

# **Computing with CosmoTherm and CosmoRec/Recfast++**

*A Toolbox Overview*

**Boris Bolliet**

- 5': little introduction
- 10': Installing the codes and work directory
- 15': Playing with a notebook

GitHub Desktop

CosmoTherm

Atom



Jupyter Notebook

CosmoRec/Recfast++

Specdist



- I haven't had to use a git terminal command for many months now !

A screenshot of the GitHub Desktop application interface. The top bar displays the current repository as "cosmotherm.rel\_corr", the current branch as "photon\_injection\_paper...", and the status "Fetch origin Last fetched 8 minutes ago". The main area shows a message "No local changes" with a sub-message: "There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next." Below this are two call-to-action boxes: "Open the repository in your external editor" (with "Open in Atom" button) and "View the files of your repository in Finder" (with "Show in Finder" button). On the left side, there's a sidebar titled "Recent" containing a list of repositories: specdist, CCL, borisbolliet, borisbolliet.github.io, class\_sz, cobaya\_sz, Likelihoods\_sz, specdist, brinckmann, montepython\_sz, cmbspec, Photon\_Injection\_Movies\_BCB20, lsstdesc, CCL, and Other (cosmotherm.rel\_corr). A "Filter" input field and an "Add" button are also present in the sidebar.



- I haven't had to use a git terminal command for many months now !

Current Repository  
cosmotherm.rel\_corr

Changes      History

0 changed files

Current Branch  
photon\_injection\_paper\_2020

Fetch origin  
Last fetched just now

Filter      New Branch

Default Branch

- CosmoTherm      2 years ago

Recent Branches

- ✓ photon\_injection\_paper\_2020      8 hours ago
- stim-criterion      2 months ago
- control-all-runtime-memory      a month ago

Other Branches

- origin/Add-extra-ODE      10 months ago
- origin/Added-Xe-ODEs      10 months ago
- origin/Added-Xe-ODEs-II      9 months ago
- origin/Added-Xe-perturbative      9 months ago
- origin/Boris-runs      9 months ago
- origin/Boris-runs-April-2020-jens      7 months ago
- origin/CT\_dev\_Tom\_Aniso\_rework      5 months ago
- origin/CT\_dev\_anisotropies      6 hours ago
- origin/CosmoTherm-dev      8 months ago
- origin/CosmoTherm-dev-jens      a year ago
- oriain/CosmoTherm-dev-iens-add-ini...      2 years ago

Here are some friendly suggestions

Open in Atom

Show in Finder

Summary (required)

Description

Commit to photon\_injection\_paper\_2020

Choose a branch to merge into photon\_injection\_p...



- I can develop code by looking at as many files as I want simultaneously!

- I can edit files directly on sirius

Project

specdist

cosmotherm\_rel\_corr

.git

Anisotropy\_Module

Atomic\_Source\_Module

CosmoTherm\_Plugins

outputs

plots

runtimes

runfiles-photon-injection-paper

Thermalization\_Module

Tools

DS\_Store

gitignore

Cooling\_and\_absorption\_regimes.cpp

CosmoTherm

CosmoTherm.cpp

libCosmoTherm.a

Makefile

Makefile.in

README.rst

tmp\_test.ini

TODO

class\_sz

borisbolliet.github.io

Likelihoods\_sz

cobaya\_sz

montepython\_sz

yg

Recfast++.vx

SO-SZ

szfastlks

Recfast\_wrapper.py

```
73     for k, v in p_dict.items():
74         f.write(str(k) + ' = ' + v)
75     f.close()
76     subprocess.call([path_to_recfast])
77     r_dict = {}
78
79     R = np.loadtxt(p_dict['path'] for
80     try:
81         r_dict['z'] = R[:,0]
82         r_dict['Xe'] = R[:,1]
83     except IndexError:
84         a = np.empty(1)
85         a[:] = np.nan
86         r_dict['z'] = a
87         r_dict['Xe'] = a
88     #print(r_dict)
89     return r_dict
90
91     def compute_recfast_parallel(self, i):
92         dE_dz+=13.6*1.602176487e-19*DRHI *NI
93         dE_dz+=24.6*1.602176487e-19*DRHeI*NI
94
95         # dict_for_fisher = kwargs.get(
96         # sd_lib_for_fisher = kwargs.get(
97         params_values_dict = self.load_
98         params_values_dict[param_name] :
99         # if float(params_values_dict['')
100         # if params_values_dict['photons']
101         # if params_values_dict['pi_fin']
```

