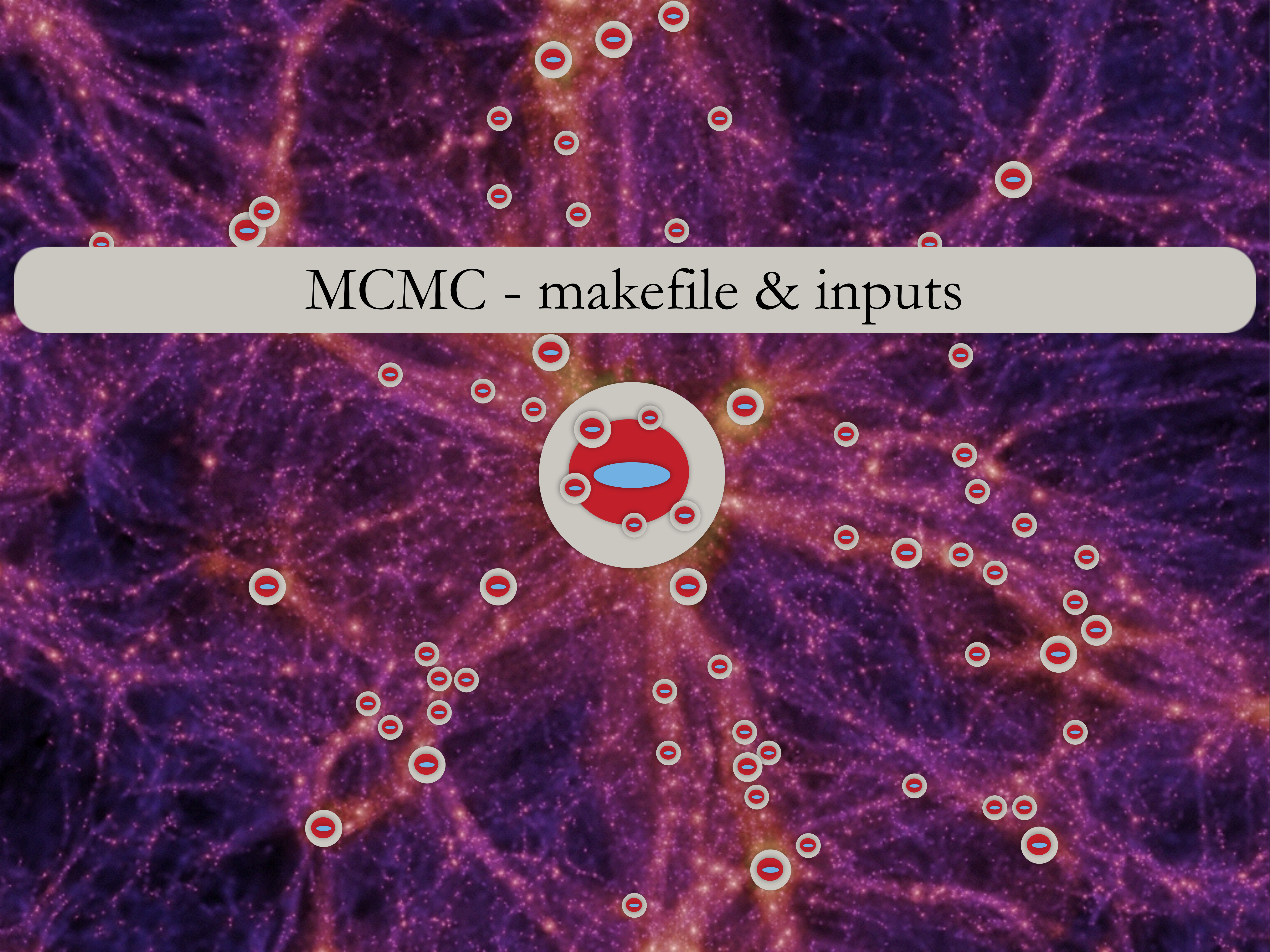
MCMC - makefile & inputs



MCMC - change parameters and constraints

Max-Planck-Institut für Astrophysik

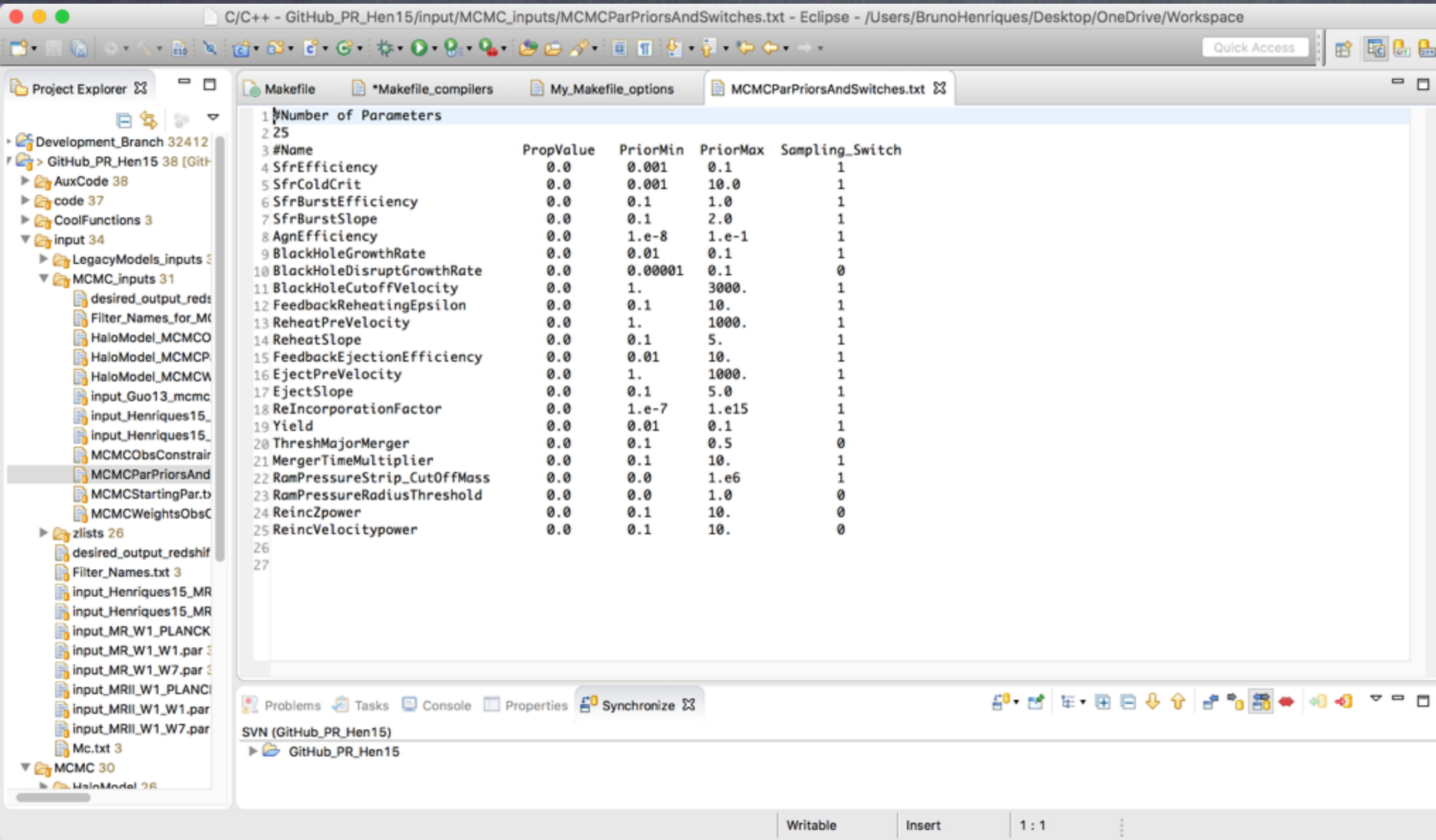
2. Quick Guide to include more Parameters or Observational Constraints:

If you are just starting to use the MCMC you'll likely want to try to constrain different parameters in the model with different observational constrains. It is recommended that you start with what is currently available and from which you can easily choose by using the parameters switches at ./input/MCMC_inputs/MCMCParameterPriorsAndSwitches.txt, the observational constraints switches at ./input/MCMC_inputs/MCMCObsConstraints.txt and the observational constraints weights at ./input/MCMC_inputs/MCMCWeightsObsConstraints.txt. All these are described in the next section and allow you to select any combination of parameters to be sampled, and any combination of pre-defined constraints, with different weights, to be used at any redshifts.

