Building a matter power spectrum emulator from N-body simulations in $f(R)$CDM cosmology

Iñigo Sáez Casares
Work in progress with Yann Rasera

October 15, 2021

Accelerated expansion of the Universe

Accelerated expansion of the Universe discovered in 1998 (SNe).
Acceleration condition, fluid with an equation of state:

$$w = P/\rho c^2 < -1/3$$

$\longrightarrow$Concept of **Dark Energy**.

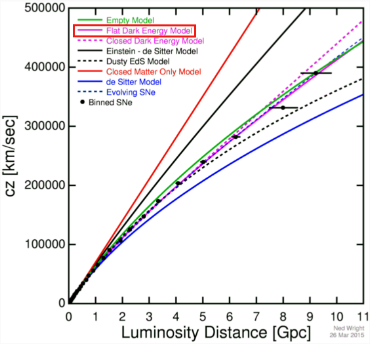


Figure 1: Hubble diagram, credits : N. Wright.

Accelerated expansion of the Universe

Accelerated expansion of the Universe discovered in 1998 (SNe).
Acceleration condition, fluid with an equation of state:

$$w = P/\rho c^2 < -1/3$$

$\longrightarrow$Concept of **Dark Energy**.

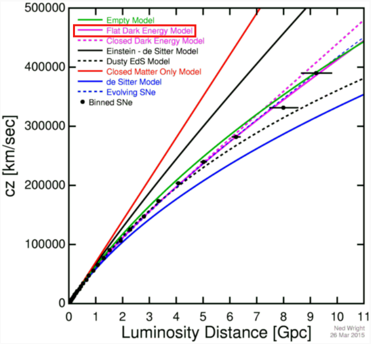


Figure 1: Hubble diagram, credits : N. Wright.

Question : What is the nature of DE ?

$\longrightarrow$Standard cosmological model: Cosmological Constant Λ
$\longrightarrow$DE fluid ?
$\longrightarrow$Modified gravity?

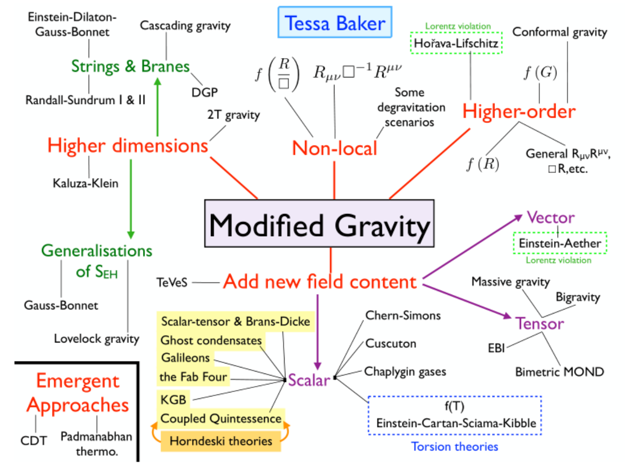
Models beyond ΛCDM

$\longrightarrow$**Dynamical dark energy**: dark energy density evolving in space and time (quintessence...).

Models beyond ΛCDM

⟶**Dynamical dark energy**: dark energy density evolving in space and time (quintessence...).

⟶**Modified gravity**: lift assumptions from General Relativity.



Measuring the structure of the Universe

$\longrightarrow$Two types of theories that can be tested by studying the Large Scale Structure of the Universe.

Measuring the structure of the Universe

$\longrightarrow$Two types of theories that can be tested by studying the Large Scale Structure of the Universe.

$\longrightarrow$One of the objectives of the ongoing (DESI) and future (Euclid, LSST $\sim$ 2022) large surveys.

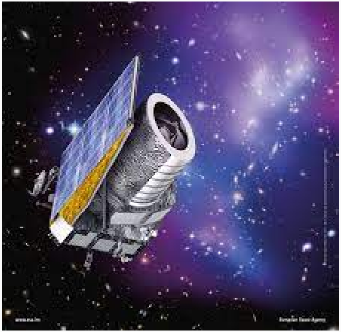


Figure 2: EUCLID satellite.