Collisions.Recfast.cpp

```
154     double aHe=Ric_expc_HeI_Bell(28547.0);
155
156     double CicH =aH * exp(-157821.462/Ti);
157     double CicHe=aHe* exp(-285471.174/Ti);
158
159     // collisional recombination rate (i)
160     double CciH =aH * SahaBoltz_gB0(2.0);
161     double CciHe=aHe* SahaBoltz_gB0(1.0);
162
163     // net rate per z
164     double DRHI =CciH *Ne*XHII -CicH *Xi;
165     double DRHeI=CciHe*Ne*XHeII-CicHe*Xi;
166
167     fvec[0]+= DRHI ;
168     fvec[1]+= DRHeI;
169     fvec[2]+= DRHeI;
170
171     // net cooling terms
172     dE_dz+=13.6*1.602176487e-19*DRHI *NI;
173     dE_dz+=24.6*1.602176487e-19*DRHeI*NI;
174
175     fvec[4]+=fheat*dE_dz;
```

class.c

```
50     return _FAILURE_;
51 }
52
53 if (perturb_init(&pr,&ba,&th,&pt) == _FAILURE_) {
54     printf("\n\nError in perturb_init \n=>%s\n",pt.error_message);
55     return _FAILURE_;
56 }
57
58 if (primordial_init(&pr,&pt,&pm) == _FAILURE_) {
59     printf("\n\nError in primordial_init \n=>%s\n",pm.error_message);
60     return _FAILURE_;
61 }
62
63 if (nonlinear_init(&pr,&ba,&th,&pt,&pm,&nl) == _FAILURE_) {
64     printf("\n\nError in nonlinear_init \n=>%s\n",nl.error_message);
65     return _FAILURE_;
66 }
67
68 for (i=0;i<ptsz->ln_M_size;i++) {
69     x=exp(ptsz->ln_x_for_pp[i]);
70
71     plc_gnfw(&plc_gnfw_at_x,x,pvectsz,pba,ptsz);
72
73     array_for_integral[i*index_num+index_y]= log(x);
74     array_for_integral[i*index_num+index_y]= pp_at_x_and_ell_over_ell;
75 }
76
77 if (lensing_init(&pr,&pt,&sp,&nl,&le) == _FAILURE_) {
```

utils.py

```
44374 // i=0;
44375 // index_X=i;
44376 // i++;
44377 // index_y=i;
44378 // index_ddy=i;
44379 // i++;
44380 // index_num=i;
44381 // index_ddy=i;
44382 // index_num=i;
44383 // double integrand_value = 0.;
44384 // class_alloc(array_for_integral,
44385 //               ptsz->ln_M_size*index_num*sizeof(double),
44386 //               ptsz->error_message);
44387 //               ptsz->error_message);
44388 //               for (i=0;i<ptsz->ln_M_size;i++) {
44389 //               x=exp(ptsz->ln_x_for_pp[i]);
44390 //               plc_gnfw(&plc_gnfw_at_x,x,pvectsz,pba,ptsz);
44391 //               plc_gnfw(&plc_gnfw_at_x,x,pvectsz,pba,ptsz);
44392 //               array_for_integral[i*index_num+index_y]= log(x);
44393 //               array_for_integral[i*index_num+index_y]= pp_at_x_and_ell_over_ell;
44394 //               }
44395 //               double pp_at_x_and_ell_over_ell_char = x*plc_gnfw_at_x;
44396 //               array_for_integral[i*index_num+index_y]= log(x);
44397 //               array_for_integral[i*index_num+index_y]= pp_at_x_and_ell_over_ell;
44398 //               }
44399 //               }
44400 //               }
44401 //               }
44402 //               
```

config.py

```
1 import subprocess
2
3
4 path_to_sd_projects = "/Users/boris/Work/SPECTRAL-DISTORTIONS/"
5 #path_to_sd_projects = "/scratch/nas_chluba/specdist/bolliet/cosmotherm_projects/"
6
7 #path to the cosmotherm database
8 path_to_ct_database = path_to_sd_projects + "specdist/specdist/data/ct_database/"
9
10
11 #the path to the cosmotherm binary file
12 path_to_cosmotherm = path_to_sd_projects + "cosmotherm.rel_corr"
13
14 #the path to save the output from cosmotherm
15 path_to_ct_spectra_results = path_to_sd_projects + "specdist/specdist/ct_spectra"
16
17 subprocess.call(['mkdir','-p',path_to_ct_spectra_results])
18
19
20 #the path to the Recfast binary file
21 path_to_recfast = path_to_sd_projects + "Recfast++.vx"
22
23 #the path to save the output from cosmotherm
24 path_to_recfast_results = path_to_sd_projects + "specdist/specdist/recfast_outputs"
25
26 subprocess.call(['mkdir','-p',path_to_recfast_results])
27
```

Recfast++.cpp

```
1 // Author: Jens Chluba & Luke Hart
2 // First implementation: Oct 2010
3 // CITA, University of Toronto
4 // All rights reserved.
5 //
6 // 03.03.2020: fixed Saha solution with correct reduced mass
7 // 20.05.2018: all recombination settings are now centrally controlled (JC)
8 // 23.04.2018: added structures to control all the parameters and data passed
9 // 21.02.2018: added cosmic time setup for internal use and decaying particle
10 // 10.01.2018: compute better high-z Te solution (JC)
11 // 03.04.2017: added switches using Saha and changed redshift grid (JC)
12 // 04.01.2017: added new routines for separate H and HeI correction functions
13 // 05.12.2016: added dependence of Saha_HeIII on fundamental constants
14
15 //=====
16 //=====
17 // Main Recfast++ header with all required function to access recombination
18 // CosmoRec. the solution to the RECFAST ODE-system is computed. Different
19 // -----
20 // -----
21 // important note: all free particle fractions are relative to the number of
22 // nuclei. I.e. xe=ne/nh, xp=np/nh etc.
23 // =====
24 // =====
25 // include <stdio.h>
26 // include <iostream>
27 // include <fstream>
28 // include <string>
```

