Modified gravity simulations

On the road to build reliable emulators for modified gravity models

Christian Arnold

05.05.22, Marseille

ICC, Durham University

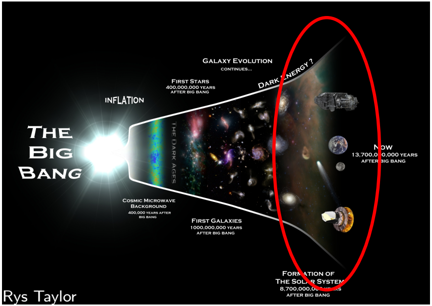
Institute for Computational Cosmology

European Research Council
Established by the European Commission

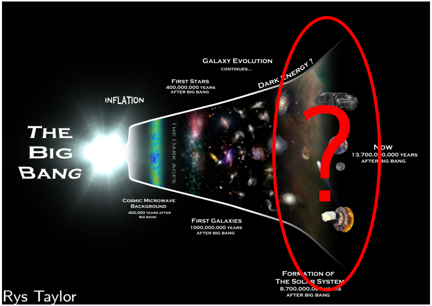
Durham University

Why modified gravity?

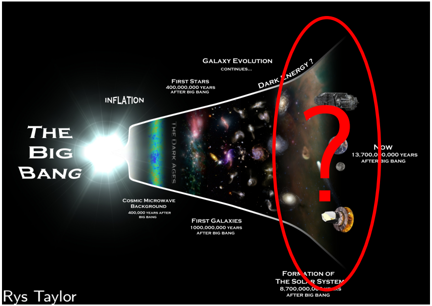
Why modified gravity?



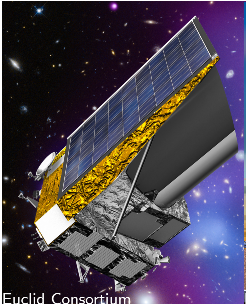
Why modified gravity?

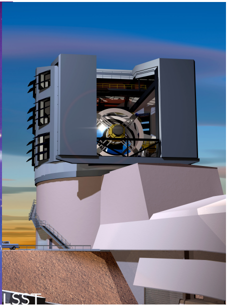


Why modified gravity?

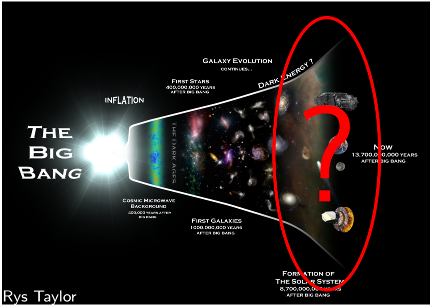


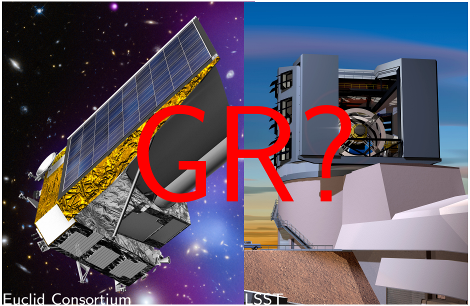
Rys Taylor


Euclid Consortium


LSST

Why modified gravity?





Modified gravity models in simulations

We want 'normal' gravity in our local environment $\Rightarrow$ screening mechanism as a classification method:

Modified gravity models in simulations

We want 'normal' gravity in our local environment ⇒ screening mechanism as a classification method:

- Chameleon: f(R), Symmetron, JBD (Jordan-Brans-Dicke)

Modified gravity models in simulations

We want 'normal' gravity in our local environment $\Rightarrow$ screening mechanism as a classification method:

- Chameleon: f(R), Symmetron, JBD (Jordan-Brans-Dicke)
- Vainshtain: DGP, nDGP, Galileon

Modified gravity models in simulations

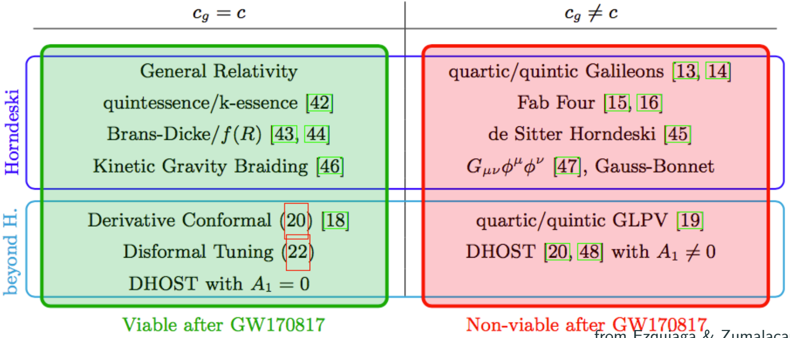
We want 'normal' gravity in our local environment $\Rightarrow$ screening mechanism as a classification method:

- Chameleon: f(R), Symmetron, JBD (Jordan-Brans-Dicke)
- Vainshtain: DGP, nDGP, Galileon
- K-mouflage, γ-gravity, δ-gravity, ...

Modified gravity models in simulations

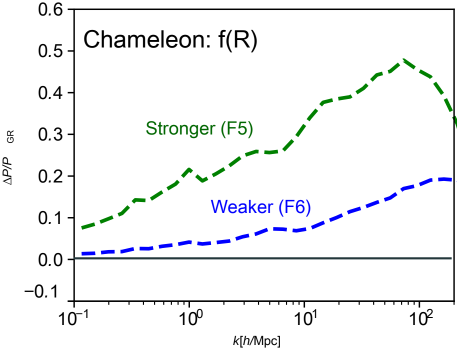
We want ‘normal’ gravity in our local environment ⇒ screening mechanism as a classification method:

- Chameleon: f(R), Symmetron, JBD (Jordan-Brans-Dicke)
- Vainshtain: DGP, nDGP, Galileon
- K-mouflage, γ-gravity, δ-gravity, ...

