Synthetic Catalogs with HACC and Galacticus: The ABC Collaboration

Rahul Biswas, Nick Frontiere, Ben Gutierrez, Salman Habib, Katrin Heitmann,
Eve Kovacs, Tanu Malik, Adrian Pope, Esteban Rangel
Argonne National Laboratory

Andrew Benson
Carnegie Observatories

Joanne Cohn
UC Berkeley

Martin White
UC Berkeley and Lawrence Berkeley National Laboratory

more work planned with Los Alamos, UWashington and IPAC collaborators --

LSST Synthetic Catalog Requirements

From Andrew Connolly (two pages of requirements/object):

glat	FLOAT		Galactic latitude (deg)
pa_b	FLOAT		Position angle of bulge (deg)
pa_d	FLOAT		Position angle of disk (deg)
inclination_b	FLOAT		Inclination of bulge to line of sight (deg)
inclination_d	FLOAT		Inclination of disk to line of sight (deg)
sb_eb	FLOAT		Central surface brightness of bulge (mag/arcsec^2)
r_eb	FLOAT		Half light radius of bulge (arcsec)
a_b	FLOAT		Semi-major axis of bulge (arcsec)
b_b	FLOAT		Semi-major axis of bulge (arcsec)
bulge_index	FLOAT		Sersic index of bulge (4)
bra	FLOAT		RA of bulge center (deg)
bdec	FLOAT		Dec of bulge center (deg)
sb_ed	FLOAT		Central surface brightness of disk (mag/arcsec^2)
r_ed	FLOAT		Half light radius of disk (arcsec)

Hence: Galacticus

Galacticus

https://sites.google.com/site/galacticusmodel/about

About Galacticus | Downloads | Documentation | Community | Develop | Blog | Contact Information | Help | Sitemap

About Galacticus

Galacticus is a semi-analytic model of galaxy formation written and maintained by Andrew Benson at The Carnegie Observatories.

Get Galacticus

Galacticus solves the equations that govern the formation and evolution of galaxies within a hierarchically assembling network of dark matter halos as occurs in cold dark matter dominated universes. It is designed for modularity and flexibility to allow for easy extension and adaptation.

You can get Galacticus her

Galacticus is freely availab

N-body merger trees can be input into Galacticus

(tested on MS with 800M halos)

Thursday, January 10, 2013 (Galacticus)

Enhanced N-body Merger Tree Handling

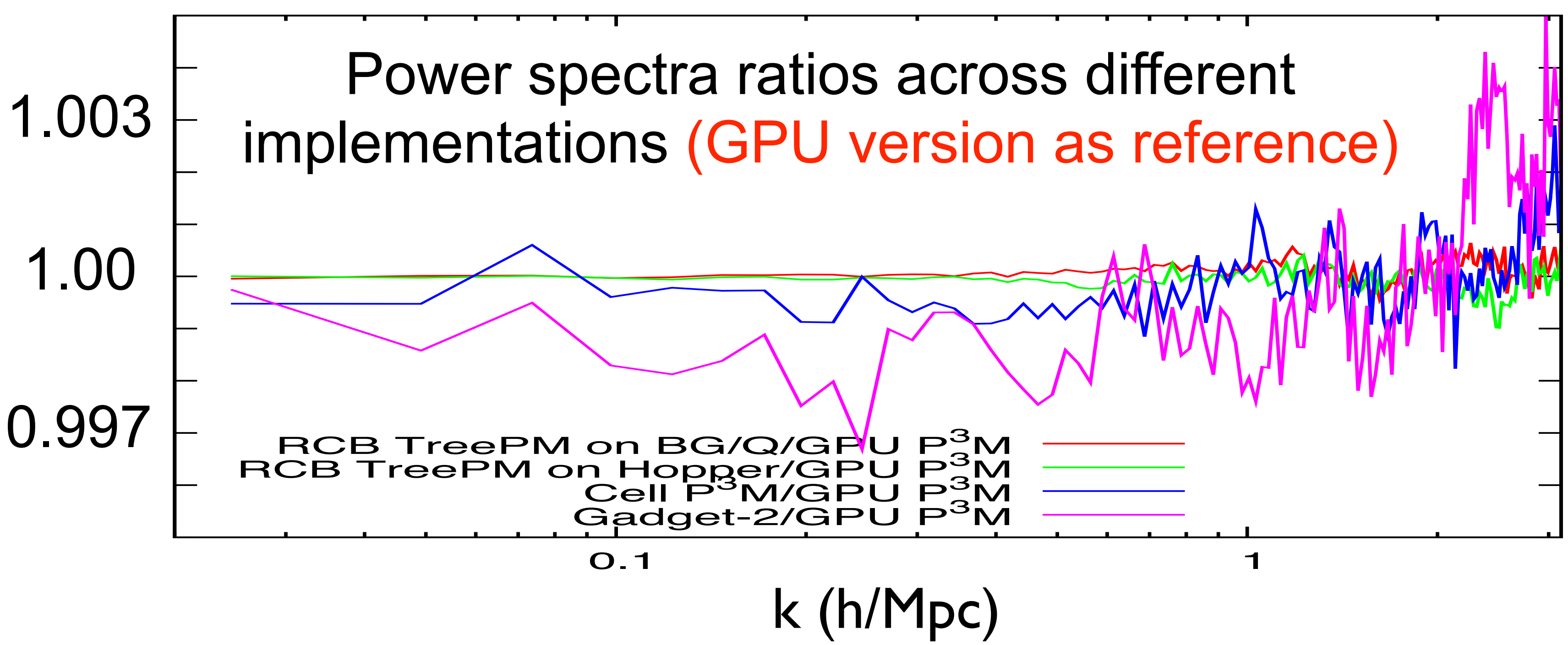
In addition to generating its own merger trees, Galacticus can read trees from file and process them to populate each tree with galaxies. Typically, these merger trees have been extracted from an N-body simulation.