massfunction.py

```
56 // tool to add y-line: halo_bias.
57
58 mdef = hal.MassDef(overdensity, 'matter')
59 mf_par = cosmo._config_init_kwargs['mass_function']
60 if mf_par == 'tinker10':
61     bf = hal.HaloBiasTinker10(cosmo, mdef)
62 elif mf_par == 'shethorm':
63     bf = hal.HaloBiasSheth99(cosmo)
64 else:
65     bf = hal.HaloBiasTinker10(cosmo, mdef)
66     #raise CCLError("No b(M) fitting function implemented for "
67     #               "mass_function_method: "+mf_par)
68 return bf.get_halo_bias(cosmo,
69                         halo_mass,
70                         a)
71
72 @deprecated(hal.mass2radius_lagrangian)
73 def massfunc_m2r(cosmo, halo_mass):
74     """Converts smoothing halo mass into smoothing halo radius.
75
76     .. note:: This is :math:`R=(3M/(4\pi\rho_M))^{1/3}` , where
77             :math:`\rho_M` is the mean comoving matter density.
78
79     .. note:: Note that this function is deprecated. Please use
80             :meth:`~pyccl.halos.massdef.mass2radius_lagrangian`.
81
82 Args:
83     cosmo (:class:`~pyccl.core.Cosmology`): Cosmological parameters.
84     halo mass (float or array like): Halo masses: Msun.
```

No results found for 'rho\_nfw'

rho\_nfw

use\_halo\_model

Finding with Options: Case Insensitive .,\* Aa ⌂ Find Find All Replace Replace All



- I can make nice figures and movies!

In [1]:

```
1 import specdist as pi
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import matplotlib
5 from scipy.interpolate import interp1d
6 from matplotlib.pyplot import cm
7 import matplotlib.ticker as ticker
8 from mpl_toolkits.axes_grid1.inset_locator import inset_axes
9 from matplotlib.collections import LineCollection
10 from matplotlib.ticker import LogLocator
11 import re
12 scientific_notation = pi.scientific_notation
13 from scipy.linalg import cholesky, LinAlgError
14 from scipy.linalg import block_diag
15 from numpy import linalg as LA
16 from scipy import optimize
17
18 from matplotlib.ticker import (MultipleLocator, FormatStrFormatter,
19                               AutoMinorLocator)
20
21 import warnings; warnings.simplefilter('ignore')
22
23 path_to_figures = "/Users/boris/Work/SPECTRAL-DISTORTIONS/figures/"
24
25 xinj_to_mass = 4.698e-4 #m in ev = 4.698e-4 xinj
26 xinj_to_Einj = 4.698e-4/2. #Einj = 4.698e-4/2 xinj
27 mass_to_Einj = 1./2.
28 #1./4.698e-4*2. = 4257.130693912303
29
```



- I can make nice figures and movies!
- But wouldn't advise to use for development.



Logout

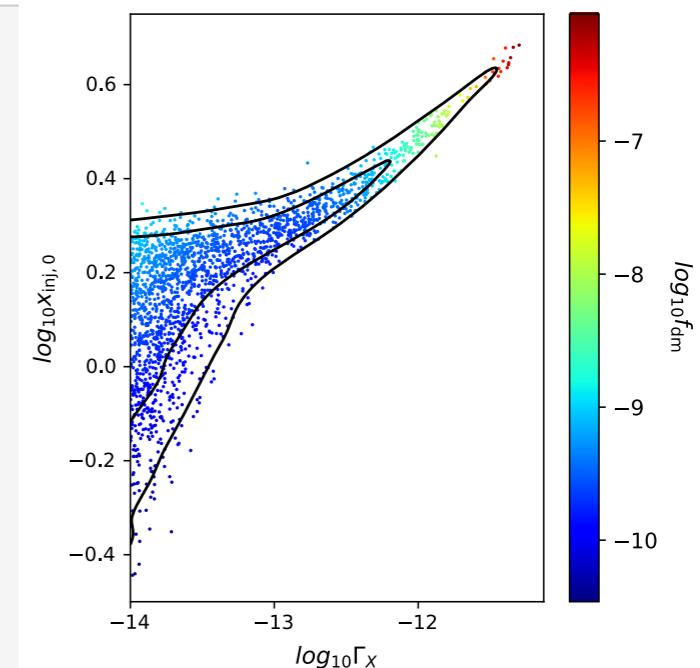
File Edit View Insert Cell Kernel Widgets Help

Not Trusted

Python 3



```
In [1]: 1 import specdist as pi
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import matplotlib
5 from scipy.interpolate import interp1d
6 from matplotlib.pyplot import cm
7 import matplotlib.ticker as ticker
8 from mpl_toolkits.axes_grid1.inset_locator import inset_axes
9 from matplotlib.collections import LineCollection
10 from matplotlib.ticker import LogLocator
11 import re
12 scientific_notation = pi.scientific_notation
13 from scipy.linalg import cholesky, LinAlgError
14 from scipy.linalg import block_diag
15 from numpy import linalg as LA
16 from scipy import optimize
17
18 from matplotlib.ticker import (MultipleLocator, FormatStrFormatter,
19                               AutoMinorLocator)
20
21 import warnings; warnings.simplefilter('ignore')
22
23 path_to_figures = "/Users/boris/Work/SPECTRAL-DISTORTIONS/figures/"
24
25 xinj_to_mass = 4.698e-4 #m in ev = 4.698e-4 xinj
26 xinj_to_Einj = 4.698e-4/2. #Einj = 4.698e-4/2 xinj
27 mass_to_Einj = 1./2.
28 #1./4.698e-4*2. = 4257.130693912303
29
```



