`Gevolution`: general simulations and void science

Justyna Borkowska
Boudewijn Roukema
Ahsan Nazer

Toruń Centre for Astronomy, Nicolaus Copernicus University

Toruń, 16.07.2019

What gevolution is and isn't

What it is and what it can do:

- particle-mesh N-body C++ code
- weak-field approximation scheme
- small in size (0.5 Mb without outputs) and relatively fast
- front/back end potential
- allows a lot of cosmological background customization
- linked to CLASS (and other IC generators)
- compatible with Gadget-2
- ray-tracing (new feature in 1.2)

Downsides:

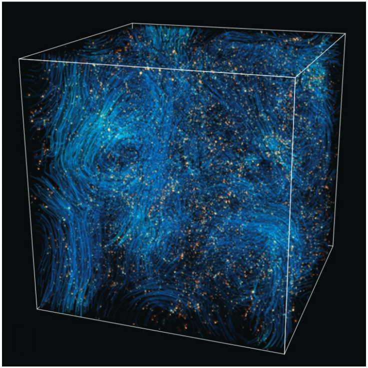
- only choice of background is flat FLRW
- currently not AMR
- lack of documentation (limited manual)
- only collisionless dust (no baryonic physics)
- limited compatibility
- weak-field approximation

Inputs and settings

- IC GENERATOR INPUT
  - power spectrum file (using CMB parameters)
  - transfer function file (CLASS format)
- IC GENERATOR
  - taken from CLASS
  - generated "on the fly"
  - read from disc (resuming a simulation)
  - an IC generator for ultra-relativistic particle species
- PARTICLE TEMPLATE (Gadget-2 format)
- SIMULATION BOX SETTINGS
  - box size
  - tiling factor
  - time step limit
  - time resolution (Courant factor)
  - initial redshift
  - ...

- PRIMORDIAL PERTURBATIONS PARAMETERS
  - spectral amplitude A_s
  - spectral index n_s
  - pivot scale k
- COSMOLOGICAL PARAMETERS
  - Ω's (baryonic, CDM, other particle species)
  - Hubble parameter (h)
- CHOICES OF GRAVITY THEORY
  - Newtonian
  - GR
- OUTPUT SETTINGS
  - power spectrum snapshots
  - particle snapshots
  - lightcone snapshots
  - redshifts of snapshots
- LIGHTCONE SETTINGS

Output



Frame dragging by vortical matter flows (perturbation B_i) on a simulation volume of (512 Mpc/h)^3; orange spots mark dark matter halos [Adamek et.al., Nature Phys. 12 (2016) 346–349]

- outputs (amount, quantities) depend on the simulation settings
- possible formats: HDF5 files, Gadget-2 particle files
- possible output data:
  - metric data ($h_{ij}, B_i, \phi, \chi = \phi - \psi$)
  - particle positions
  - lightcone(s) (`gevolution 1.2`)
  - total T^0_0 and T^i_j
  - total momentum density
  - velocity
  - Newtonian case: ρ and ψ
- output is in Poisson gauge
- for each quantity, a power spectrum can be saved as a snapshot

Theoretical basis

Poisson gauge

$$\mathrm{d}s^2 = a^2(\tau)\left[-(1+2\psi)\mathrm{d}\tau^2 - 2B_i\mathrm{d}x\mathrm{d}\tau + (1-2\phi)\delta_{ij}\mathrm{d}x^i\mathrm{d}x^j + h_{ij}\mathrm{d}x^i\mathrm{d}x^j\right]$$

gauge conditions: $\delta^{ij}B_{i,j} = \delta^{ij}h_{ij} = \delta^{jk}h_{ij,k} = 0$

Line of reasoning:

1. take initial data and calculate initial positions, velocities and other values in the simulation (on-the-fly IC generation)
2. calculate stress-energy tensor
3. calculate metric components
4. output files (if specified at that time step)
5. update particle positions
6. increment the time step
7. repeat steps 2-6 up to the final output redshift

details: Adamek et.al. (2016)

Numerical aspects

LATfield2

- developed for classical field theory [David Daverio et.al.]
- partial differential equations solver
- parallelisation
- handles FFT
- separates and connects fields and particles
- gevolution: keeping numerical aspects “behind the scenes”

Useful links

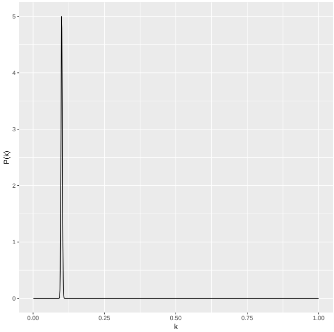
- official website
- main paper
- reference and documentation

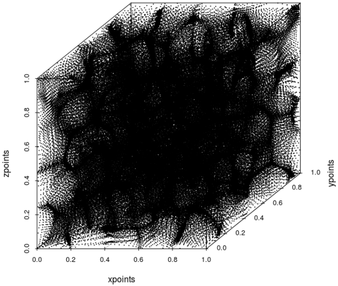
gevolution and void science

What can we learn by studying voids in gevolution?

- going past the linear regime (second order ϕ and ψ)
- Pisani, Massara, Spergel et.al.: large scale N-body simulations of cosmic voids can reduce degeneracies between cosmological models

Modelling voids:





Summary: workshop plans

QMUL contact

- `gevolution` **developers:** gevolution@unige.ch
- **Julian Adamek:** julian.adamek@qmul.ac.uk

Workshop plans

- installation of the code and necessary libraries
- linking CLASS
- running simple simulations
- visualization
- short introduction to C++ syntax
- using LATfield2 classes (for `gevolution` modification)
- implementation of equations in the code