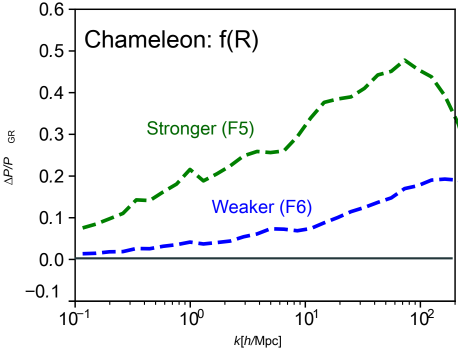


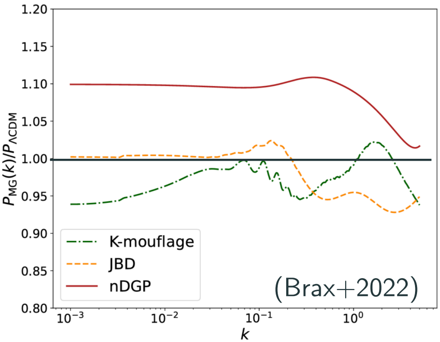
from Ezquiaga & Zumalacarregui (2017)

Modified gravity models: Effect on the matter power spectrum

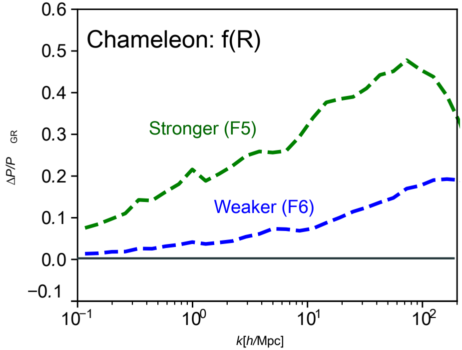


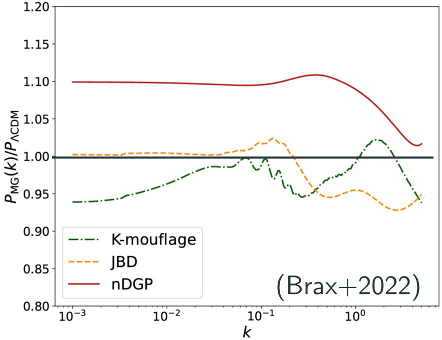
Modified gravity models: Effect on the matter power spectrum





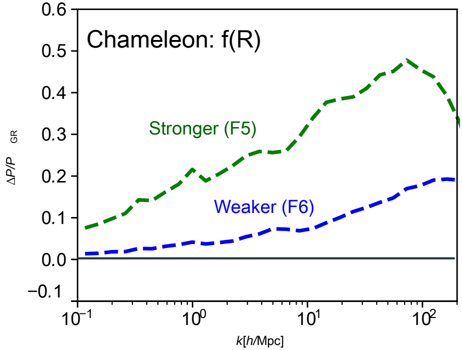
Modified gravity models: Effect on the matter power spectrum

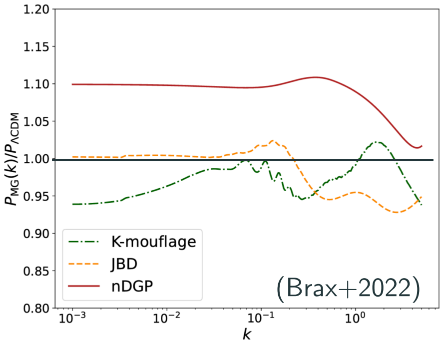




$\Rightarrow$ need predictions for many models/parameters if we want to analyse observational data beyond GR

Modified gravity models: Effect on the matter power spectrum





⇒ need predictions for many models/parameters if we want to analyse observational data beyond GR

⇒ simulations for MG

Modified Gravity Simulation Codes

Classic codes	Fast / approximate codes	Emulators
ECOSMOG f(R), nDGP, Galileon	MG-GLAM f(R), nDGP, ...	FORGE f(R)
ISIS f(R), Symmetron	MG-HALOFIT/CAMB f(R)	REACT f(R), nDGP
MG-GADGET f(R) ($+\nu$)	MG-COLA f(R), DGP, Symmetron, JBD	BRIDGE nDGP
AREPO f(R), nDGP ($+\nu$)	MG-EVOLUTION f(R), nDGP	

Classic Codes

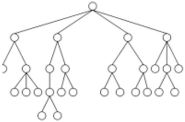
- Screening mechanisms make the equations to solve very non-linear

$$\nabla^2 f_R = \frac{1}{3}\left[\delta R(f_R) - 8\pi G \delta\rho\right]$$

Classic Codes

- Screening mechanisms make the equations to solve very non-linear
- but: most traditional gravity solvers (oct-tree, particle mesh, ...) require linearity in the equations

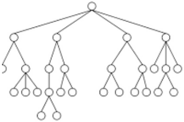
$$\nabla^2 f_R = \frac{1}{3}\left[\delta R(f_R) - 8\pi G \delta\rho\right]$$

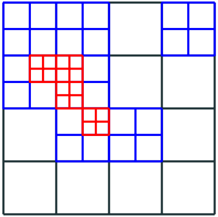


Classic Codes

- Screening mechanisms make the equations to solve very non-linear
- but: most traditional gravity solvers (oct-tree, particle mesh, ...) require linearity in the equations
- use iterative solvers on grids

$$\nabla^2 f_R = \frac{1}{3}\left[\delta R(f_R) - 8\pi G\delta\rho\right]$$

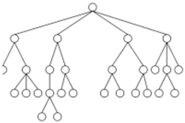


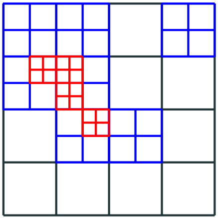


Classic Codes

- Screening mechanisms make the equations to solve very non-linear
- but: most traditional gravity solvers (oct-tree, particle mesh, ...) require linearity in the equations
- use iterative solvers on grids
- gain resolution and efficiency by AMR and multigrid methods

$$\nabla^2 f_R = \frac{1}{3}\left[\delta R(f_R) - 8\pi G\delta\rho\right]$$

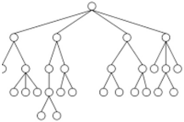


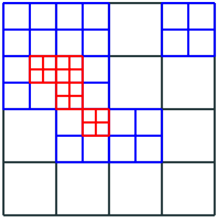


Classic Codes

- Screening mechanisms make the equations to solve very non-linear
- but: most traditional gravity solvers (oct-tree, particle mesh, ...) require linearity in the equations
- use iterative solvers on grids
- gain resolution and efficiency by AMR and multigrid methods
- can simulate hydrodynamics or even galaxy formation (e.g. SHIBONE) and neutrinos