- Other tools:

掣 [brinckmann / montepython\\_public](#)  
forked from [baudren/montepython\\_public](#)



latest

## CLASS\_SZ

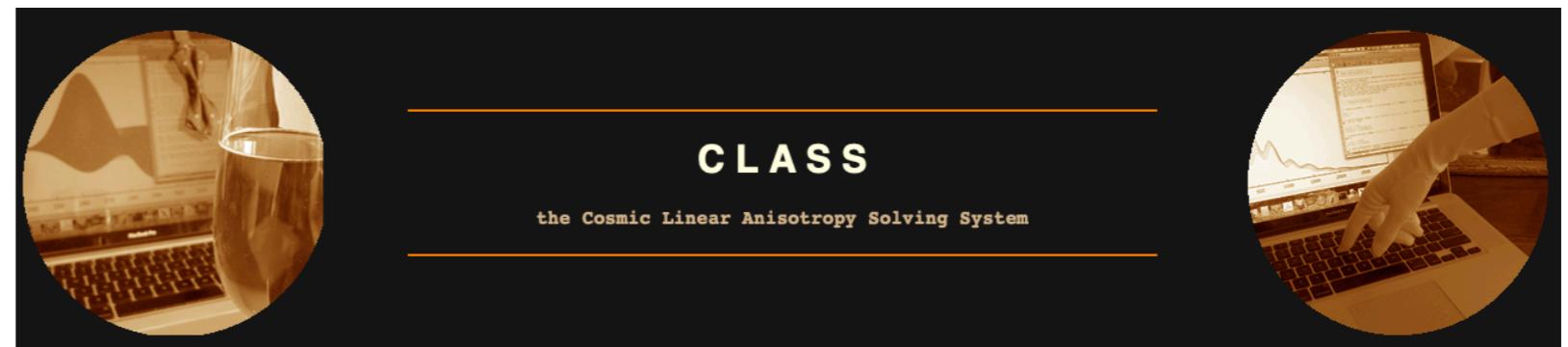
Cosmic Linear Anisotropy Solving System

With Thermal Sunyaev Zeldovich Power Spectrum Computation

- *In development: ALL possible halo model auto and cross-correlation power spectra*

## GetDist

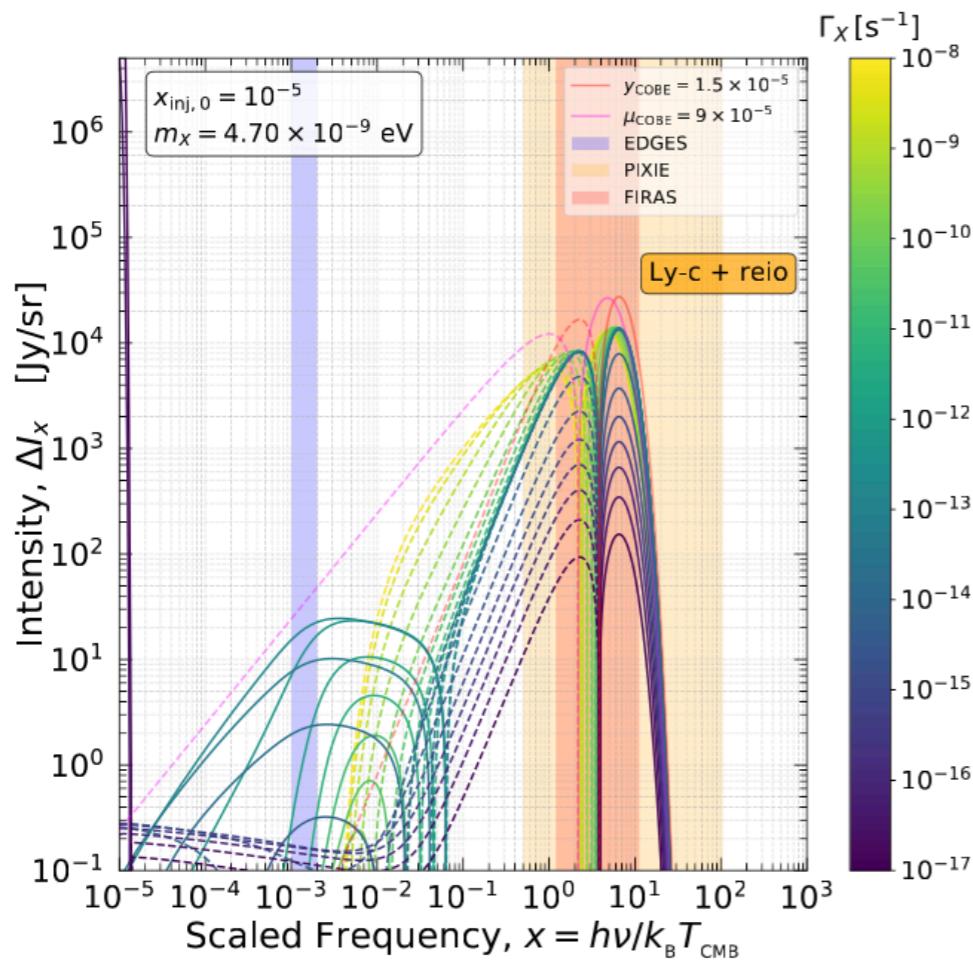
GetDist is a Python package for analysing and plotting Monte Carlo (or other) samples.