Figure 3: LSST telescope.

N-body simulations

⟶In **the non-linear regime** ($k \gtrsim 0.1/\mathrm{Mpc}$) we can discriminate between **dark energy** et **modified gravity**.
⟶**N-body simulations** needed to obtain precise predictions deep into the NL regime (but very slow...).

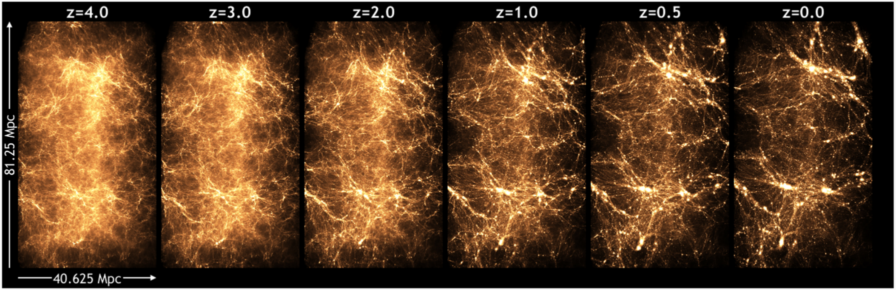


Figure 4: N-body cosmological simulation, snapshots at different redshifts. Credits : Heitmann et al. (2015).

Objectives

Study a simple MG model $f(R)$, belonging to the scalar-tensor family.

Objectives

Study a simple MG model $f(R)$, belonging to the scalar-tensor family.

Objectives:

- Impact of MG on the distribution of matter.
- Explore the space of cosmological parameters.
- Build an **emulator** for $P(k)$ calibrated on N-body simulations: obtain fast predictions for the non-linear matter power spectrum.

The $f(R)$ model

$$S_{\mathrm{EH}} = \frac{c^4}{16\pi G}\int \mathrm{d}^4x\sqrt{-g}\,[R - 2\Lambda] \longrightarrow \frac{c^4}{16\pi G}\int \mathrm{d}^4x\sqrt{-g}\,[R + f(R)]$$

The $f(R)$ model

$$S_{\mathrm{EH}} = \frac{c^4}{16\pi G}\int \mathrm{d}^4x\sqrt{-g}\,[R-2\Lambda] \longrightarrow \frac{c^4}{16\pi G}\int \mathrm{d}^4x\sqrt{-g}\,[R+f(R)]$$

$\longrightarrow$New dynamical degree of freedom, scalar field $f_R = \frac{\mathrm{d}f}{\mathrm{d}R}$.

The $f(R)$ model

$$S_{\mathrm{EH}} = \frac{c^4}{16\pi G} \int \mathrm{d}^4 x \sqrt{-g}\, [R - 2\Lambda] \longrightarrow \frac{c^4}{16\pi G} \int \mathrm{d}^4 x \sqrt{-g}\, [R + f(R)]$$

$\longrightarrow$New dynamical degree of freedom, scalar field $f_R = \frac{\mathrm{d}f}{\mathrm{d}R}$.

$\longrightarrow$Depending on the form of $f(R)$ this model can:

- produce cosmic acceleration,
- satisfy local constraints (”chameleon” screening mechanism) where gravity is well constrained (laboratory tests, solar system...).

$f(R)$ gravity - The Hu & Sawicki model (2007)

The Hu & Sawicki model:

$$f(R) = -m^2 \frac{c_1 (R/m^2)^n}{c_2 (R/m^2)^n + 1} \quad \text{with} \quad m^2 \equiv \Omega_\mathrm{m} H_0^2$$

$f(R)$ gravity - The Hu & Sawicki model (2007)

The Hu & Sawicki model:

$$f(R) = -m^2 \frac{c_1 (R/m^2)^n}{c_2 (R/m^2)^n + 1} \quad \text{with} \quad m^2 \equiv \Omega_\mathrm{m} H_0^2$$

In the high curvature limit (i.e. high density) :

$$\lim_{m^2/R \to 0} f(R) \simeq -\frac{c_1}{c_2} m^2 + \frac{c_1}{c_2^2} m^2 \left(\frac{m^2}{R} \right)^n \xrightarrow[m^2/R \to 0]{} -2\Lambda$$

We fix $c_1/c_2 = 6\Omega_\Lambda/\Omega_\mathrm{m}$ to match ΛCDM at high redshift ($R \to \infty$).

