Galacticus Overview

- Open source (compiles with GNU compilers)
- Modular design
  - Each function can have multiple implementations, selected by input parameter.
  - “Node” can have arbitrary number of components (e.g. DM halo, disk, spheroid), all with multiple implementations
- Combination of smooth (ODE) evolution and instantaneous events (e.g. mergers)

Galacticus Overview

- New implementation of function easily added:
  - Write a class containing the implementation
  - Add directives indicating that this function is for, e.g., disk star formation timescale calculations
  - Recompile – build system automatically finds this new module and works out how to compile it into the code
- Modules are self-contained and independent
- Self-initializing and recursive

Galacticus Overview

- Component could be, e.g. disk (exponential)
- Stores various types of data:
  - Scalars
  - Vectors
  - Orbits
  - Elemental abundances
  - Time Series
  - Stellar populations
- Allows for multiple components of each type

Galacticus Overview

- Defining a component:
  - Set of ODEs giving rates of change of properties (can access properties of other components/nodes as needed)
  - Responses to events (merging, becoming satellite etc.)
  - Handling of various tasks (e.g. computing quantities need to determine size)
  - Specify properties to be output
  - Definition is via XML – boilerplate functions are auto-generated

Node evolution

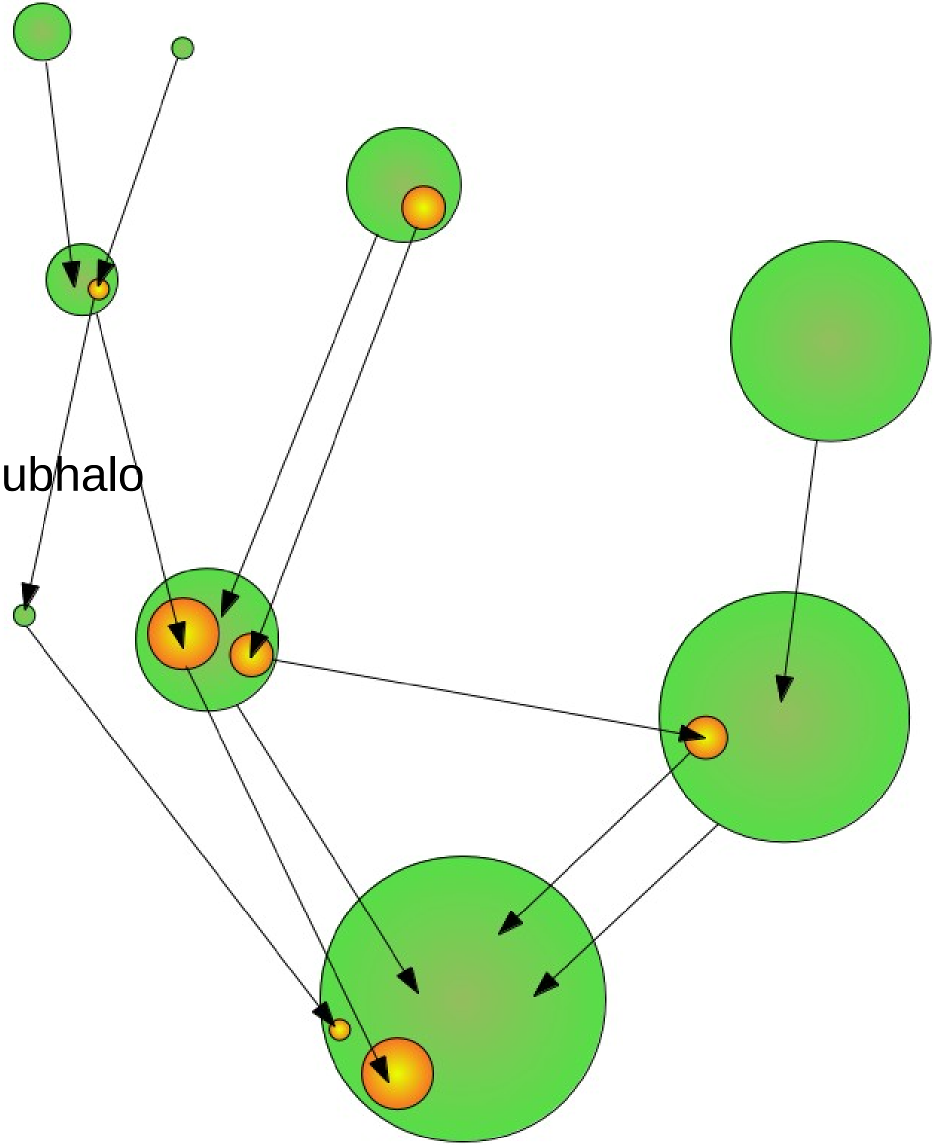
- Repeatedly walk tree – find nodes that to evolve:
  - Cannot evolve if have children
  - Can't evolve beyond their satellites
  - Limit on timestep
  - Arbitrary other factors can be included