$$\nabla^2 f_R = \frac{1}{3}\left[\delta R(f_R) - 8\pi G \delta\rho\right]$$

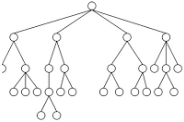


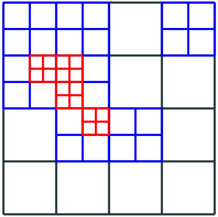


Classic Codes

- Screening mechanisms make the equations to solve very non-linear
- but: most traditional gravity solvers (oct-tree, particle mesh, ...) require linearity in the equations
- use iterative solvers on grids
- gain resolution and efficiency by AMR and multigrid methods
- can simulate hydrodynamics or even galaxy formation (e.g. SHIBONE) and neutrinos
- slow and computationally expensive if many runs are required

$$\nabla^2 f_R = \frac{1}{3}\left[\delta R(f_R) - 8\pi G \delta\rho\right]$$





Modified Gravity Simulation Codes

Classic codes	Fast / approximate codes	Emulators
ECOSMOG f(R), nDGP, Galileon	MG-GLAM f(R), nDGP, ...	FORGE f(R)
ISIS f(R), Symmetron	MG-HALOFIT/CAMB f(R)	REACT f(R), nDGP
MG-GADGET f(R) $(+\nu)$	MG-COLA f(R), DGP, Symmetron, JBD	BRIDGE nDGP
AREPO f(R), nDGP $(+\nu)$	MG-EVOLUTION f(R), nDGP	

Fast/approximate codes

- Neglect smaller scale effects (PM) or use halo model (e.g. HALOFIT)

Fast/approximate codes

- Neglect smaller scale effects (PM) or use halo model (e.g. HALOFIT)
- parametrised approach: can simulate many different MG-models

Fast/approximate codes

- Neglect smaller scale effects (PM) or use halo model (e.g. HALOFIT)
- parametrised approach: can simulate many different MG-models
- factors of 100s to 1000s faster than classic codes

Fast/approximate codes

- Neglect smaller scale effects (PM) or use halo model (e.g. HALOFIT)
- parametrised approach: can simulate many different MG-models
- factors of 100s to 1000s faster than classic codes
- unsuitable for e.g. halo profiles

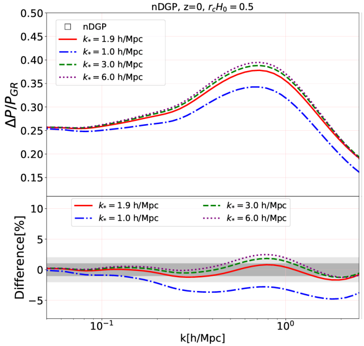
Fast/approximate codes

- Neglect smaller scale effects (PM) or use halo model (e.g. HALOFIT)
- parametrised approach: can simulate many different MG-models
- factors of 100s to 1000s faster than classic codes
- unsuitable for e.g. halo profiles
- but great for HMF or power spectra

Fast/approximate codes

- Neglect smaller scale effects (PM) or use halo model (e.g. HALOFIT)
- parametrised approach: can simulate many different MG-models
- factors of 100s to 1000s faster than classic codes
- unsuitable for e.g. halo profiles
- but great for HMF or power spectra

MG-EVOLUTION



Hassani&Lombriser(2020)

Modified Gravity Simulation Codes

Classic codes	Fast / approximate codes	Emulators
ECOSMOG f(R), nDGP, Galileon	MG-GLAM f(R), nDGP, ...	FORGE f(R)
ISIS f(R), Symmetron	MG-HALOFIT/CAMB f(R)	REACT f(R), nDGP
MG-GADGET f(R) $(+\nu)$	MG-COLA f(R), DGP, Symmetron, JBD	BRIDGE nDGP
AREPO f(R), nDGP $(+\nu)$	MG-EVOLUTION f(R), nDGP	

Emulators

- Tailored to a certain observable

Emulators

- Tailored to a certain observable
- Fill the parameter space between simulation results

Emulators

- Tailored to a certain observable
- Fill the parameter space between simulation results
- Much faster than approximate codes: running CAMB/HALOFIT still takes $\mathcal{O}(1)\mathrm{s}$, FORGE takes $\mathcal{O}(1)\mathrm{ms}$

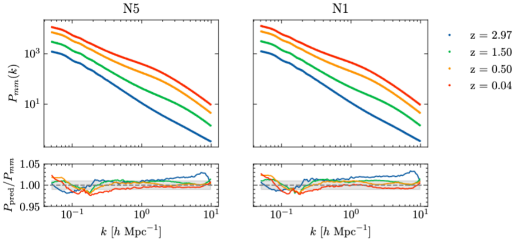
Emulators

- Tailored to a certain observable
- Fill the parameter space between simulation results
- Much faster than approximate codes: running CAMB/Halofit still takes $\mathcal{O}(1)\mathrm{s}$, FORGE takes $\mathcal{O}(1)\mathrm{ms}$
- much faster for MCMCs

Emulators

- Tailored to a certain observable
- Fill the parameter space between simulation results
- Much faster than approximate codes: running CAMB/HALOFIT still takes $\mathcal{O}(1)$s, FORGE takes $\mathcal{O}(1)$ms
- much faster for MCMCs

BRIDGE (preliminary)



Modified Gravity Simulation Codes

Classic codes	Fast / approximate codes	Emulators
ECOSMOG f(R), nDGP, Galileon	MG-GLAM f(R), nDGP, ...	FORGE f(R)
ISIS f(R), Symmetron	MG-HALOFIT/CAMB f(R)	REACT f(R), nDGP
MG-GADGET f(R) $(+\nu)$	MG-COLA f(R), DGP, Symmetron, JBD	BRIDGE nDGP
AREPO f(R), nDGP $(+\nu)$	MG-EVOLUTION f(R), nDGP	

