0.0
r_gal=50.0d0
sigmav_max=1d15
T2maxAGN=1d10
boost_acc=2d0
boost_drag=2d0
metal=.true.
eta_sn=0.3d0
yield=0.05d0
f_w=10d0
t_delay=10d0
/
```

Things to bear in mind when running a simulation

- Distances larger than ~ 0.33 box size L affected by periodic boundary condition
- Cosmic variance effects due to box size L (pairing...)
- Check gravitational softening length ϵ (typically $\frac{1}{30}L/N^{1/3}$)
- Distances smaller than 2-3 cells size or $3 \times \epsilon$ affected by numerical error (unresolved force)
- Too large time step can introduce error in particle trajectories



Credits : NewHorizon

For this tutorial:

- Visualization with YT[a]
- Power spectrum with Pylians[b]
- Halo Mass Function with YT theoretical line with colossus[c]

They are very well documented and with a large community helping and developing!

---

[a] can also give you access to data including underlying gravity fields (rotation curves, modified gravity...), 3D visualization (volume rendering), derived quantities...

[b] can also give you density field, real space correlation function, bispectrum, voids, redshift space distorsion...

[c] can also give you halo mass function and halo bias, density profiles, splashback radius, concentration

Other tools include POWMES, Pynbody, Nbodykit, bskit, Hipster, Corrfunc, HALOTOOLS, ytree, Glnemo, uns_io...


THE #1 PROGRAMMER EXCUSE FOR LEGITIMATELY SLACKING OFF:
"MY CODE'S COMPILING."
HEY! GET BACK TO WORK!
COMPILING!
OH. CARRY ON.