$f(R)$ gravity - The Hu & Sawicki model (2007)

The Hu & Sawicki model:

$$f(R) = -m^2 \frac{c_1 (R/m^2)^n}{c_2 (R/m^2)^n + 1} \quad \text{with} \quad m^2 \equiv \Omega_\mathrm{m} H_0^2$$

In the high curvature limit (i.e. high density) :

$$\lim_{m^2/R \to 0} f(R) \simeq -\frac{c_1}{c_2} m^2 + \frac{c_1}{c_2^2} m^2 \left(\frac{m^2}{R}\right)^n \xrightarrow[m^2/R \to 0]{} -2\Lambda$$

We fix $c_1/c_2 = 6\Omega_\Lambda/\Omega_\mathrm{m}$ to match ΛCDM at high redshift ($R \to \infty$).

$\longrightarrow$Two remaining free parameters: $c_1/c_2^2 \sim f_{R_0}$ and n.

N-body simulations in $f(R)$ gravity

N-body cosmological simulations for the Hu & Sawicki model with ECOSMOG [Li et al. 2012, Bose et al. 2017], modified version of RAMSES [Teyssier 2002].
$\longrightarrow$fast $f(R)$ solver but limited to $n = 1$, only free parameter: f_{R_0}.
$\longrightarrow$Multigrid solver for the Poisson eq. + Adaptive Mesh Refinement (AMR)

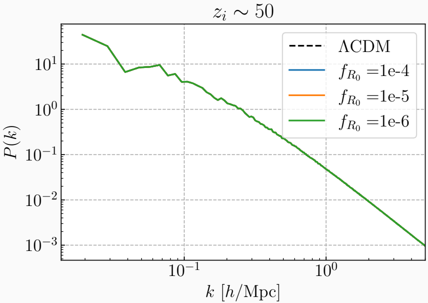
N-body simulations in $f(R)$ gravity

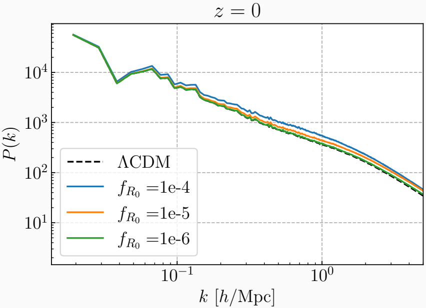
N-body cosmological simulations for the Hu & Sawicki model with ECOSMOG [Li et al. 2012, Bose et al. 2017], modified version of RAMSES [Teyssier 2002].
$\longrightarrow$fast $f(R)$ solver but limited to $n = 1$, only free parameter: f_{R_0}.
$\longrightarrow$Multigrid solver for the Poisson eq. + Adaptive Mesh Refinement (AMR)

Simulations:

- BoxSize: $(325\,\mathrm{Mpc}/h)^3$
- # particles: $512^3 \sim 135$ million
- Mass resolution: $m_\mathrm{p} \sim 10^{10}\mathrm{M_{sun}}h^{-1}$
- In ΛCDM, at $z = 0$, $P(k)$ at 1% for $k \lesssim 2h^{-1}$Mpc.
- CPU time: $5 - 20$ hours on 512 cores ($\sim 2 - 10$ times slower than ΛCDM)