FORGE – f(R)-gravity

Add another term to the action of GR:

$$S = \int \mathrm{d}^4x \sqrt{-g}\left[\frac{R + f(R)}{16\pi G} + \mathcal{L}_m\right]$$

FORGE – f(R)-gravity

Add another term to the action of GR:

$$S = \int \mathrm{d}^4x \sqrt{-g} \left[\frac{R + f(R)}{16\pi G} + \mathcal{L}_m \right]$$

Newtonian limit:

$$f_R = \frac{\mathrm{d}f(R)}{\mathrm{d}R}$$

$$\nabla^2 \Phi = \frac{4}{3} \times 4\pi G \delta\rho - \frac{1}{6} \delta R \qquad \nabla^2 f_R = \frac{1}{3} \left(\delta R - 8\pi G \delta\rho \right)$$

FORGE – f(R)-gravity

Add another term to the action of GR:

$$S = \int \mathrm{d}^4x\sqrt{-g}\left[\frac{R+f(R)}{16\pi G} + \mathcal{L}_m\right]$$

Newtonian limit:

$$f_R = \frac{\mathrm{d}f(R)}{\mathrm{d}R}$$

$$\nabla^2\Phi = \frac{4}{3} \times 4\pi G\delta\rho - \frac{1}{6}\delta R \qquad \nabla^2 f_R = \frac{1}{3}\left(\delta R - 8\pi G\delta\rho\right)$$

$$f(R) = -m^2\frac{c_1\left(\frac{R}{m^2}\right)^n}{c_2\left(\frac{R}{m^2}\right)^n+1} \qquad m^2 \equiv H_0^2\Omega_m$$

Add another term to the action of GR:

$$S = \int \mathrm{d}^4x \sqrt{-g} \left[\frac{R + f(R)}{16\pi G} + \mathcal{L}_m \right]$$

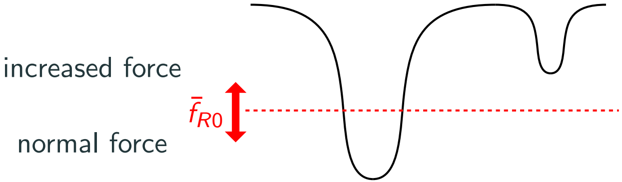
Newtonian limit:

$$f_R = \frac{\mathrm{d}f(R)}{\mathrm{d}R}$$

$$\nabla^2 \Phi = \frac{4}{3} \times 4\pi G \delta\rho - \frac{1}{6}\delta R \qquad \nabla^2 f_R = \frac{1}{3}\left(\delta R - 8\pi G \delta\rho\right)$$

$$f(R) = -m^2 \frac{c_1 \left(\frac{R}{m^2}\right)^n}{c_2 \left(\frac{R}{m^2}\right)^n + 1} \qquad m^2 \equiv H_0^2 \Omega_m$$

Chameleon screening:



FORGE – f(R)-gravity

Add another term to the action of GR:

$$S = \int \mathrm{d}^4x\sqrt{-g}\left[\frac{R+f(R)}{16\pi G} + \mathcal{L}_m\right]$$

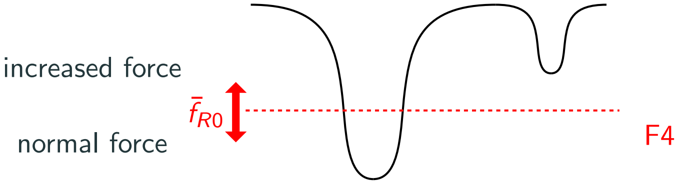
Newtonian limit:

$$f_R = \frac{\mathrm{d}f(R)}{\mathrm{d}R}$$

$$\nabla^2\Phi = \frac{4}{3}\times 4\pi G\delta\rho - \frac{1}{6}\delta R \qquad \nabla^2 f_R = \frac{1}{3}\left(\delta R - 8\pi G\delta\rho\right)$$

$$f(R) = -m^2\frac{c_1\left(\frac{R}{m^2}\right)^n}{c_2\left(\frac{R}{m^2}\right)^n+1} \qquad m^2 \equiv H_0^2\Omega_m$$

Chameleon screening:



FORGE – f(R)-gravity

Add another term to the action of GR:

$$S = \int \mathrm{d}^4x \sqrt{-g} \left[\frac{R + f(R)}{16\pi G} + \mathcal{L}_m \right]$$

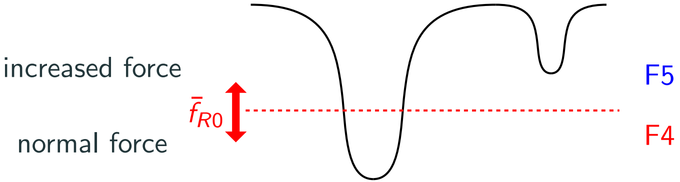
Newtonian limit:

$$f_R = \frac{\mathrm{d}f(R)}{\mathrm{d}R}$$

$$\nabla^2 \Phi = \frac{4}{3} \times 4\pi G \delta\rho - \frac{1}{6}\delta R \qquad \nabla^2 f_R = \frac{1}{3}\left(\delta R - 8\pi G \delta\rho\right)$$

$$f(R) = -m^2 \frac{c_1 \left(\frac{R}{m^2}\right)^n}{c_2 \left(\frac{R}{m^2}\right)^n + 1} \qquad m^2 \equiv H_0^2 \Omega_m$$

Chameleon screening:



FORGE – f(R)-gravity

Add another term to the action of GR:

$$S = \int \mathrm{d}^4x \sqrt{-g} \left[\frac{R + f(R)}{16\pi G} + \mathcal{L}_m \right]$$

