Contents

1 Compiling

If you use the GNU C compiler version 4.0 or above on a 64-bit machine, compiling should be as simple as typing `make` at the command prompt.

The tree code does not support compiling on 32-bit machines and has not been tested with other compilers. Additionally, the tree code does not support non-Unix environments. (Mac OS X is fine; Windows is not). OpenMP is now required to compile.

The tree code can use up to 32 cores, but is not completely parallelized. In its current version, it can be reasonably used with up to 4096^3-particle simulations.

2 Running

2.1 Quick Start for Rockstar Users

If you already have halo catalogs generated using the Rockstar Halo Finder (https://bitbucket.org/gfcstanford/rockstar), running is especially easy.

The Rockstar output files (`out_*.list`) are already in the correct input format. To generate all the output directories and the config file, run the following command:

```
prompt> perl /path/to/rockstar/scripts/gen_merger_cfg.pl <rockstar.cfg>
```

where `<rockstar.cfg>` refers to your rockstar configuration file. Then, follow the instructions the script gives you to run the merger tree code.

If you run into problems with this, make sure that you are running the script from the same working directory as you ran Rockstar. If you still have problems, see below for manual instructions.

2.2 Guide for Other Halo Finders

You will need to generate several files in order to run the merger tree code. First, you will need to convert your halo catalogs into the correct input format. The merger tree code currently accepts input only in ASCII files with one halo per line. The columns should be in the following order:

```
#ID DescID Mass Vmax Vrms Radius Rs Np X Y Z VX VY VZ JX JY JZ Spin
```

The columns should contain:

- `ID`: the halo ID, which must be unique across a single snapshot and must be at least 0.

- `DescID`: the halo's descendant id at the next timestep. If no descendant exists, this should be `-1`. At least 100 halos should have descendants (identified by particle-based methods) in order for the consistency checks to work properly.

- `Mass`: the halo mass. This does **not** have to be $M_{\rm vir}$, but it must correspond to the mass with the radius in the `Radius` column. The **units** for this must be $M_\odot/h$.

- `Vmax`: the maximum circular velocity, in units of km/s (physical, not comoving).

- `Vrms`: the velocity dispersion in units of km/s (physical, not comoving); may be set to 0 if not available.

- `Radius`: the radius at which the mass is calculated in column 3. The **units** must be in kpc / h. (**Note** that this is different from the position units!)

- `Rs`: the scale radius of the halo, in units of kpc / h; may be set to 0 if not available.

- `Np`: number of particles in the halo.

- `X/Y/Z`: 3D position of the halo, in units of Mpc / h. (**Note** that this is different from the radius units!)

- `VX/VY/VZ`: 3D velocity of the halo, in units of km/s (physical, not comoving).

- `JX/JY/JZ`: 3D angular momentum of the halo, in units of $(M_\odot/h) * (Mpc/h) * km/s$ (physical, not comoving); may be set to 0 if not available.

- `Spin`: Dimensionless spin parameter; may be set to 0 if not available.

Note that non-numbers (like `NaN` and `Inf`) are **not acceptable** as inputs and may cause either the halo or the entire file to be rejected.

Here's a sample input from a scale factor of 0.075:

```
#ID DescID Mvir Vmax Vrms Rvir Rs Np X Y Z VX VY VZ JX JY JZ Spin
165 305 6.1824e+10 210.91 213.66 106.167 13.025 49 63.97294 100.14336 36.89451 78.44 82.65 117.79 3.852e+09 9.976e+08 -9.825e+09 0.14646
166 307 9.3673e+09 100.62 108.34 56.599 14.260 34 28.23423 88.26684 27.98982 -42.16 112.89 61.36 -2.751e+08 -8.645e+07 -1.513e+08 0.10760
167 309 1.4988e+10 119.98 127.34 66.199 14.129 22 30.63638 101.30433 25.59135 64.70 106.94 115.71 -5.153e+08 8.236e+08 -9.118e+07 0.16328
168 310 2.4355e+10 135.83 144.66 77.827 24.273 22 98.15601 12.42515 30.58211 58.75 31.41 66.24 -9.074e+08 6.611e+08 -9.837e+08 0.05737
3 -1 3.7469e+09 73.56 139.43 41.702 11.390 36 12.10802 0.52037 157.83209 -57.47 -11.73 -119.23 -1.918e+08 -4.043e+07 5.996e+08 1.72959
```

Once you have created a conversion script, you should save each timestep in a file called "`out_XYZ.list`," where `XYZ` is the timestep number. For example, if you had three timesteps, you would save the first in "`out_0.list`," the second in "`out_1.list`," and the third in "`out_2.list`."