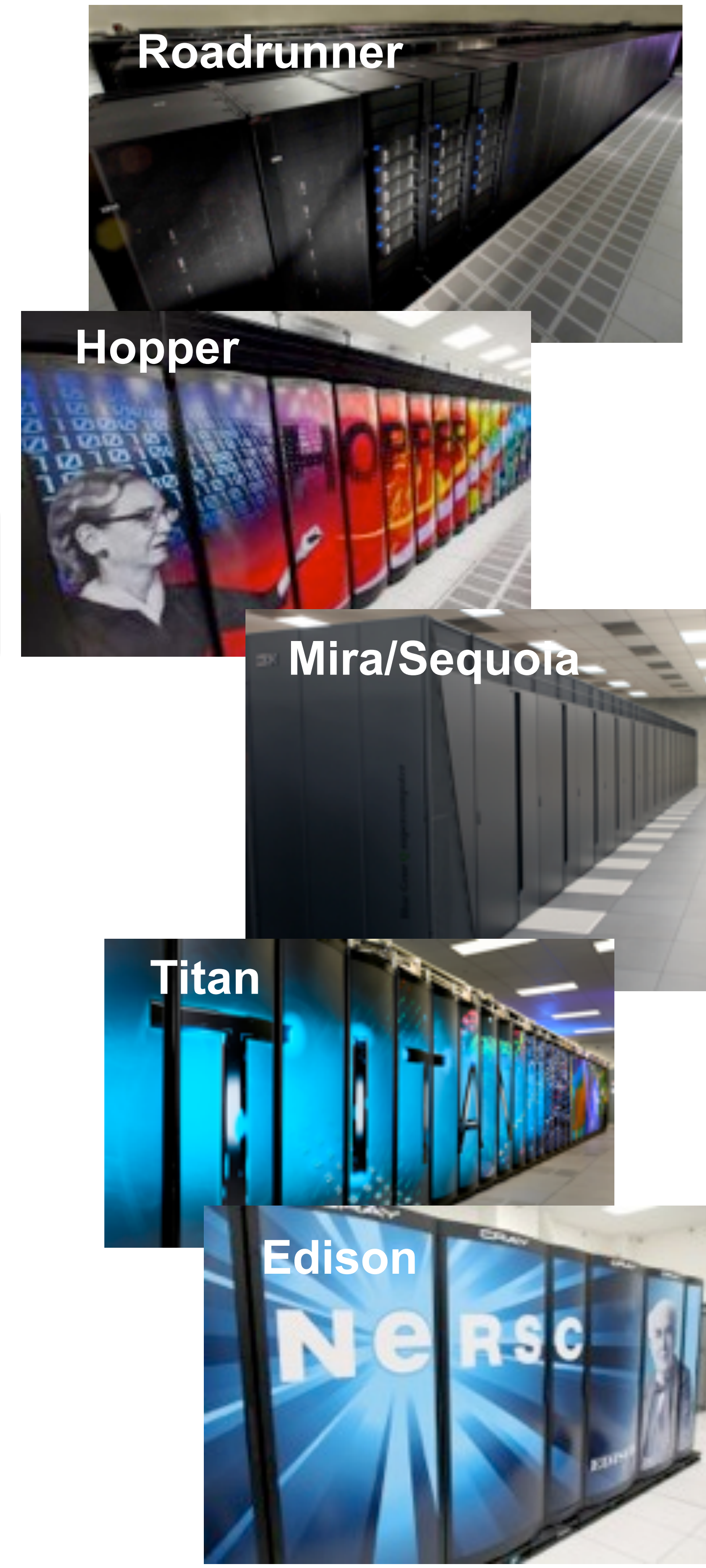
The original implementation of processing such trees in Galacticus had some limitations - which were necessary to make the trees fit with certain assumptions made in the way in which Galacticus processes trees as it creates and evolves galaxies. This processing modified the structure of trees and so could affect the resulting physical properties of galaxies.

We've just finished some improvements to v0.9.2 of Galacticus which remove these assumptions and allow N-body trees to be handled without the need to modify them in any way[1].