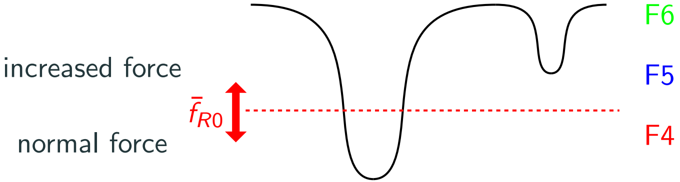
Newtonian limit:

$$f_R = \frac{\mathrm{d}f(R)}{\mathrm{d}R}$$

$$\nabla^2 \Phi = \frac{4}{3} \times 4\pi G \delta\rho - \frac{1}{6}\delta R \qquad \nabla^2 f_R = \frac{1}{3}\left(\delta R - 8\pi G \delta\rho\right)$$

$$f(R) = -m^2 \frac{c_1 \left(\frac{R}{m^2}\right)^n}{c_2 \left(\frac{R}{m^2}\right)^n + 1} \qquad m^2 \equiv H_0^2 \Omega_m$$

Chameleon screening:



The FORGE simulation suite

- Vary Ω_m, $\Omega_\Lambda = 1 - \Omega_m$, h, σ_8 and $\bar{f}_{\rm R0}$ while keeping the other parameters fixed

The FORGE simulation suite

- Vary Ω_m, $\Omega_\Lambda = 1 - \Omega_m$, h, σ_8 and $\bar{f}_{\rm R0}$ while keeping the other parameters fixed
- Use a Latin Hypercube to sample the parameter space with 50 'nodes' and in total 200 MG simulations

The FORGE simulation suite

- Vary Ω_m, $\Omega_\Lambda = 1 - \Omega_m$, h, σ_8 and $\bar{f}_{\mathrm{R0}}$ while keeping the other parameters fixed
- Use a Latin Hypercube to sample the parameter space with 50 'nodes' and in total 200 MG simulations
- 2 DMO simulation pairs ($= 4$) per node run with AREPO; minimise large scale variance in each pair on average

The FORGE simulation suite

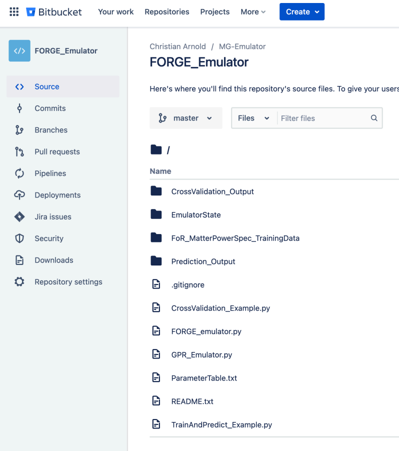
- Vary Ω_m, $\Omega_\Lambda = 1 - \Omega_m$, h, σ_8 and $\bar{f}_{\rm R0}$ while keeping the other parameters fixed
- Use a Latin Hypercube to sample the parameter space with 50 'nodes' and in total 200 MG simulations
- 2 DMO simulation pairs ($= 4$) per node run with AREPO; minimise large scale variance in each pair on average
- High-res pair with 1024^3 DM particles in $500\,\mathrm{Mpc}/h$ box

The FORGE simulation suite

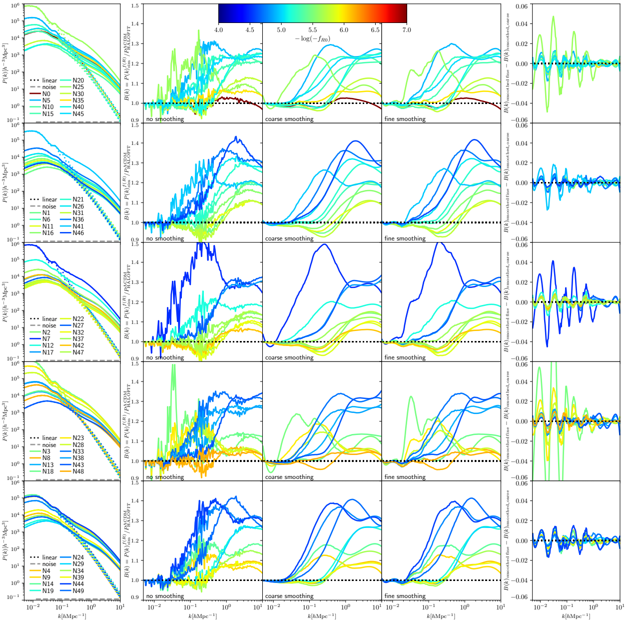
- Vary Ω_m, $\Omega_\Lambda = 1 - \Omega_m$, h, σ_8 and $\bar{f}_{\rm R0}$ while keeping the other parameters fixed
- Use a Latin Hypercube to sample the parameter space with 50 'nodes' and in total 200 MG simulations
- 2 DMO simulation pairs ($= 4$) per node run with AREPO; minimise large scale variance in each pair on average
- High-res pair with 1024^3 DM particles in $500\,\mathrm{Mpc}/h$ box
- Low-res pair with 512^3 DM particles in $1500\,\mathrm{Mpc}/h$ box

The FORGE simulation suite

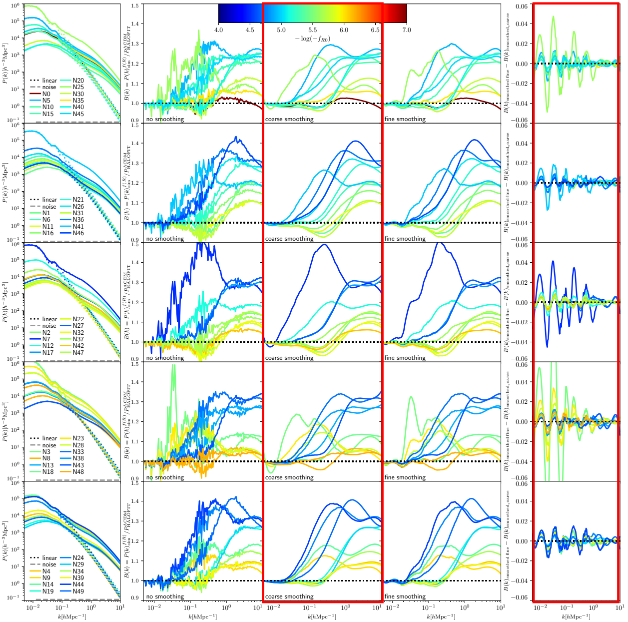
- Vary Ω_m, $\Omega_\Lambda = 1 - \Omega_m$, h, σ_8 and $\bar{f}_{\rm R0}$ while keeping the other parameters fixed
- Use a Latin Hypercube to sample the parameter space with 50 'nodes' and in total 200 MG simulations
- 2 DMO simulation pairs ($= 4$) per node run with AREPO; minimise large scale variance in each pair on average
- High-res pair with 1024^3 DM particles in $500\,\mathrm{Mpc}/h$ box
- Low-res pair with 512^3 DM particles in $1500\,\mathrm{Mpc}/h$ box