Influence of $f(R)$ on the matter distribution

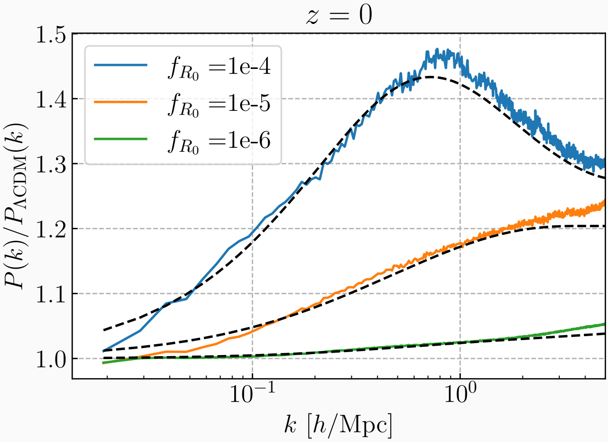




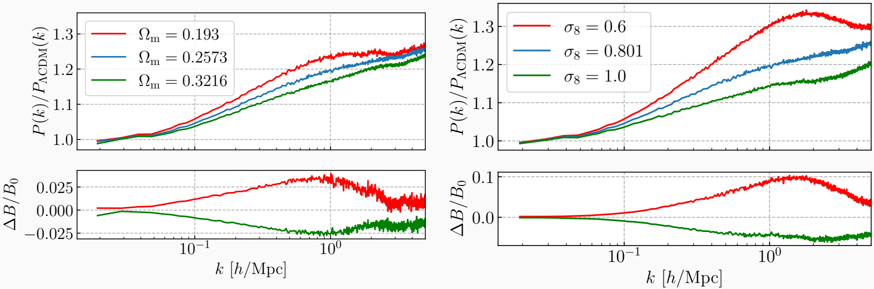
Left : Initial spectrum (beginning of the simulation).
Right : Final spectrum (end of the simulation).

$\longrightarrow$The spectrum is amplified by $f(R)$ gravity.
$\longrightarrow$This amplification is stronger at non-linear scales ($k > 0.1$).

Power spectrum boost - comparison with Winther 2020



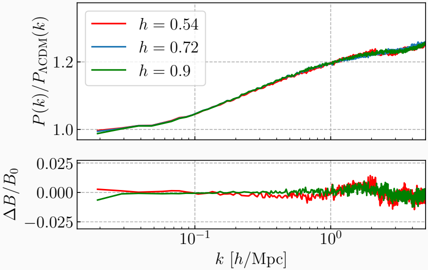
Influence of the cosmology on the boost due to $f(R)$

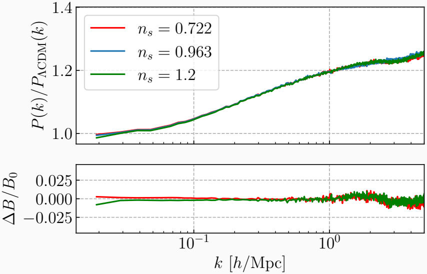


$\longrightarrow$Increasing the amount of matter (Ω_m) or the normalisation of the initial spectrum (σ_8) increases the density.

$\Rightarrow$ Screening of $f(R)$ more efficient $\Rightarrow$ weaker boost.

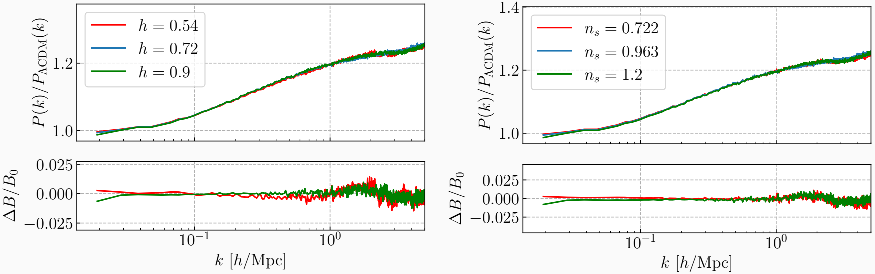
Influence of the cosmology on the boost due to $f(R)$





$\longrightarrow$The influence of h et n_s (and $\Omega_\mathrm{b} h^2$) on the power spectrum boost is negligible ($< 1\%$).

Influence of the cosmology on the boost due to $f(R)$



$\longrightarrow$The influence of h et n_s (and $\Omega_b h^2$) on the power spectrum boost is negligible ($< 1\%$).

$\Rightarrow$ Not necessary to take these parameters into account if we build an emulator for the boost !