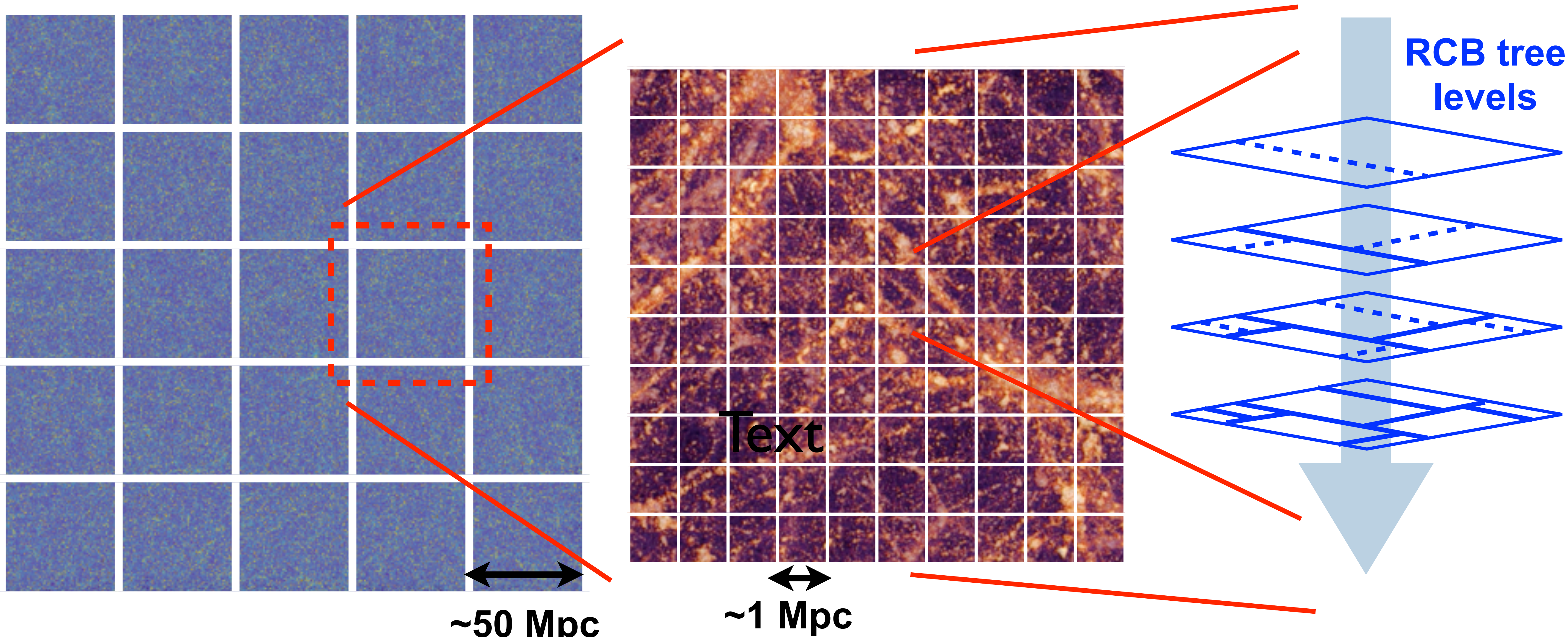
HACC: Extreme-Scale N-body Code

- **Architecture-independent performance/scalability:** ‘Universal’ top layer + ‘plug in’ node-level components; minimize data structure complexity and data motion
- **Programming model:** ‘C++/MPI + X’ where X = OpenMP, Cell SDK, OpenCL, CUDA, --
- **Algorithm Co-Design:** Multiple algorithm options, stresses accuracy, low memory overhead, no external libraries in simulation path
- **Analysis tools:** Major analysis framework, tools deployed in stand-alone and in situ modes





'HACC In Pictures'



HACC Top Layer:

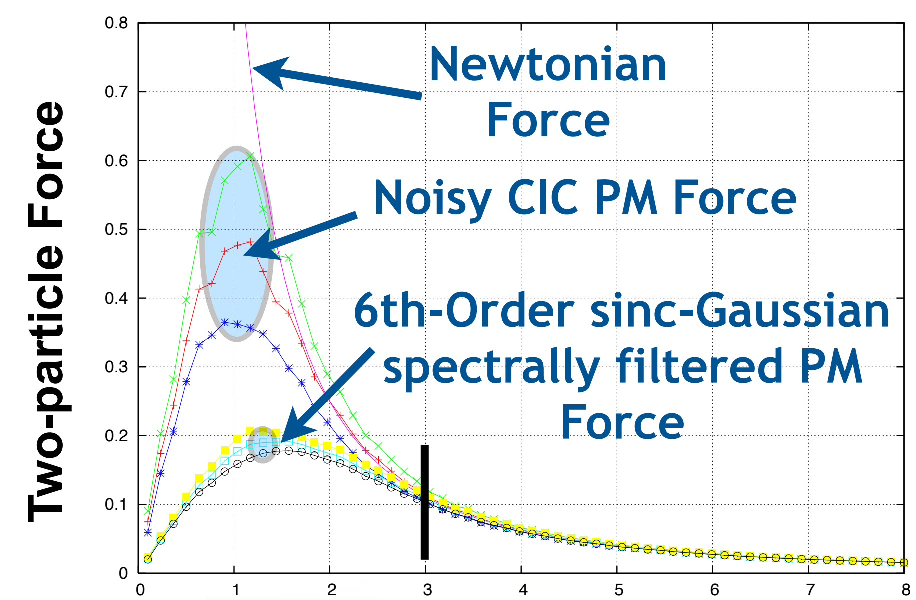
3-D domain decomposition with particle replication at boundaries ('overloading') for Spectral PM algorithm (long-range force)

Host-side

HACC 'Nodal' Layer:

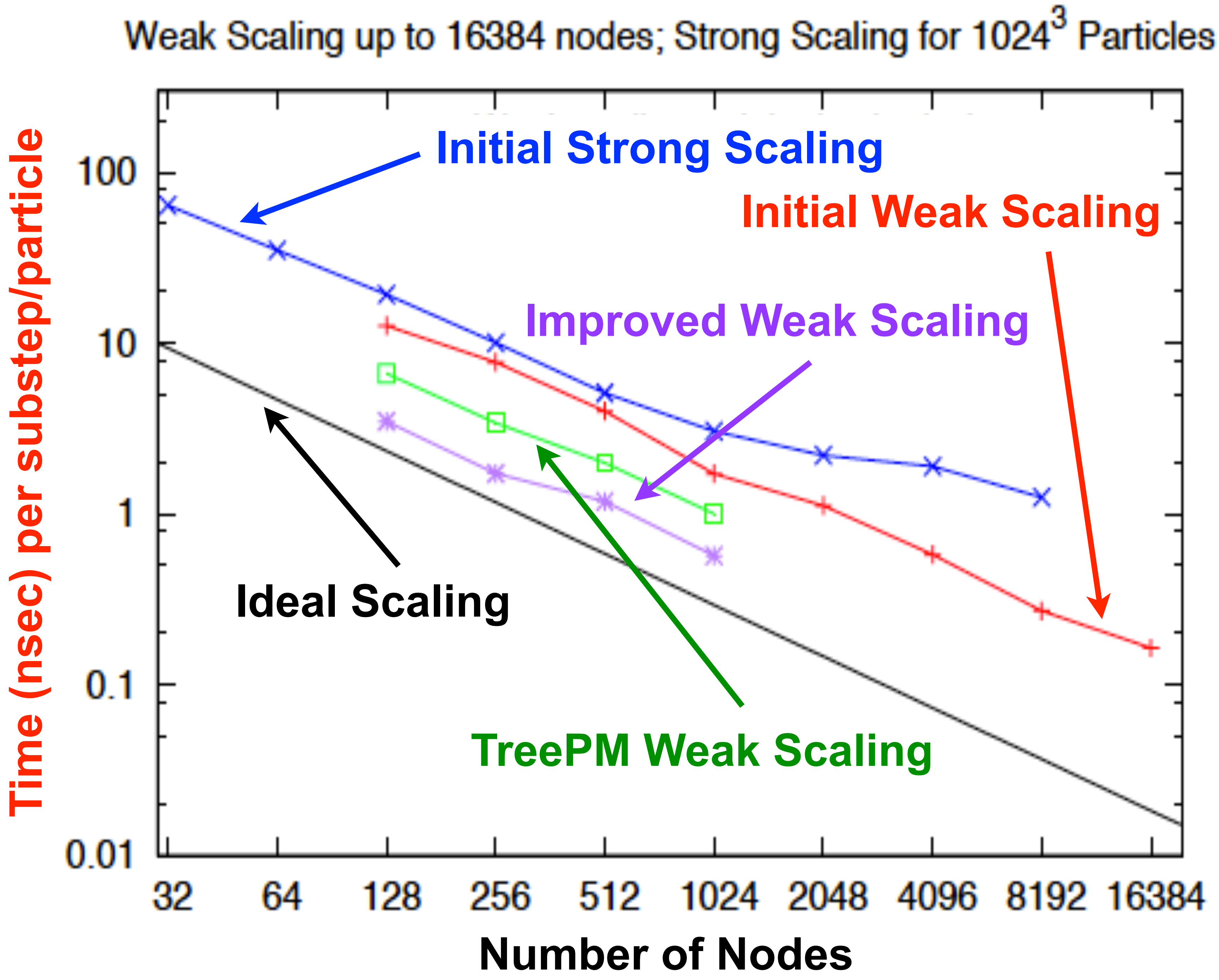
Short-range solvers employing combination of flexible chaining mesh and RCB tree-based force evaluations

GPU: two options, P3M vs. TreePM



HACC on Titan: GPU Implementation Performance

- P3M kernel runs at 1.6TFlops/node at 40.3% of peak (73% of algorithmic peak)
- TreePM kernel was run on 77% of Titan at 20.54 PFlops at almost identical performance on the card
- Because of less overhead, P3M code is (currently) faster by factor of two in time to solution
- Large new INCITE allocation on Titan and Mira for 2014