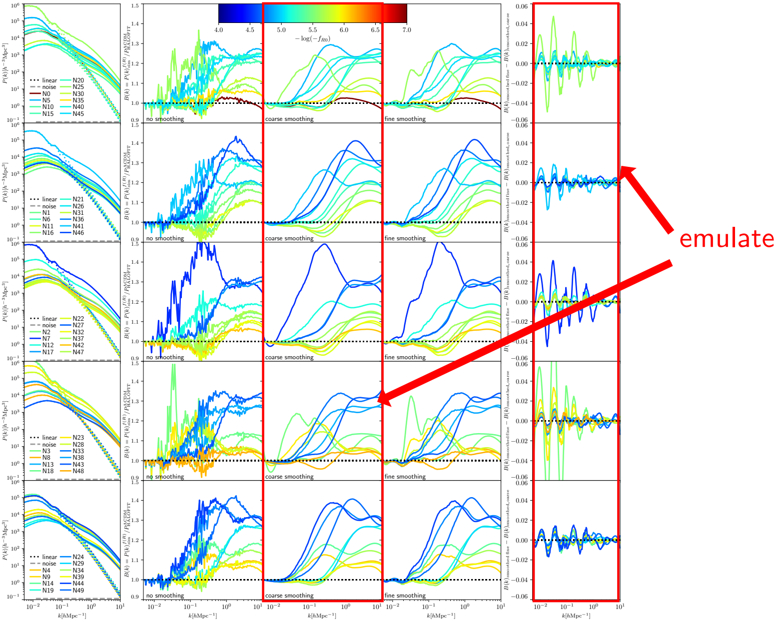
3D matter power spectrum emulator



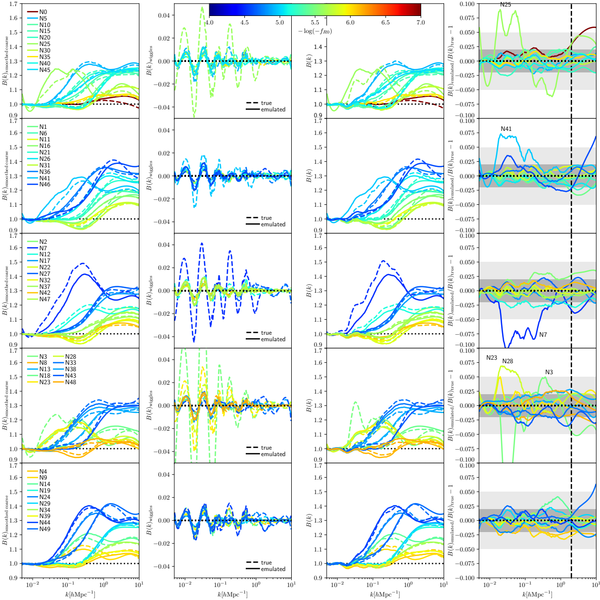
3D matter power spectrum emulator



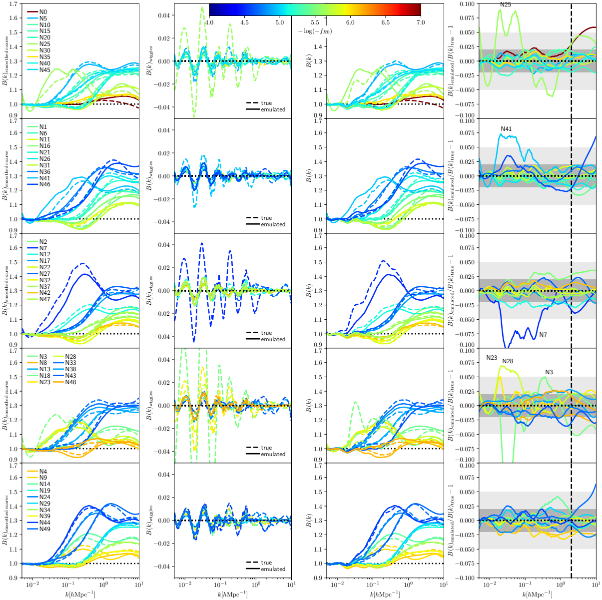
3D matter power spectrum emulator

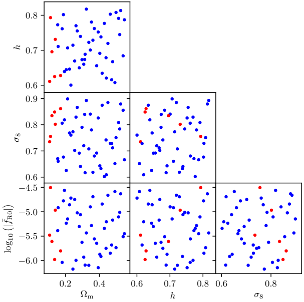


Emulator cross validation

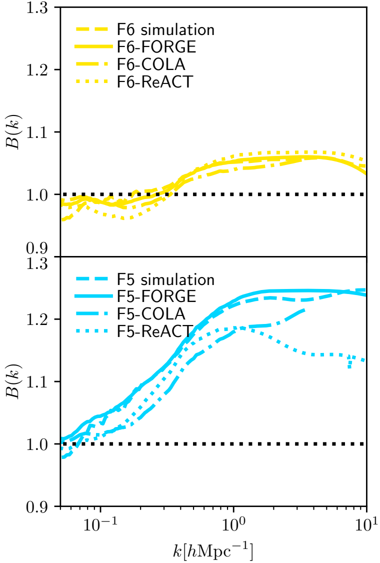


Emulator cross validation



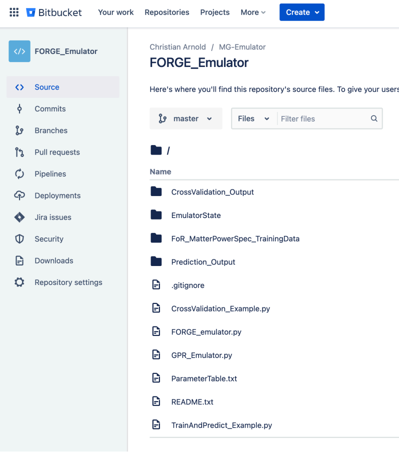


Making a prediction for independent simulations and comparing to other emulators



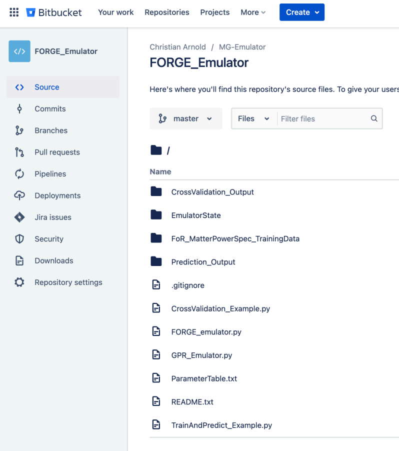
FORGE - what is there to come?