Node evolution

- All component properties fed into ODE solver
- Evaluate derivatives – evolve forward in time
- No need for fixed timesteps or analytic solutions
  - Makes implementing, for example, Kennicutt-Schmidt law trivial (just add new star formation timescale function)
- Evolution can be interrupted as needed (e.g. when galaxy merges)

Merger Trees

- Assumptions
  - ⭐ Host halo not in merger tree file?
    ⟹ Host descendent is same as that of hosted subhalo
  - ⭐ Hosting loop? (2 halos are each others hosts)
    ⟹ Reset more massive to be unhosted
  - ⭐ Subhalo-Subhalo mergers
    ⟹ Allowed
  - ⭐ Subhalo becomes non-subhalo
    ⟹ Allowed
  - ⭐ Halo mass decreases with time
    ⟹ Allowed (optional)
  - ⭐ Subhalo jumps between branches
    ⟹ Allowed



Advantages

- Modularity makes it highly flexible:
  - Add new star formation rule in 5 minutes
  - Change in cooling model confined to few modules which compute cooling time and rate
- Unified ODE solver makes new features simple:
  - Timestepping handled automatically
  - No need for analytic solutions
  - Implemented noninstantaneous recycling in one afternoon rather than two months!

Physics Included – IGM Accretion

- Simple model:
  - Assume accretion at universal fraction for any smooth accretion
  - Also get hot gas from ram pressure stripping of satellites
  - Accretion from IGM shut off post-reionization in low circular velocity halos

Physics Included – Cooling/Inflow

- Cooling:
  - Calculate cooling radius assuming isothermal (virial temperature) gas, beta-profile density distribution, CIE cooling curves (metal dependent)
  - Rate of growth of cooling radius gives cooling rate
- Infall:
  - Compute rate of growth of freefall radius in the halo – gives and infal rate
- Take the smaller of the two rates

Physics Included – Star Formation

- Lots of star formation rules available for disks
  - Default option is Krumholtz, McKee, Tumlinson
- For spheroids, simple timescale argument based on spheroid dynamical time

Physics Included – Feedback

- Typically use an empirical model
  - Outflow rate proportional to star formation rate
  - Proportionality factor scales with characteristic velocity of system
- Outflowed gas returns to the hot halo on a timescale of order the dynamical time of the halo

Physics Included – Metal Enrichment

- Typically assume instantaneous recycling
  - Tracks only total metals
  - Assume a fixed yield and recycled fraction
- Can also do non-instantaneous recycling
  - Slower
  - Allows tracking of individual elements

Physics Included – Merging

- Merging timescales taken from N-body dynamics
  - Can add on some delay time for sub-resolution evolution
- Mergers classified as minor/major based on simple rules
  - Major mergers lead to destruction of disks, formation of spheroid
- Spheroids can also be formed via disk instabilities

Physics Included – Sizes

- Disks
  - Angular momentum content of disk from cooling model, ultimately from halo spin
  - Solve for scale length of disk assuming flat rotation curve
  - Include adiabatic contraction
- Spheroids
  - Energy conservation argument based on internal and orbital energies of merging galaxies