Change parameters:
./input/MCMC_inputs/MCMCParameterPriorsAndSwitches.txt

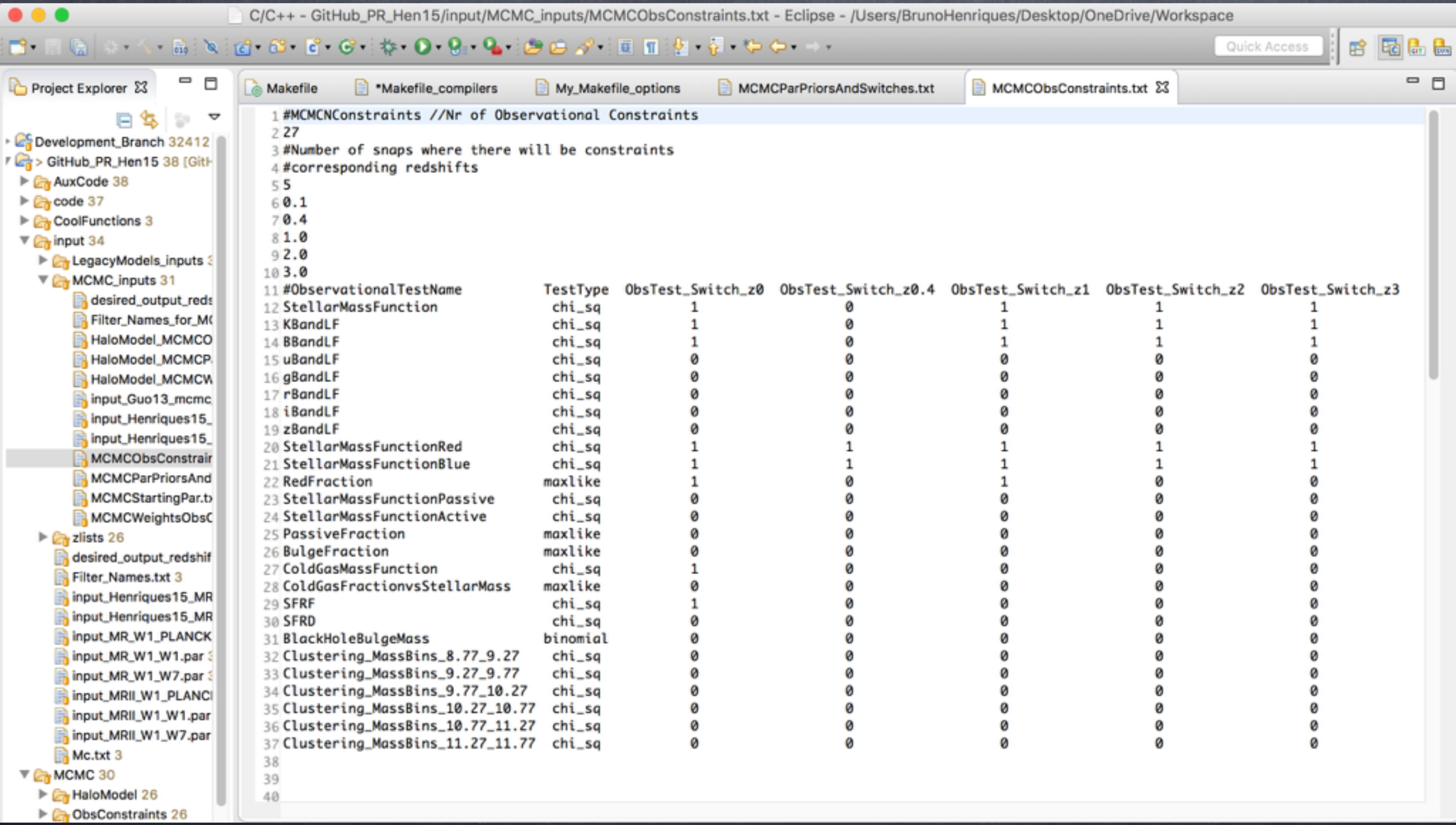
Change observational constraints:
./input/MCMC_inputs/MCMCObsConstraints.txt
./input/MCMC_inputs/MCMCWeightsObsConstraints.txt