99.2% Parallel Efficiency

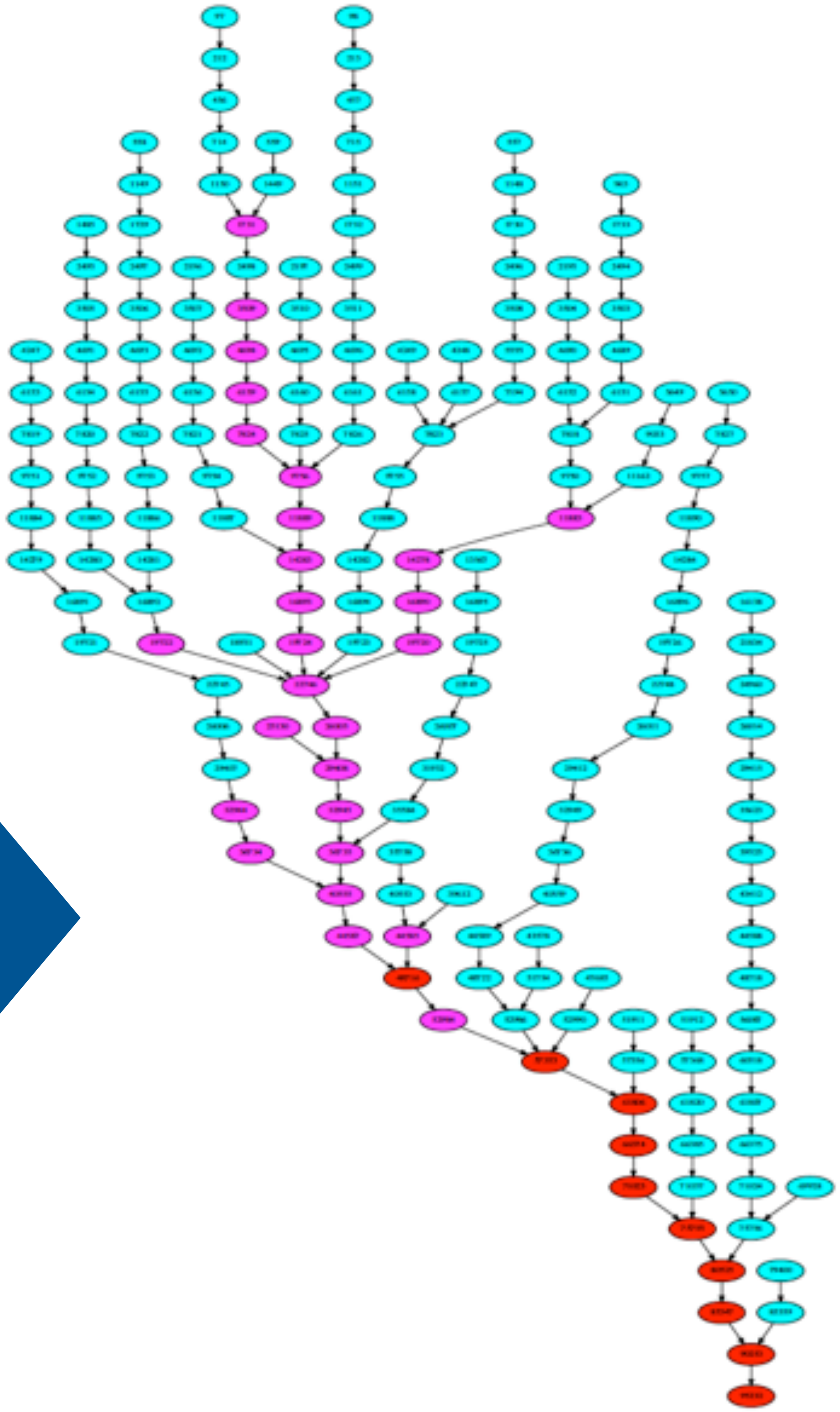
HACC Merger Trees

Particle id sort plus O(N) halo intersection cardinality determination (sparse matrix representation to reduce memory overhead); windowed history of ‘missing’ (zombie) halos maintained to prevent misidentifications of halos near mass threshold

Parallel tree construction on 3-D domain data structure (need nearest neighbor searches only)

Number of tests underway with small ‘toy’ simulations (256 Mpc/h, billion particles, MS-like mass resolution)

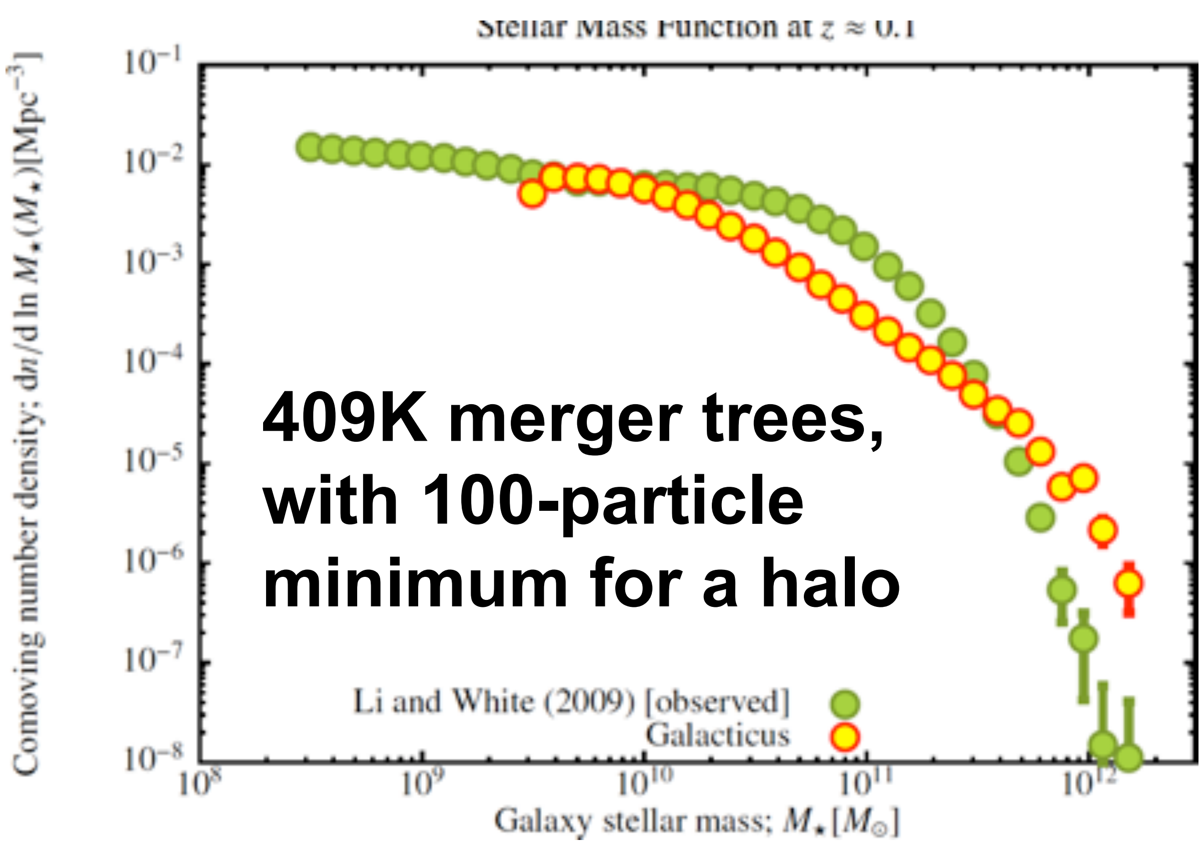
Will include sub-halos next (Galacticus knows how to deal with this)