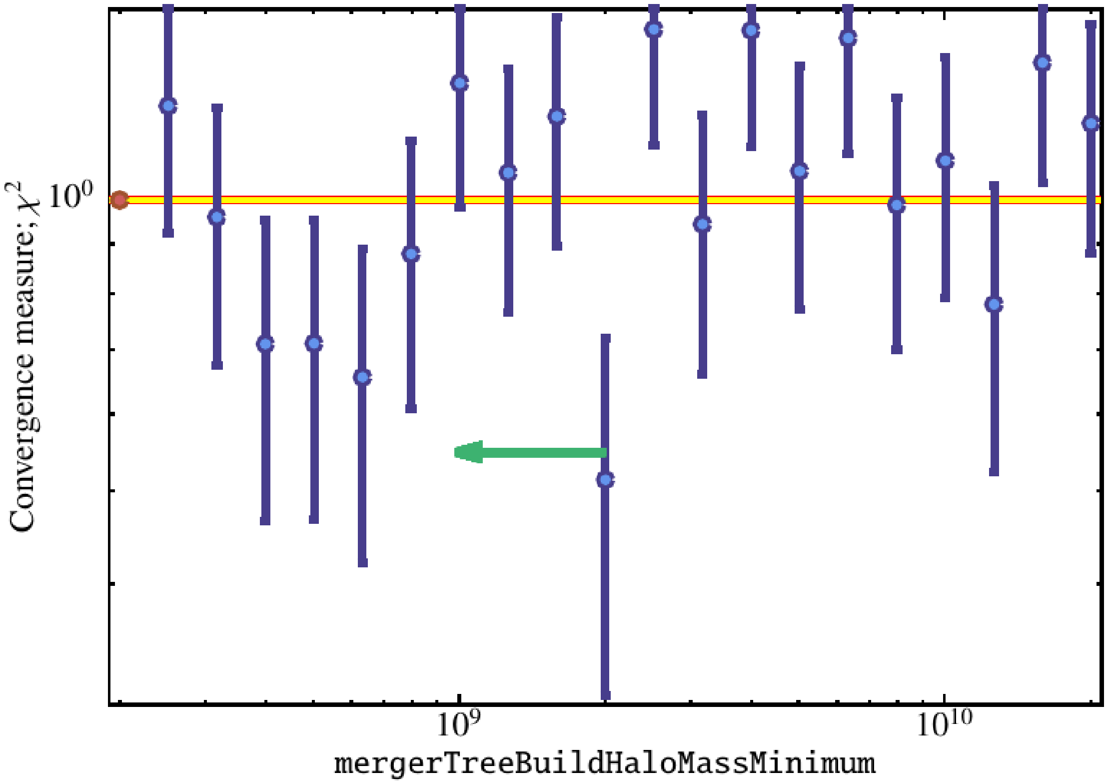
Physics Included – Black Holes

- Grow from spheroid and hot halo gas
  - Bondi-Hoyle accretion with some fudge factor
  - Spin also tracked (Benson & Babul approach)
- Merging
  - Normally assume instant merging when galaxies merge
  - Can also track migration through disk, ejection, etc.
- Feedback
  - Jet power computed from accretion rate and spin
  - Used to offset cooling rate in hot halo and drive gas out of halo

Physics Included – Luminosities

- Typically assume fixed IMF
- Luminosities in any filters found by convolving star formation history with stellar population library
- Effects of dust applied in post-processing
  - Various models available

Convergence – Minimum Halo Mass

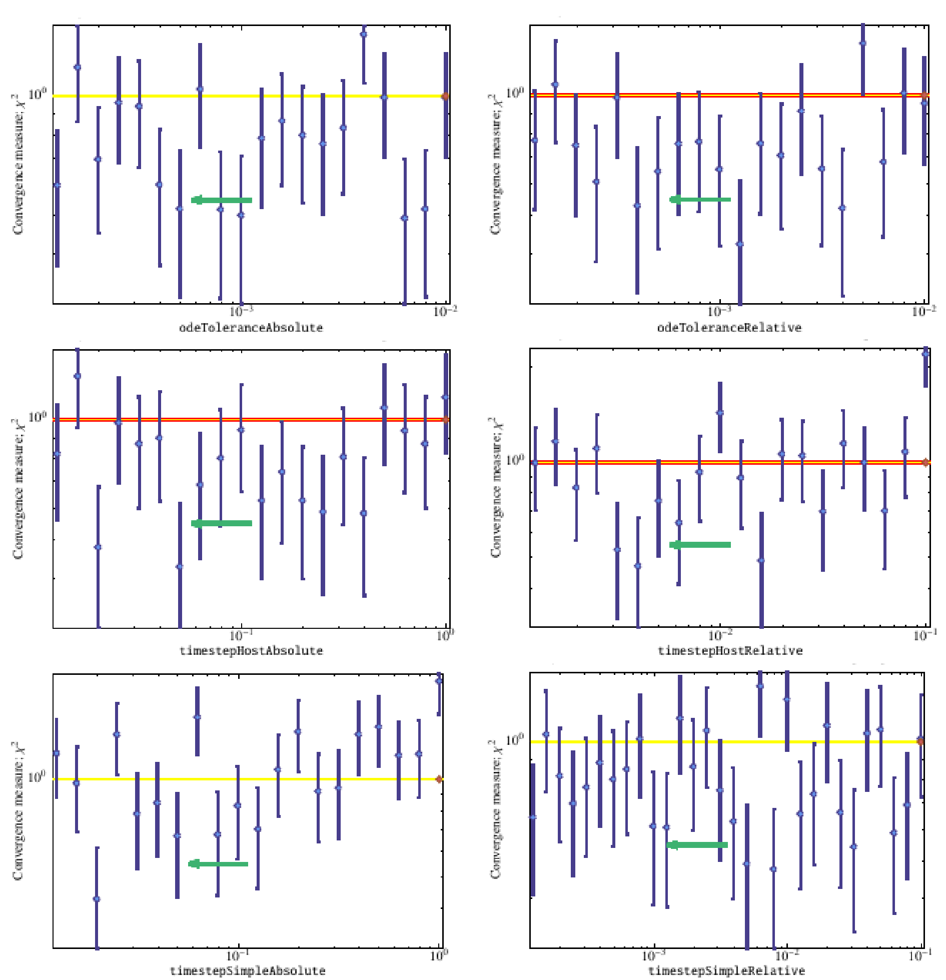


Limiting physics:

Pre-reionization: cooling function

Post-reionization: IGM temperature

Convergence – Timestepping



Parameter Sensitivities

- Complex problem
  - SNe feedback is often the dominant effect
  - But highly problem-specific

Parameter Sensitivities