MCMC - change parameters


C/C++ - GitHub_PR_Hen15/input/MCMC_inputs/MCMCParPriorsAndSwitches.txt - Eclipse - /Users/BrunoHenriques/Desktop/OneDrive/Workspace

```
#Number of Parameters
25
#Name                        PropValue   PriorMin   PriorMax   Sampling_Switch
SfrEfficiency                  0.0        0.001      0.1           1
SfrColdCrit                    0.0        0.001      10.0          1
SfrBurstEfficiency             0.0        0.1        1.0           1
SfrBurstSlope                  0.0        0.1        2.0           1
AgnEfficiency                  0.0        1.e-8      1.e-1         1
BlackHoleGrowthRate            0.0        0.01       0.1           1
BlackHoleDisruptGrowthRate     0.0        0.00001    0.1           0
BlackHoleCutoffVelocity        0.0        1.         3000.         1
FeedbackReheatingEpsilon       0.0        0.1        10.           1
ReheatPreVelocity              0.0        1.         1000.         1
ReheatSlope                    0.0        0.1        5.            1
FeedbackEjectionEfficiency     0.0        0.01       10.           1
EjectPreVelocity               0.0        1.         1000.         1
EjectSlope                     0.0        0.1        5.0           1
ReIncorporationFactor          0.0        1.e-7      1.e15         1
Yield                          0.0        0.01       0.1           1
ThreshMajorMerger              0.0        0.1        0.5           0
MergerTimeMultiplier           0.0        0.1        10.           1
RamPressureStrip_CutOffMass    0.0        0.0        1.e6          1
RamPressureRadiusThreshold     0.0        0.0        1.0           0
ReincZpower                    0.0        0.1        10.           0
ReincVelocitypower             0.0        0.1        10.           0
```

MCMC - change constraints



MCMC - makefile options

Max-Planck-Institut für Astrophysik

3. Makefile & Input Options

3.1 Makefile options:

Makefile options for the MCMC are simple. Once the Makefile has been changed to compile using My_Makefile_options_MCMC that has **OPT += -DMCMC** on. That, in combination with the right input file, switches the code into MCMC mode. In addition, there is the option to use a sample both with Millennium and MillenniumII merger trees: **OPT += -DMR_PLUS_MRII** (which requires the appropriate input file).

When OPT += -DMCMC is selected an additional OPT += -DHALOMODEL option is available. This will compute satellite profiles using a halo model, allowing the correlation function of galaxies to be accurately calculated from a very small subset of dark matter haloes. This makes it possible to use clustering measurements as observational constraints in the MCMC sampling. Further details are given in section Halo Model below.

MCMC - input file

Max-Planck-Institut für Astrophysik

3.2 MCMC input files:

A large number of additional inputs are needed in MCMC mode.

3.2.1 Parameters to be sampled:

- MCMCParameterPriorsAndSwitches: This file list the parameters sampled in the MCMC. It contains priors and a switch for each parameter.
  A parameter is only used in the sampling if the switch is turned to 1. For parameters with switch=1, the values are not read from input_mcmc_***.par.
- MCMCStartingParFile: File containing starting values for all parameters to be sampled. Parameters are only read from this file **if the output from previous runs is not available at ./output/senna_g*_**.txt** . The number of values in this file must correspond to the numbers of parameters with switch=1.
  If there is a previous output available at ./output/senna_g*_**.txt the starting values are read from these files and set to the last available step. Therefore, when re-starting in mcmc mode, the code continues from where it stopped (MCMC_Initial_Par_DisplacementMCMC_LogStep_Size=0 when re-starting).

3.2.2 Observational constraints:

- MCMCObsConstraints: file containing the names of observational constraints to be used.
  The first number in the file is the maximum number of observational constraints available (it must be the same as MCMCNConstraints in mcmc_vars.h). Then, there is the number of redshifts for which constraints are available (**must be smaller than NOUT**) and the values for those redshifts (**these must correspond to the outputs of the galaxy formation model listed in ./input/MCMC_inputs/desired_output_redshifts_mcmc.txt**). For each observational constraint a type of test is listed, in addition to a switch for each redshift available.
- MCMCWeightsObsConstraints: file specifying a weight for each observational constraint at each redshift.
- ObsConstraintsDir: folder containing the data for the observational constraints to be used.
  There must a file in this directory corresponding to each constraint and redshift listed in MCMCObsConstraints, e.g. StellarMassFunction_z0.10.txt. The files contain binned values in standard format: **x_bin_low, x_bin_high, y_value, err_y_value**, with the first line in each file containing the number of bins.