Finally, you must create a file listing the number and the scale factor for each snapshot, one combination per line, in order of increasing scale factor. For example:

```
0 0.100
1 0.120
2 0.140
...
98 0.990
99 1.000
```

You should save this in a file called "`DescScales.txt`" or similar.

2.3   Including Extra Parameters

If your halo finder calculates extra parameters that you wish to propagate through the merger trees, you'll have to add a few options to the config file:

```
EXTRA_PARAMS = <number of extra parameters>
```

This number of parameters must correspond exactly to additional columns in the `out_*.list` files. To keep things straight, you should also specify

```
EXTRA_PARAM_LABELS = "label of each extra column"
EXTRA_PARAM_DESCRIPTIONS = "#Desc of column 1\n#Desc of column 2..."
```

to make sure that users of your merger trees understand what the extra columns mean.

Finally, you may optionally specify

```
EXTRA_PARAM_INTERPOLATIONS = "..."
```

This should be a list of characters, one per extra parameter, to specify whether the extra parameter should be linearly interpolated ("l"), or whether the square or the cube of the parameter should be linearly interpolated ("s" or "c," respectively). Most parameters should be linearly interpolated; however, parameters which are related by non-linear power laws (e.g., halo mass and halo radius) should make use of the alternate methods (linear for the halo mass and linear-cubic for the radius, in this example). No commas or other separators should be used; for example, with five extra parameters, the value of this config option defaults to "lllll."

2.4 Using Binary Input and Intermediate Formats

Consistent Trees can run faster if binary input formats are used (note that output formats will still be ASCII-formatted). To use, generate ASCII inputs as above and run

```
prompt>  /path/to/consistent/trees/convert_format merger_tree.cfg out_0.list ...
```

This will generate binary versions of the ASCII inputs, appending .bin to each filename. The merger_tree.cfg is required in order to interpret the input files, as the input files may have extra columns (see EXTRA_PARAMS above). In order to use the binary files with Consistent Trees, you will then have to specify "INPUT_FORMAT = BINARY" in the config file. The convert_format command automatically recognizes whether the input is in binary or ASCII format, so the same command can be used to convert files back into ASCII formats:

```
prompt>  /path/to/consistent/trees/convert_format merger_tree.cfg out_0.list.bin ...
```

This will generate ASCII files with ".txt" appended to the input filenames. Note that specifying merger_tree.cfg is again required in order to have the correct columns in the output file.

2.5 Configuration Options

It's best to copy one of the example scripts already provided (e.g., bolshoi.cfg or consuelo.cfg) and to modify it as necessary.

Config options you need to modify:

```
Om = 0.27 # Omega Matter
Ol = 0.73 # Omega Lambda
h0 = 0.70 # h0

BOX_WIDTH = 250 # The size of the simulation in Mpc/h
BOX_DIVISIONS = 5 # For large sims, putting all the halos in one tree
      # file gets too unwieldy.  If BOX_DIVISIONS is larger than 1, then
      # the trees will be split into BOX_DIVISIONS^3 files, each
      # containing a cubic subregion of the box.

SCALEFILE = "/path/to/DescScales.txt"
INBASE = "/path/to/halo/catalog/directory" #Where out_*.list are.
```

```
OUTBASE = "/path/for/intermediate/steps/and/logfiles"
TREE_OUTBASE = "/path/for/tree/files"
HLIST_OUTBASE = "/path/for/output/halo/catalogs"

#The last three directories are for storing the outputs of the tree code.
#The "OUTBASE" will contain intermediate calculations, statistics, and
#detailed logfiles about which halos are created or destroyed (and why).
#These files can occupy a significant amount of space, on the order of
#5x the size of the original input files.

MASS_RES_OK=1e11 #Halo mass at which there are at least 1000 particles
                 #in the halo

#These options set the number of timesteps over which the halo has to
# appear in order to be considered "valid."
MIN_TIMESTEPS_TRACKED = 5
MIN_TIMESTEPS_SUB_TRACKED = 10
MIN_TIMESTEPS_SUB_MMP_TRACKED = 10
#Generally, MIN_TIMESTEPS_TRACKED should be set to 1/30 of the number of
# timesteps, and MIN_TIMESTEPS_SUB_* should be set to 1/15 the number of
# timesteps; these limits are a good balance between catching spurious
# halos but also allowing for halos which merge soon after they appear.

#Option for applying fix for Rockstar spins (only for versions prior
#to Rockstar v0.99.9-RC3).  Set to 0-indexed column of T/|U| in EXTRA_PARAMS
#E.g., if EXTRA_PARAM_LABELS is "Spin_bullock M200c T/|U|", then
#FIX_ROCKSTAR_SPINS would be set to 2.
FIX_ROCKSTAR_SPINS = -1

#Option to limit memory usage; set to 1 if calculating merger tree for a large
#(>2048^3 particles) box.  This turns off gzipping intermediate files and also
#limits some parallelization to avoid having too many snapshots in memory.
LIMITED_MEMORY = 0
```