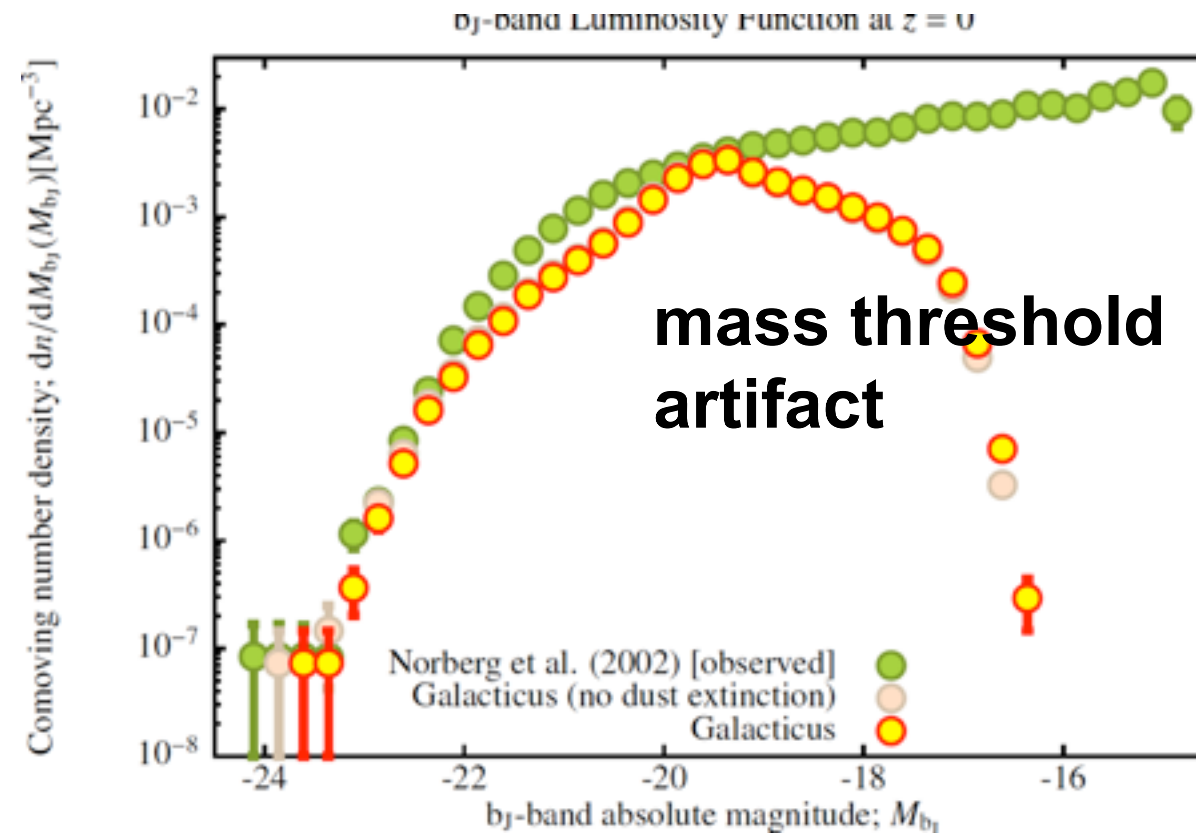


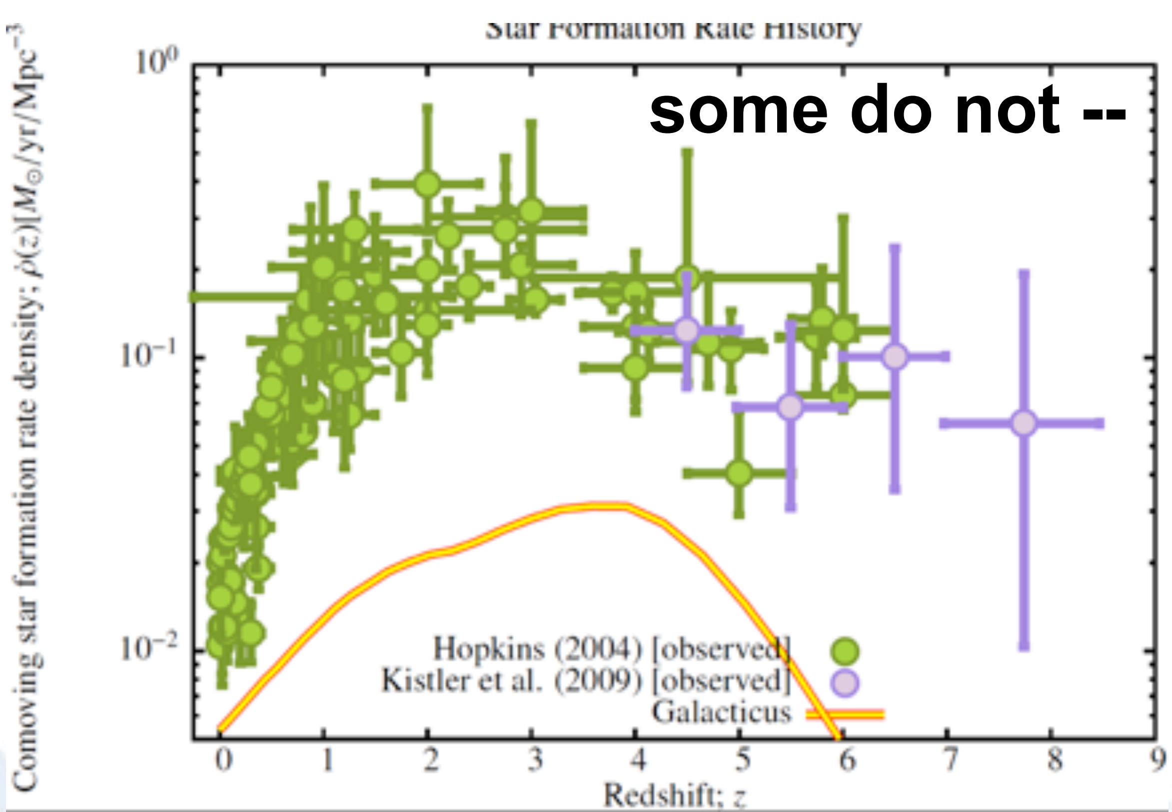
Galacticus: ‘Minimum” Approach

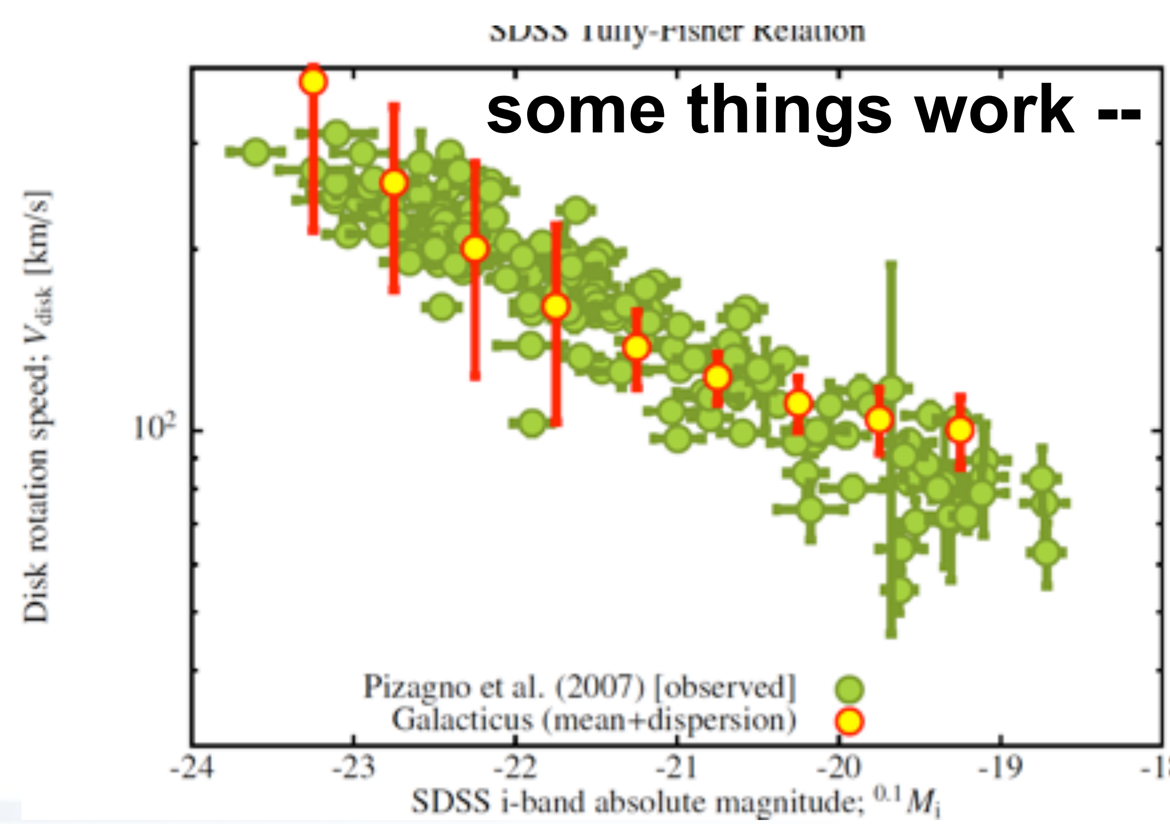
- In principle, Galacticus (like all SAMs) has a large number of free parameters (~50-100), too many to deal with as a start
- Approach: take a ‘reasonable’ number of parameters, run Galacticus on a set of merger trees from one simulation, sampling this parameter space, construct an emulator for a number of one-point statistics, compare to data and see if a more or less ‘validated’ model emerges (can also do two-point statistics)
- Reasonable number of parameters is taken to be 17 (Andrew and Martin), with this number, Galacticus runs fairly quickly
- Evaluate N-body merger trees using 100 snapshots from the simulation (need to keep only particle ids for merger tree construction), can do finer/ coarser
- Run Galacticus at a 175 different sampling points for emulator construction (quite practical, problem is trivially task parallel)
- Validation data being set up by Joanne (also later with IPAC collaborators)
- Current test results appear to be ‘in the SAM ballpark’