- BRIDGE - nDGP emulator (Cesar Hernandez-Aguayo, Carol Cuesta-Lazaro)



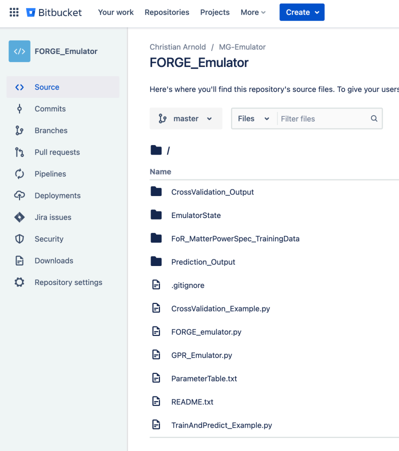
FORGE - what is there to come?

- BRIDGE - nDGP emulator (Cesar Hernandez-Aguayo, Carol Cuesta-Lazaro)
- MCMCs with HALOFIT are dead slow - HALOFIT emulator



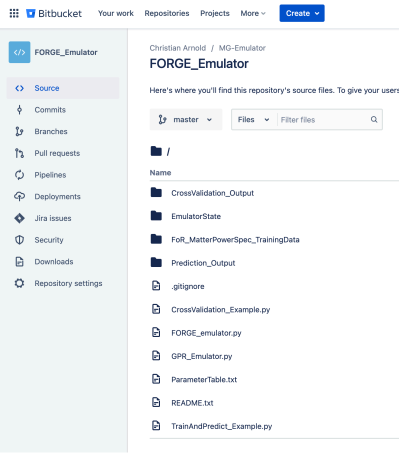
FORGE - what is there to come?

- BRIDGE - nDGP emulator (Cesar Hernandez-Aguayo, Carol Cuesta-Lazaro)
- MCMCs with HALOFIT are dead slow - HALOFIT emulator
- Machine Learning based emulators with PyTorch (better for 'strange' behaviour of models)



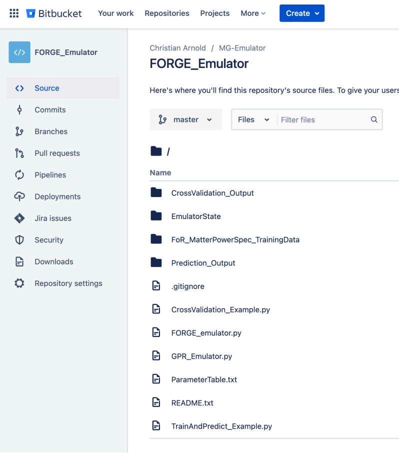
FORGE - what is there to come?

- BRIDGE - nDGP emulator (Cesar Hernandez-Aguayo, Carol Cuesta-Lazaro)
- MCMCs with HALOFIT are dead slow - HALOFIT emulator
- Machine Learning based emulators with PyTorch (better for 'strange' behaviour of models)
- HMF emulators



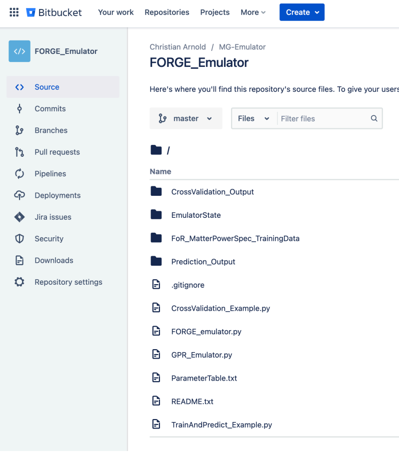
FORGE - what is there to come?

- BRIDGE - nDGP emulator (Cesar Hernandez-Aguayo, Carol Cuesta-Lazaro)
- MCMCs with HALOFIT are dead slow - HALOFIT emulator
- Machine Learning based emulators with PyTorch (better for 'strange' behaviour of models)
- HMF emulators
- Lensing power spectrum emulators (Joachim Harnois Deraps)



FORGE - what is there to come?

- BRIDGE - nDGP emulator (Cesar Hernandez-Aguayo, Carol Cuesta-Lazaro)
- MCMCs with HALOFIT are dead slow - HALOFIT emulator
- Machine Learning based emulators with PyTorch (better for 'strange' behaviour of models)
- HMF emulators
- Lensing power spectrum emulators (Joachim Harnois Deraps)
- Larger simulation suites for training: MG-GLAM



MG-GLAM

Fast full N-body simulations of generic modified gravity: conformal coupling models

arXiv:2110.00328

Cheng-Zong Ruan,[a] César Hernández-Aguayo,[b,c] Baojiu Li,[a] Christian Arnold,[a] Carlton M. Baugh,[a] Anatoly Klypin,[d] and Francisco Prada[e]

[a]Institute for Computational Cosmology, Department of Physics, Durham University, South Road, Durham DH1 3LE, UK

- Based on GLAM (Kyplin&Prada 2018)

MG-GLAM

Fast full N-body simulations of generic modified gravity: conformal coupling models

arXiv:2110.00328

Cheng-Zong Ruan,[a] César Hernández-Aguayo,[b,c] Baojiu Li,[a] Christian Arnold,[a] Carlton M. Baugh,[a] Anatoly Klypin,[d] and Francisco Prada[e]