Building an emulator to predict the effect of $f(R)$

Emulator: Interpolates between the results of simulations performed for different cosmological models. An emulator allows us to obtain fast predictions in the non-linear regime, unlike N-body simulations which are very slow.

Building an emulator to predict the effect of $f(R)$

Emulator: Interpolates between the results of simulations performed for different cosmological models. An emulator allows us to obtain fast predictions in the non-linear regime, unlike N-body simulations which are very slow.

Current emulators:

- [Winther 2020]: doesn't take into account the variation of the cosmology.
- [Ramachandra 2021]: takes into account the cosmology but calibrated using COLA simulations, less precise than N-body methods.
- [Arnold 2021]: very recent (few weeks ago) more accurate emulator, complementary to our work.

Building an emulator to predict the effect of $f(R)$

Emulator: Interpolates between the results of simulations performed for different cosmological models. An emulator allows us to obtain fast predictions in the non-linear regime, unlike N-body simulations which are very slow.

Current emulators:

- [Winther 2020]: doesn't take into account the variation of the cosmology.
- [Ramachandra 2021]: takes into account the cosmology but calibrated using COLA simulations, less precise than N-body methods.
- [Arnold 2021]: very recent (few weeks ago) more accurate emulator, complementary to our work.

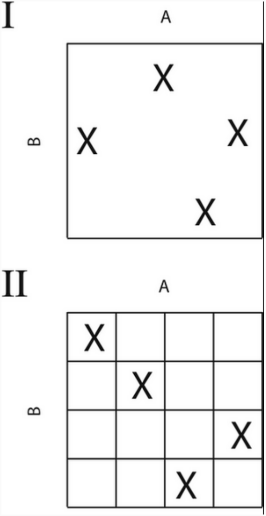
Objectives :

- Explore the parameter space: $\{\Omega\mathrm{m}, \sigma_8, f_{R_0}\}$.
- Build an emulator that predicts the boost $P(k)/P_{\Lambda\mathrm{CDM}}(k)$ as a function of these parameters.

Latin Hypercube Sampling

⟶**Example I:** each point is randomly sampled without taking into account the positions of the other points.

⟶**Example II:** each point is sampled so that it has no common coordinates with the already chosen points.

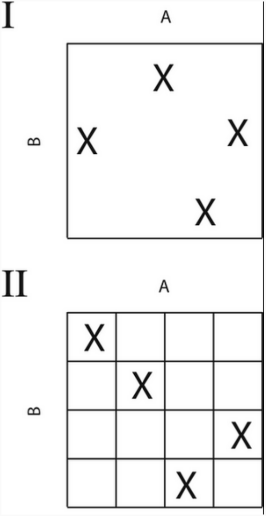


Latin Hypercube Sampling

⟶**Example I:** each point is randomly sampled without taking into account the positions of the other points.

⟶**Example II:** each point is sampled so that it has no common coordinates with the already chosen points.

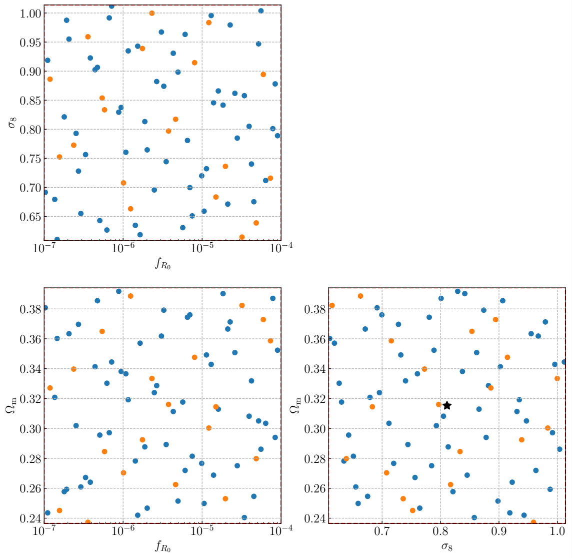
⇒ efficiently samples the parameter space
⟶important in order to minimise the number of simulations required to build an accurate emulator.