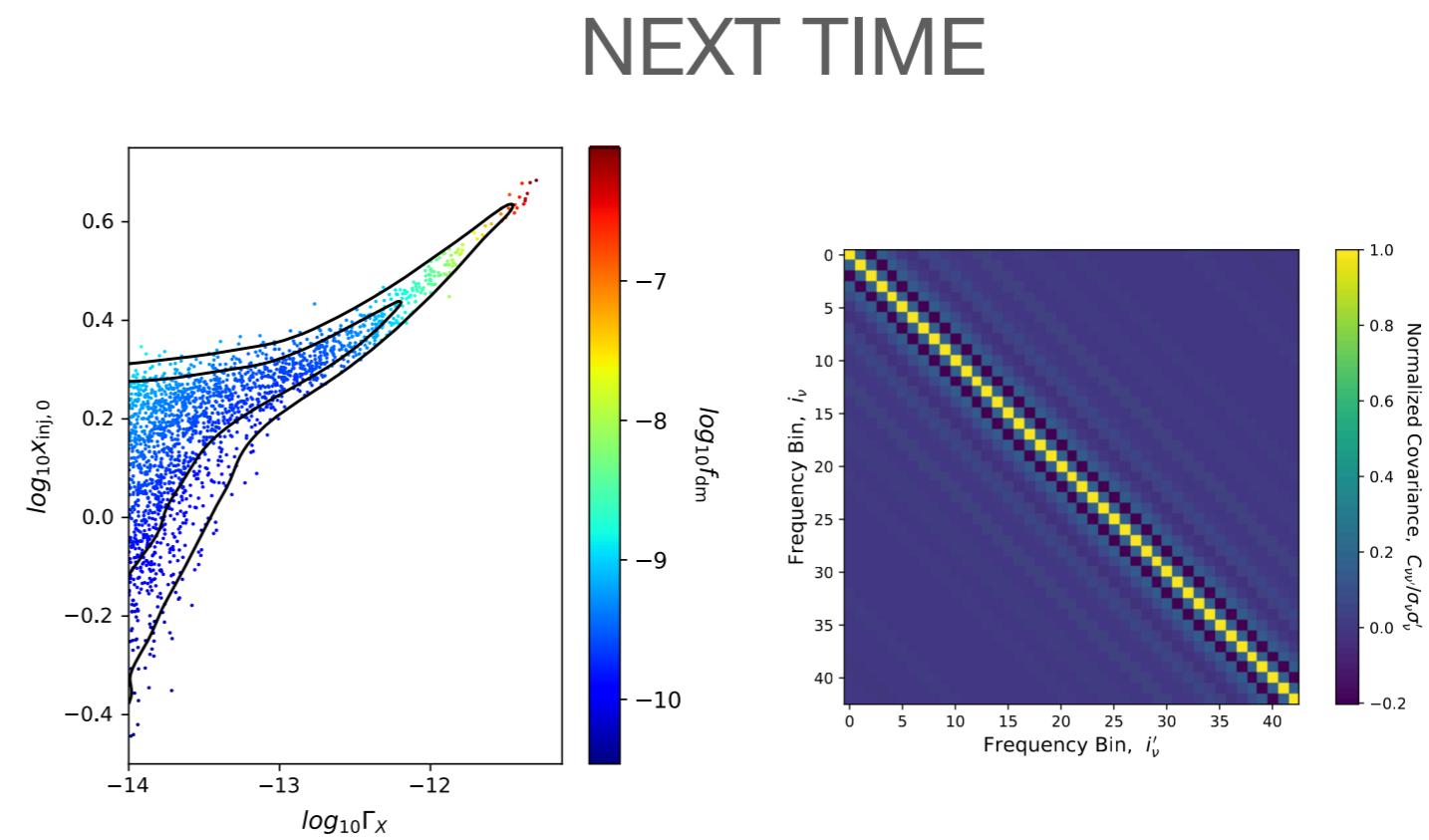
# Plan

- Today: Specdist + Photon Injection Library
- Next Time: MCMC/Fisher's with Spectral Distortion Data