[a]Institute for Computational Cosmology, Department of Physics, Durham University, South Road, Durham DH1 3LE, UK

- Based on GLAM (Kyplin&Prada 2018)
- Particle Mesh (fixed grid) code

MG-GLAM

Fast full N-body simulations of generic modified gravity: conformal coupling models

arXiv:2110.00328

Cheng-Zong Ruan,[a] César Hernández-Aguayo,[b,c] Baojiu Li,[a] Christian Arnold,[a] Carlton M. Baugh,[a] Anatoly Klypin,[d] and Francisco Prada[e]

[a]Institute for Computational Cosmology, Department of Physics, Durham University, South Road, Durham DH1 3LE, UK

- Based on GLAM (Kyplin&Prada 2018)
- Particle Mesh (fixed grid) code
- Sacrifice resolution for speed

MG-GLAM

Fast full N-body simulations of generic modified gravity: conformal coupling models

arXiv:2110.00328

Cheng-Zong Ruan,[a] César Hernández-Aguayo,[b,c] Baojiu Li,[a] Christian Arnold,[a] Carlton M. Baugh,[a] Anatoly Klypin,[d] and Francisco Prada[e]

[a]Institute for Computational Cosmology, Department of Physics, Durham University, South Road, Durham DH1 3LE, UK

- Based on GLAM (Kyplin&Prada 2018)
- Particle Mesh (fixed grid) code
- Sacrifice resolution for speed
- OpenMP parallelised (needs a cluster with reasonably large nodes)

MG-GLAM

Fast full N-body simulations of generic modified gravity: conformal coupling models

arXiv:2110.00328

Cheng-Zong Ruan,[a] César Hernández-Aguayo,[b,c] Baojiu Li,[a] Christian Arnold,[a] Carlton M. Baugh,[a] Anatoly Klypin,[d] and Francisco Prada[e]

[a]Institute for Computational Cosmology, Department of Physics, Durham University, South Road, Durham DH1 3LE, UK

- Based on GLAM (Kyplin&Prada 2018)
- Particle Mesh (fixed grid) code
- Sacrifice resolution for speed
- OpenMP parallelised (needs a cluster with reasonably large nodes)
- 100 times faster than AREPO and 300 times faster than ECOSMOG for f(R) gravity

MG-GLAM

Fast full N-body simulations of generic modified gravity: conformal coupling models

arXiv:2110.00328

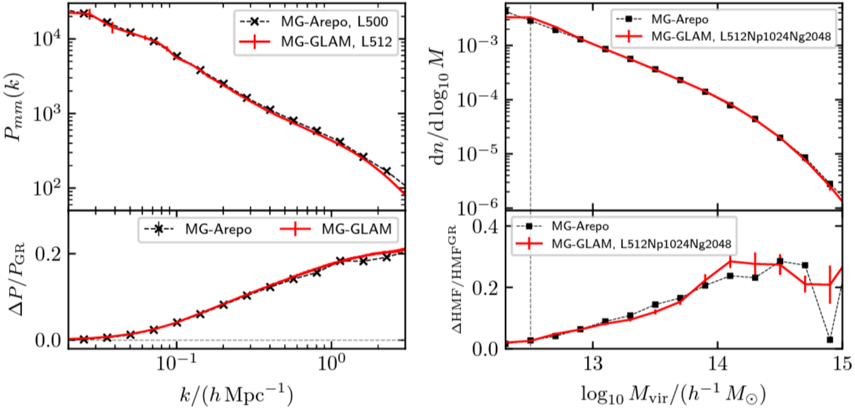
Cheng-Zong Ruan,[a] César Hernández-Aguayo,[b,c] Baojiu Li,[a] Christian Arnold,[a] Carlton M. Baugh,[a] Anatoly Klypin,[d] and Francisco Prada[e]

[a]Institute for Computational Cosmology, Department of Physics, Durham University, South Road, Durham DH1 3LE, UK

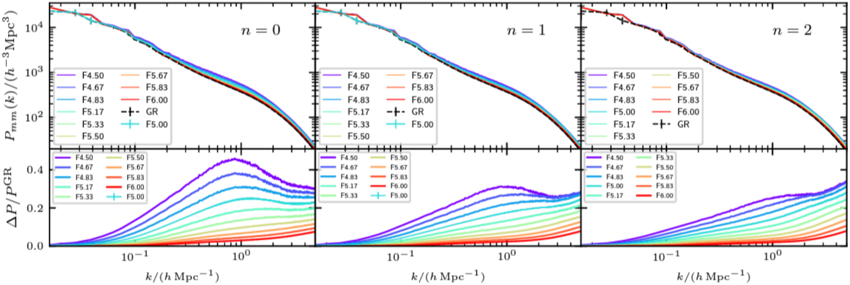
- Based on GLAM (Kyplin&Prada 2018)
- Particle Mesh (fixed grid) code
- Sacrifice resolution for speed
- OpenMP parallelised (needs a cluster with reasonably large nodes)
- 100 times faster than AREPO and 300 times faster than ECOSMOG for f(R) gravity
- f(R) run with 2048^3 particles in a $1\mathrm{Gpc}/h$ box with a 4096^3 grid takes $\approx 27\mathrm{h}$ on one cosma8 node (128 cores)

MG-GLAM - accuracy

Compare $\approx 500\mathrm{Mpc}/h$ boxes with 1024^3 particles of MG-GLAM (2048^3 grid) and AREPO



MG-GLAM - power spectrum



Summary

- Fully non-linear simulation codes can provide high resolution images of MG-universes but are very slow

Summary

- Fully non-linear simulation codes can provide high resolution images of MG-universes but are very slow
- Data analysis requires fast predictions for a certain parameter set

Summary

- Fully non-linear simulation codes can provide high resolution images of MG-universes but are very slow
- Data analysis requires fast predictions for a certain parameter set
- Emulators can supply this but ideally need big simulation set to train on

Summary

- Fully non-linear simulation codes can provide high resolution images of MG-universes but are very slow
- Data analysis requires fast predictions for a certain parameter set
- Emulators can supply this but ideally need big simulation set to train on
- Fast simulation methods can now offer the speed and accuracy for this also for MG-models