MCMC - input file

Max-Planck-Institut für Astrophysik

3.2.3 General inputs:

- ChainLength: number of steps for each chain.
- Time_Dependent_PhysPar: switch to allow parameters values to vary between desired output redshifts.
- MCMCMode: 0 for normal MCMC, 1 to only accept parameters sets that give higher likelihood and find the maximum quicker (likely to get stuck in a local maximum if a small number of chains is used).
- MCMC_LogStep_Size : size of the log_normal step. Regulate in order to get a final acceptance rate between 10-40%.
- MCMC_Initial_Par_DisplacementMCMC_LogStep_Size: initial displacement to be applied to all parameters in log_normal space. Ideally large to start each chain in very different places. In practice it cannot be too large as it will cause a lot of the chains to wonder to zero likelihood regions.
- MCMC_Minimum_Obs_Error: force a minimum error on observational constraints (a value of 10% is probably a good idea).

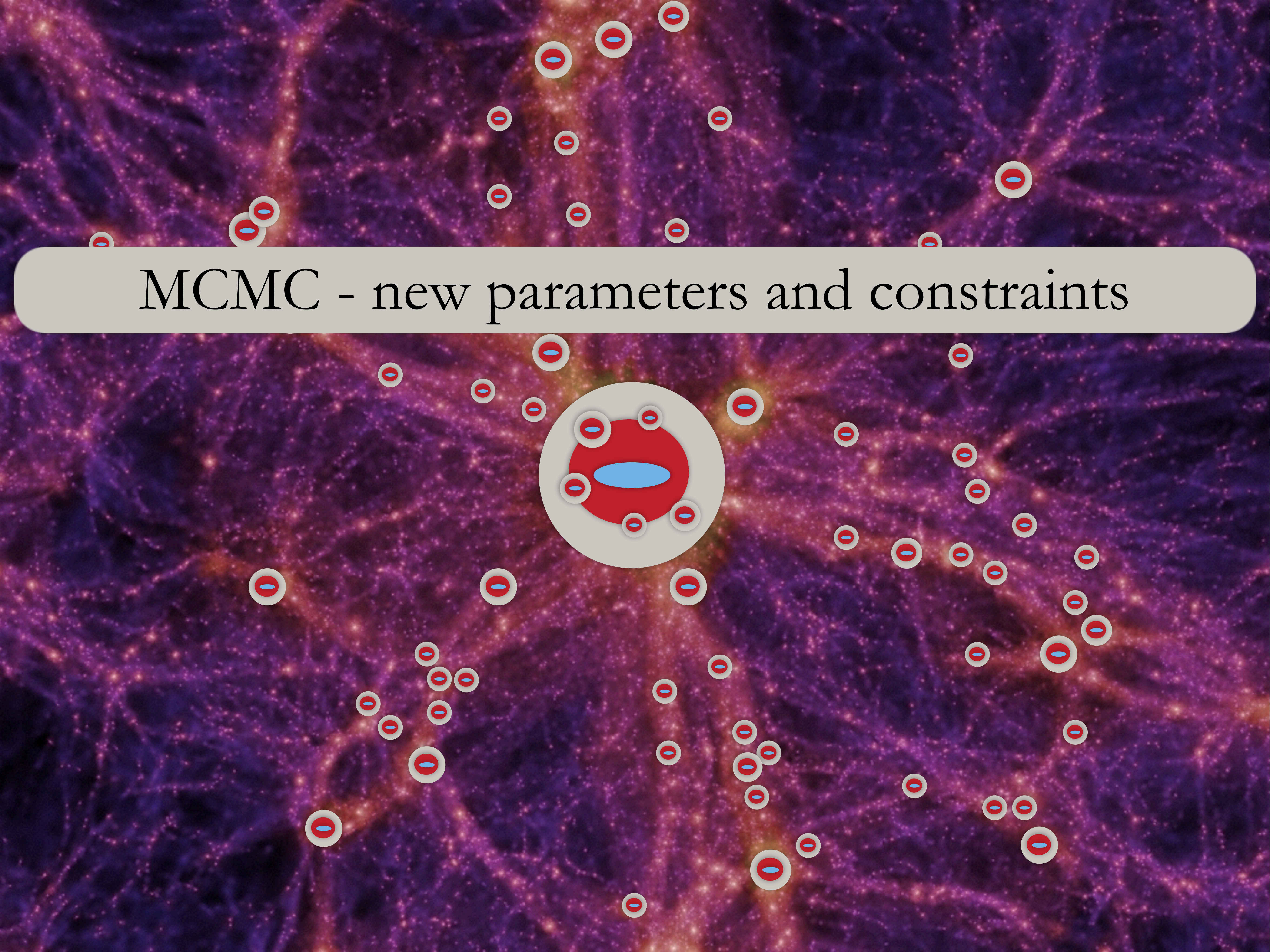
3.2.4 Dark matter merger trees:

In MCMC mode, in order to run the galaxy formation model in a reasonable time, a representative set of dark matter merger trees is used instead of the full volume of the N-body simulations. Therefore, there is a dark matter tree file containing the set of pre-selected trees (see Appendix B of Henriques et al., 2013, MNRAS, 431, 3373) and an MCMCSampleFile containing the IDs and weights of the FOF groups to be selected from those trees.