Sliced Latin Hypercube Sampling - Ba et. al 2015

⟶Modified algorithm of [Ba 2015]:

- maximises the distances between points,
- points distributed in sub-samples.



Gaussian Process Regression

Gaussian Process: collection of random variables, any finite subset of which follow a joint Gaussian distribution.

Gaussian Process Regression

Gaussian Process: collection of random variables, any finite subset of which follow a joint Gaussian distribution.

$\longrightarrow$Training data set $\{y_1, ..., y_n\}$ at training locations $\{x_1, ..., x_n\}$ follow a probability distribution:

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{0}, \mathbf{K})$$

where $\mathbf{K}_{ij} = k(x_i, x_j) \rightarrow$ **kernel** function.

Gaussian Process Regression

Gaussian Process: collection of random variables, any finite subset of which follow a joint Gaussian distribution.

$\longrightarrow$Training data set $\{y_1, ..., y_n\}$ at training locations $\{x_1, ..., x_n\}$ follow a probability distribution:

$$p(\mathbf{y}) = \mathcal{N}(\mathbf{0}, \mathbf{K})$$

where $\mathbf{K}_{ij} = k(x_i, x_j) \rightarrow$ **kernel** function.

$\longrightarrow$The kernel function describes how similar is the data at different locations. Simple example:

$$k_{\mathrm{RBF}}(x_i, x_j) = \exp\left(-\frac{(x_i - x_j)^2}{l^2}\right)$$

$\longrightarrow l$ is a free hyperparameter (inferred by likelihood maximization).

Gaussian Process Regression

We want to predict new data $\mathbf{y}^*$ at new locations $\mathbf{x}^*$. We can extract it from the joint probability distribution:

$$p(\mathbf{y}, \mathbf{y}^*) = \mathcal{N}(\mathbf{0}, K_{\text{joint}})$$

where K_{joint} is a function of $k(x, x), k(x, x^*), k(x^*, x^*)$.

Gaussian Process Regression

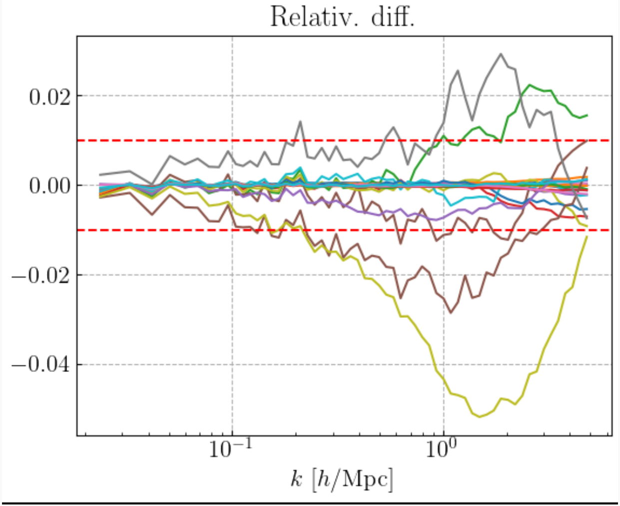
We want to predict new data $\mathbf{y}^*$ at new locations $\mathbf{x}^*$. We can extract it from the joint probability distribution:

$$p(\mathbf{y}, \mathbf{y}^*) = \mathcal{N}(\mathbf{0}, K_{\mathrm{joint}})$$

where K_{joint} is a function of $k(x, x), k(x, x^*), k(x^*, x^*)$.

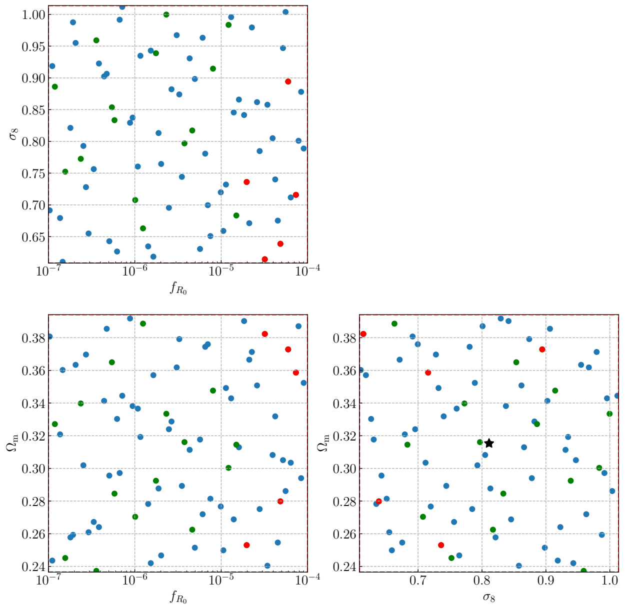
$\longrightarrow$We use the scikit-learn python library.