Galacticus Tests: First Results



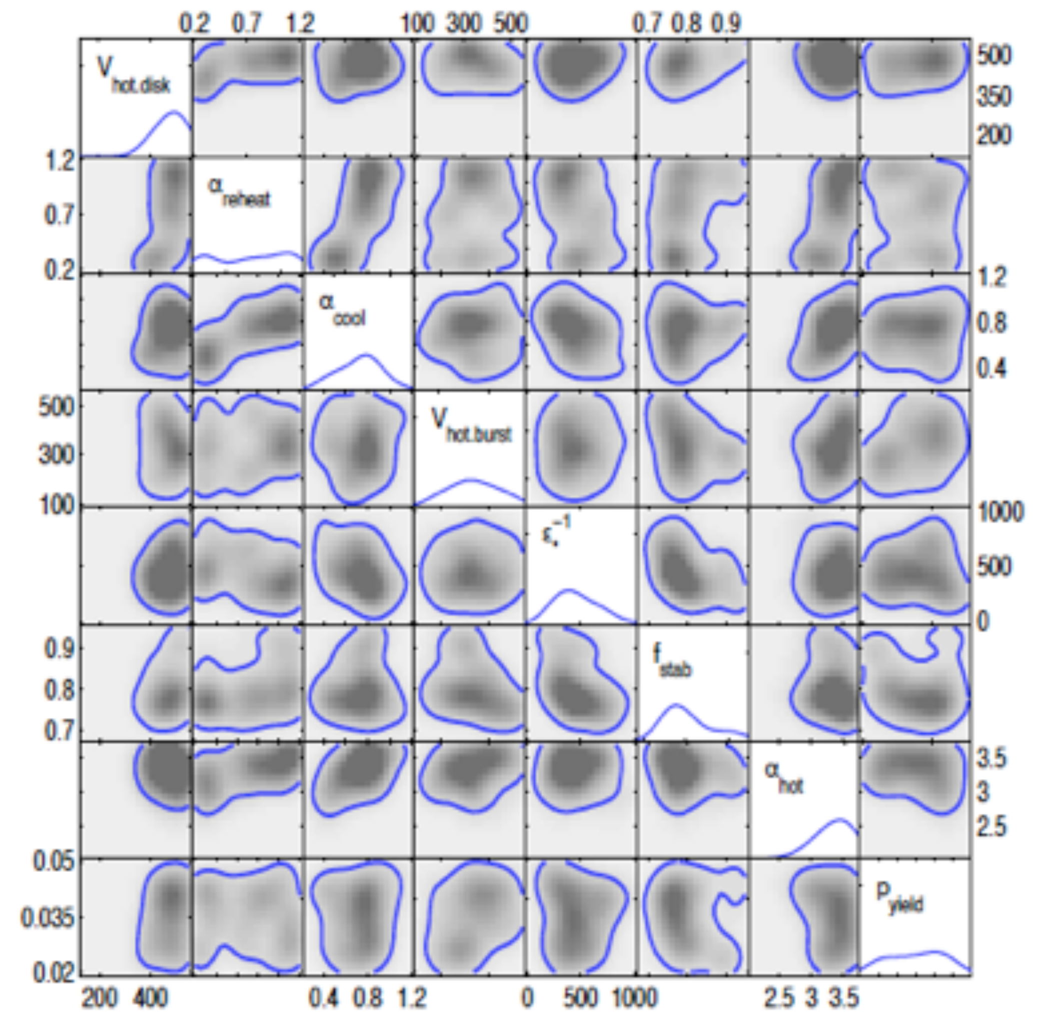






What's Next?

- Get a better feel for Galacticus parameters
- When satisfied that code is running as expected, build emulator
- Understand the galaxy formation 'response surface' from Galacticus (analogous work for Galform already perfromed by LANL collaborators)
- Then see if it all works --
- Build catalogs
- And how it depends on cosmology --
- How will we need to improve Galacticus (and we will)?
- Work with wider community of galaxy formation simulators



Lawrence and Higdon, Bayesian emulator for Galform (J. Bayes Anal.)