Name of sample file containing IDs and weights: 'MCMCSampleFilePrefix'_sample_allz_nh_'MCMCSampleFile'snapnum.txt

- MCMCSampleDir: directory for sample file
- MCMCSampleFilePrefix: prefix of sample file
- MCMCSampleFile: number of trees in sample file
- MCMCTreeSampleFile: name of file containing the representative sample of dark matter merger trees.

MCMC - new parameters and constraints



MCMC - new parameters

Max-Planck-Institut für Astrophysik

5. MCMC in depth

5.1 Including New Parameters:

If you would like to include additional parameters into the sampling that is relatively straightforward. If the parameter was not present before in the semi-analytic model you will have to include it in the **input_***.par** file and read it in in **read_parameters.c**. In addition, you will have to pass the parameter value from the MCMC to the semi-analytic model in function **void read_mcmc_par (int snapnum)**. Finally, you will have to include the new parameter in the input file **./input/MCMC_inputs/MCMCParameterPriorsAndSwitches.txt** and increase the value for the number of parameters at the top.

5.2 Including New Observational Constraints:

Including a new observational constraint is slightly more complex and basically involves **changes in the input files, mcmc_likelihood() and mcmc_save.c()**. The first thing to do is to increase the number of observational constraints defined in **mcmc_vars.h**: "#define MCMCNConstraints" and at the top of **./input/MCMC_inputs/MCMCObsConstraints.txt** and **./input/MCMC_inputs/MCMCWeightsObsConstraints.txt**. Then, the new observational constraint(s) must be included in these two input files. A type of statistical test needs to be specified together with switches and weights for all output redshifts. Then a file with the same name specified in these files must be added to the directory ./ObsConstraints/ for all the redshifts to be used.

After the inputs, mcmc_save.c() and mcmc_likelihood() must be changed. If a property not used before by the mcmc is needed, this must be saved in **mcmc_save.c** (section mcmc_save below). Then, a method of binning the theoretical predictions and a statistical test must be included in **mcmc_likelihood()**. Section mcmc_likelihood below explains how to use the functions already present in the code and how to include new ones.

→ **Include a new parameter**

MCMC - new constraints

5. MCMC in depth

5.1 Including New Parameters:

If you would like to include additional parameters into the sampling that is relatively straightforward. If the parameter was not present before in the semi-analytic model you will have to include it in the **input_***.par** file and read it in in **read_parameters.c**. In addition, you will have to pass the parameter value from the MCMC to the semi-analytic model in function **void read_mcmc_par (int snapnum)**. Finally, you will have to include the new parameter in the input file **./input/MCMC_inputs/MCMCParameterPriorsAndSwitches.txt** and increase the value for the number of parameters at the top.

5.2 Including New Observational Constraints:

Including a new observational constraint is slightly more complex and basically involves **changes in the input files, mcmc_likelihood() and mcmc_save.c()**. The first thing to do is to increase the number of observational constraints defined in **mcmc_vars.h**: "#define MCMCNConstraints" and at the top of **./input/MCMC_inputs/MCMCObsConstraints.txt** and **./input/MCMC_inputs/MCMCWeightsObsConstraints.txt**. Then, the new observational constraint(s) must be included in these two input files. A type of statistical test needs to be specified together with switches and weights for all output redshifts. Then a file with the same name specified in these files must be added to the directory ./ObsConstraints/ for all the redshifts to be used.

After the inputs, mcmc_save.c() and mcmc_likelihood() must be changed. If a property not used before by the mcmc is needed, this must be saved in **mcmc_save.c** (section mcmc_save below). Then, a method of binning the theoretical predictions and a statistical test must be included in **mcmc_likelihood()**. Section mcmc_likelihood below explains how to use the functions already present in the code and how to include new ones.

→ **Include a new observational constraint**