Testing the emulator

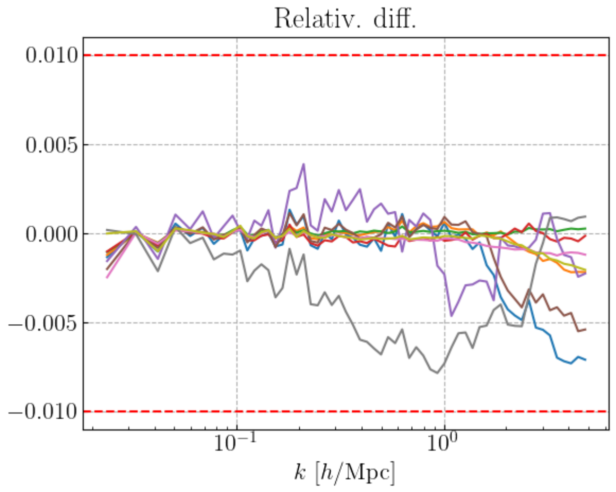


- Small average relative error: $<\epsilon>= 0.3\%$
- But a few models fail with large errors ($\sim 1 - 5\%$)

Testing the emulator

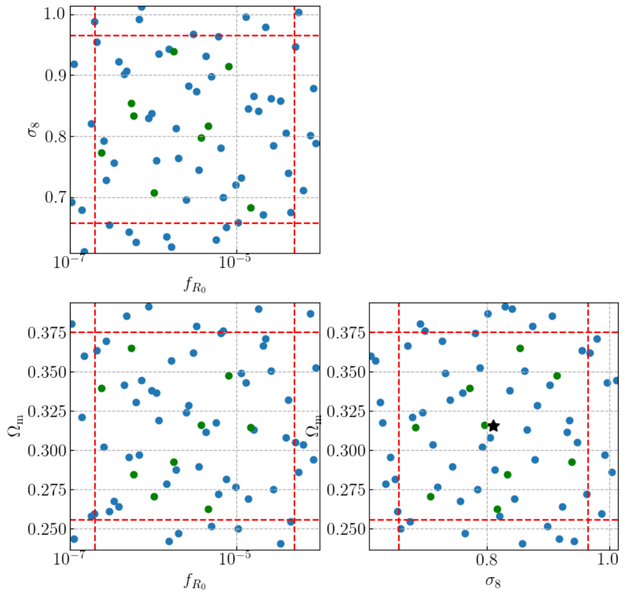


Reduced testing parameter space



- Very small average relative error: $<\epsilon>= 0.1\%$
- All models within 1% error bars !

Reduced testing parameter space



Conclusion

⟶We have studied the effect of the different cosmological parameters on the PS boost due to $f(R)$.
⟶We are in the process of building an emulator for the PS boost.
⟶We are also studying other quantities: we build a FoF halo catalogue at each snapshots.

- halo mass function
- halo power spectrum
- RSD (2PCF multipoles)
- halo bias

Conclusion

$\longrightarrow$We have studied the effect of the different cosmological parameters on the PS boost due to $f(R)$.

$\longrightarrow$We are in the process of building an emulator for the PS boost.

$\longrightarrow$We are also studying other quantities: we build a FoF halo catalogue at each snapshots.

- halo mass function
- halo power spectrum
- RSD (2PCF multipoles)
- halo bias

Long term perspectives:

- Study the effect of n.
- Can we consider and simulate more general MG models ?