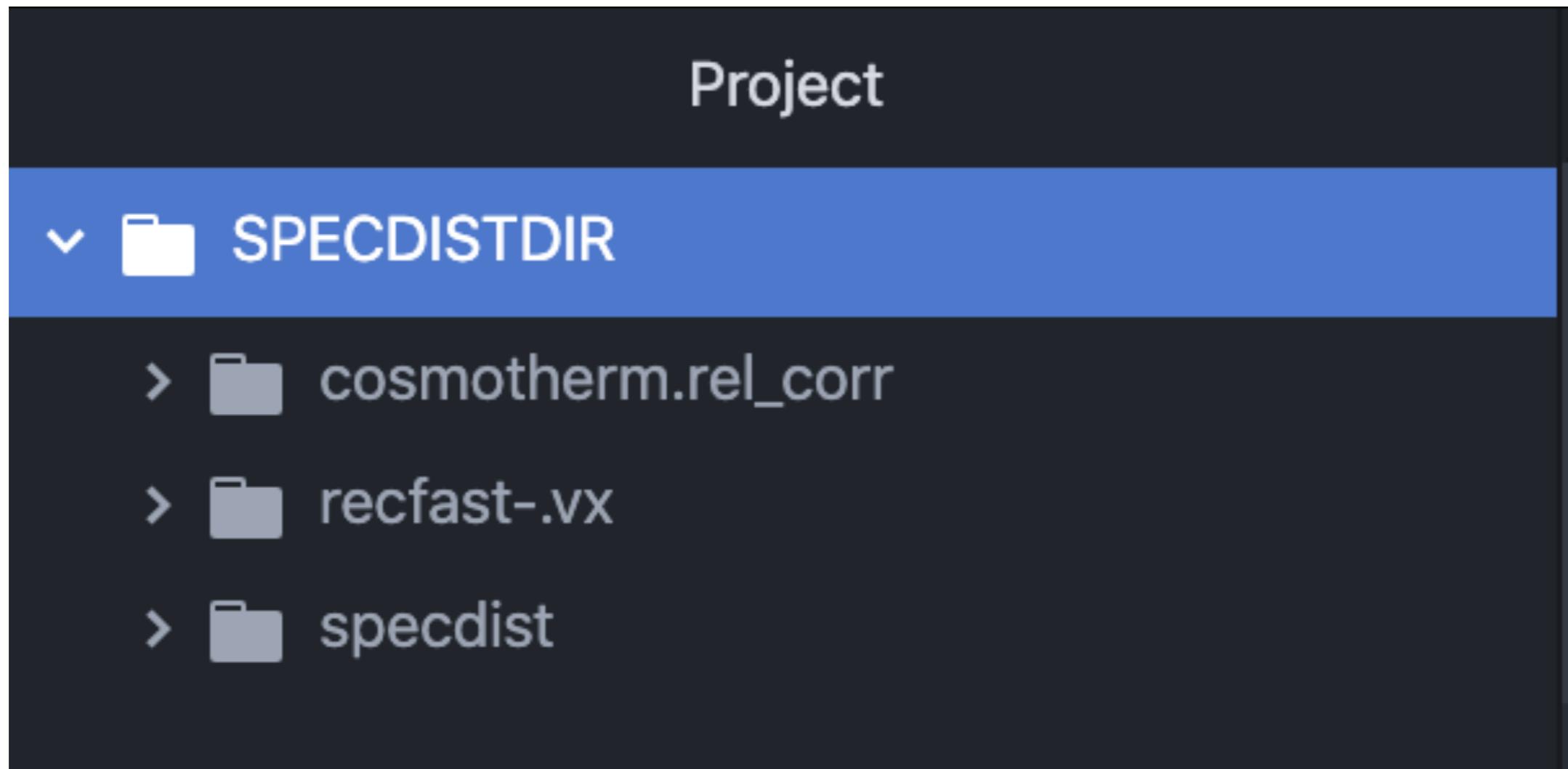
TODAY



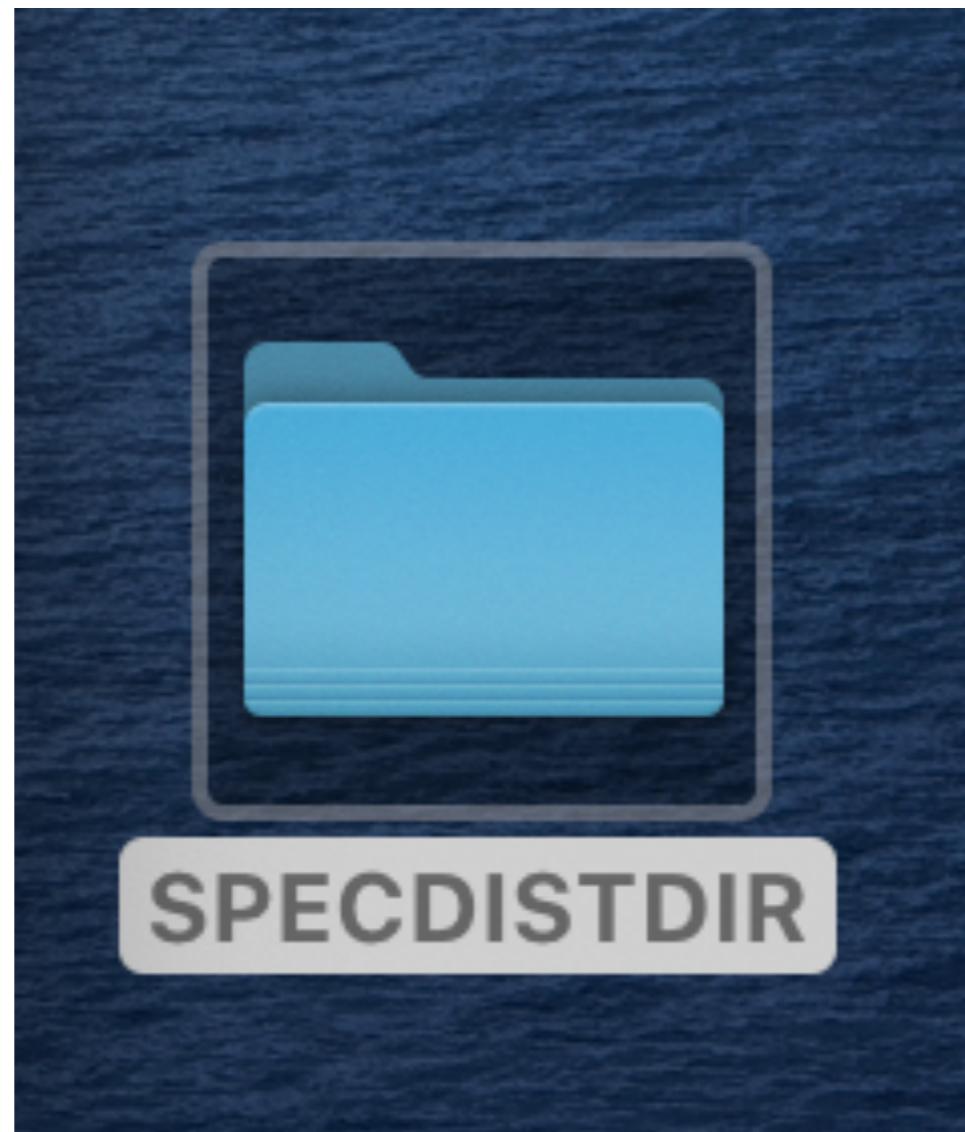
NEXT TIME



- Let's put all the useful codes for today in a directory
- Let's do it step by step together and install everything



- Create Directory **SPECDIR** on your Desktop



- \$ cd SPECDISTDIR



CosmoTherm.rel\_corr

- \$ git clone [https://xxx@bitbucket.org/Jacetoto/cosmotherm.rel\\_corr.git](https://xxx@bitbucket.org/Jacetoto/cosmotherm.rel_corr.git)

- \$ git fetch origin

- \$ git checkout -b photon\_injection\_paper\_2020 origin/photon\_injection\_paper\_2020

- In cosmotherm.rel\_corr/Makefile.in make sure you are using the correct path to gsl. You may need to change to:

```
#=====
# GSL lib
#=====
GSL      = gsl
GSL_INC_PATH = /usr/local/include/
GSL_LIB_PATH = /usr/local/lib/
LIBSGSL = -L$(GSL_LIB_PATH) -l$(GSL) -lgslcblas
```

- In the same file, make sure you comment out the grace options:

```
#=====
# grace lib (for outputs with xmgrace from the diffusion modules)
#=====
# If grace support should be used uncomment the next and set the subsequent
# definitions accordingly.
#=====
# USE_GRACE = -D GRACE_DEFINED
# #
# ifeq ($(USE_GRACE),-D GRACE_DEFINED)
#   GR      = grace_np
#   GR_INC_PATH = /opt/local/include/
#   GR_LIB_PATH = /opt/local/lib/
# 
#   LIBSGR = -L$(GR_LIB_PATH) -I$(GR)
# endif
```