The other configuration options should generally be left alone unless you
are doing development work on the tree code.

2.6 Running the Code

Once you've compiled the code and created the config file, you can run the tree code. If you have a periodic box, run

```
cd /path/to/consistent_trees
perl do_merger_tree.pl myconfig.cfg
```

If you have a non-periodic box, then you should run instead:

```
perl do_merger_tree_np.pl myconfig.cfg
```

This will generate the merger trees. To generate halo catalogs from the merger trees (which include properties like Vmax at accretion, etc.), then once the above command finishes, you should run

```
perl halo_trees_to_catalog.pl myconfig.cfg
```

(Again from the same directory as before).

2.7 Troubleshooting

The MOST COMMON failure for the tree code is the following error:

```
Error: too few halos at scale factor 0.XYZ to calculate consistency metric.
Please remove this and all earlier timesteps from the scale file and rerun.
```

If you get this, it means that there are too few halos at high redshifts to reliably calculate whether halos are consistent between timesteps or not. To fix this problem, edit your DescScales.txt file and remove any timesteps which are at or before the scale factor mentioned, and then rerun the tree code. This should solve the problem in 99% of all cases.

3 Using the Trees

3.1 Reading the Tree Files

Included with the tree code is some example code for reading in the tree files produced. (Not the halo catalogs, however). This code is located here:

```
/path/to/consistent_trees/read_tree
```

The included `Makefile` shows how to compile the code into an executable, and the included `example.c` file shows how to call the reading procedure.

Once the tree is loaded, you can access the halo tree from the `halo_tree` structure, and the list of all halos from the `all_halos` structure. The definitions for these structures are in `read_tree.h`.

3.2 Sussing Format

Merger trees in the Sussing Merger Trees format can be generated by running

```
/path/to/consistent_trees/gen_sussing_forests myconfig.cfg
```

after the default-format trees have been generated (§2.6, above). For convenience, the output halo catalogues are sorted by forest, and there is a file called "`sussing_forests.list`" which contains a list of forests, along with the number of halos in each snapshot for each forest.

Note that the Sussing Merger Trees format uses M_{200c} as the default mass. If this is not the same as the standard mass definition for the trees, but if M_{200c} is available in the `EXTRA_PARAMS` fields, then you should specify this in the config file by setting `SUSSING_MASS_FIELD` to the 0-indexed column for M_{200c} in `EXTRA_PARAMS`. E.g., if `EXTRA_PARAM_LABELS` is "`Spin_bullock M200c T/|U|`", then you would set `SUSSING_MASS_FIELD = 1` in the config file.

3.3 Halo Statistics

In the `OUTBASE` directory, a large number of statistics and logfiles are kept about which halos are added and deleted and why. In particular, the files "`statistics_XYZ.list`" and "`cleanup_statistics_XYZ.list`" contain information about how many halos are added and deleted as a function of halo mass in Phase I and Phase II of the algorithm (respectively). To determine how many halos have actually been deleted from the original catalogs, subtract the unlinked phantom column (UP) from the deleted halos column (D) in the `cleanup_statistics` files.

If too many halos are being deleted, you may want to soften some of the consistency requirements (number of timesteps tracked, etc.) or use a more consistent halo finder. `;-)`

Information about the accuracy of positions and velocities returned by the halo finder is kept in the `metric_statistics` files (for all halos) and the `metric_subs_statistics` files (for subhalos).