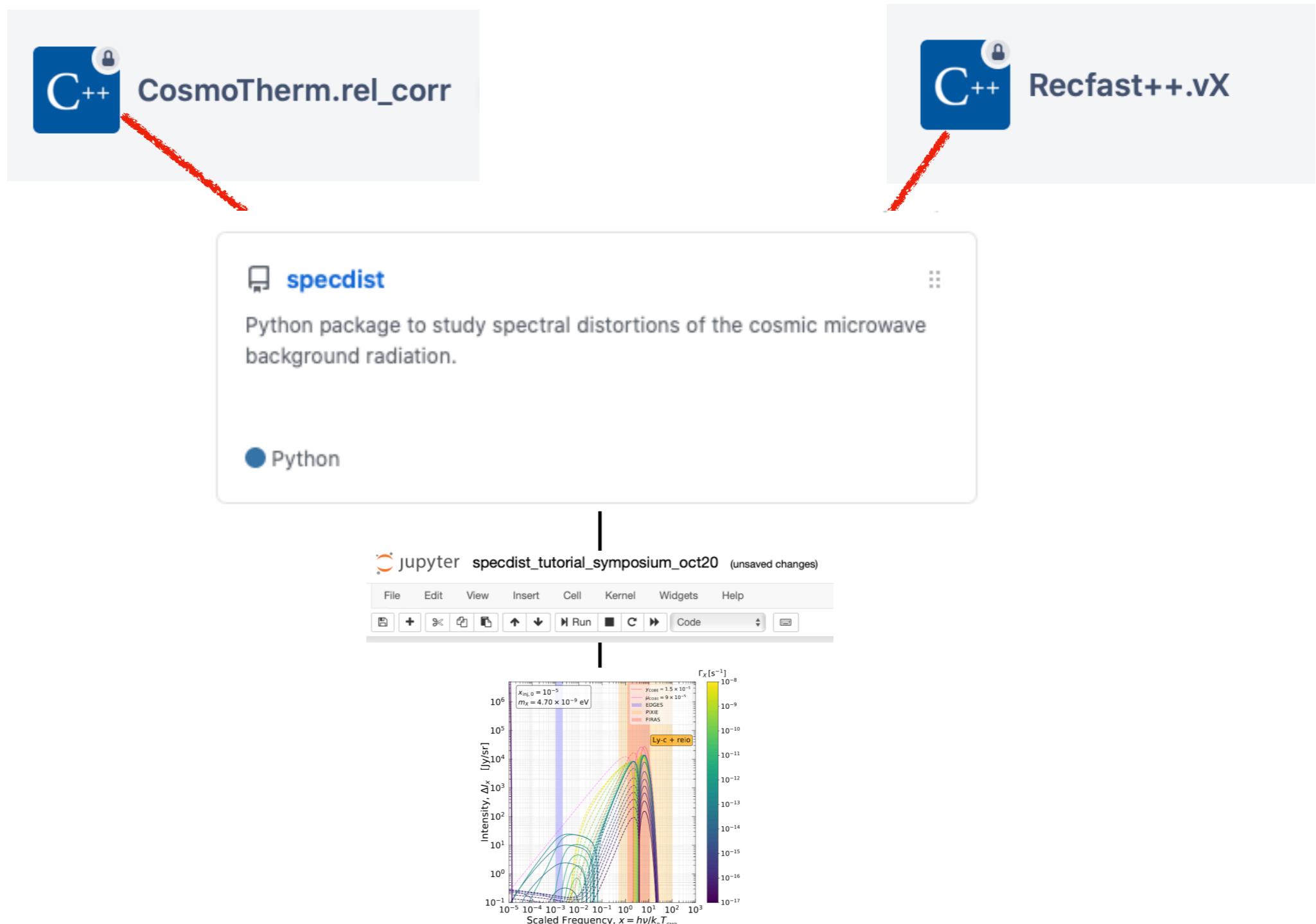
- Still in makefile.in make sure to use a g++ version you have

```
#=====
# choose c++ compiler
#=====
CC= g++-10
#CC= g++-mp-6
```

- \$ make tidy
- \$ make
- \$ ./CosmoTherm runfiles/parameters.ini
- That's all good!

- \$ cd SPECDISTDIR
- \$ git clone <https://xxxx@bitbucket.org/Jacetoto/recfast-.vx.git>
- \$ git checkout -b add-decay-heating-scenario origin/add-decay-heating-scenario
- \$ make tidy
- \$ make
- (again make sure g++ version in makefile is fine)
- \$ ./Recfast++ runfiles/parameters.ini
- That's all good!

- `$ cd SPECDISTDIR`
- `$ git clone https://github.com/borisbolliet/specdist.git`
- Before we can use specdist, we need to set-up the **paths** to the codes



- Open the file SPECDISTDIR/specdist/specdist/config.py
- Set the correct paths **there**:

```
path_to_sd_projects = "/Users/boris/Desktop/SPECDISTDIR/"

#path to the cosmothemr database
path_to_ct_database = path_to_sd_projects + "specdist/specdist/data/ct_database/"

#the path to the cosmothemr binary file
path_to_cosmothemr = path_to_sd_projects + "cosmothemr.rel_corr"

#the path to save the output from cosmothemr
path_to_ct_spectra_results = path_to_sd_projects + "specdist/specdist/ct_spectra"
subprocess.call(['mkdir', '-p', path_to_ct_spectra_results])

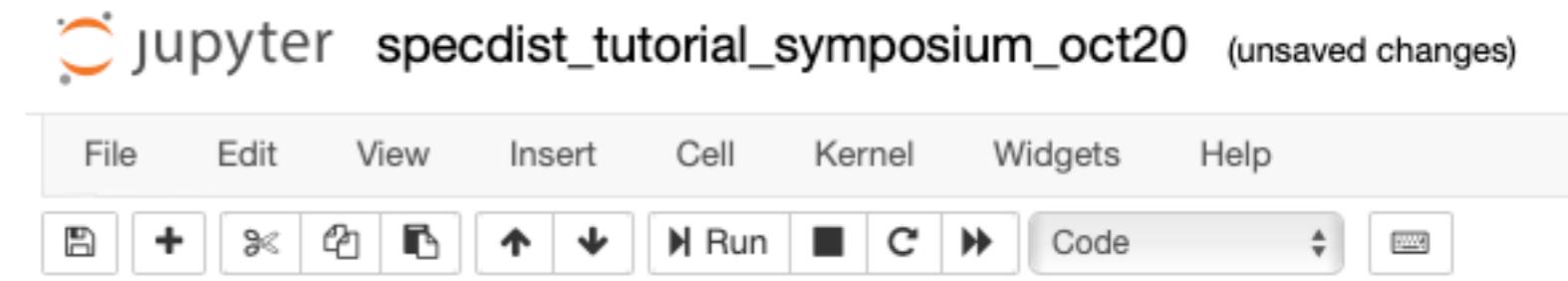
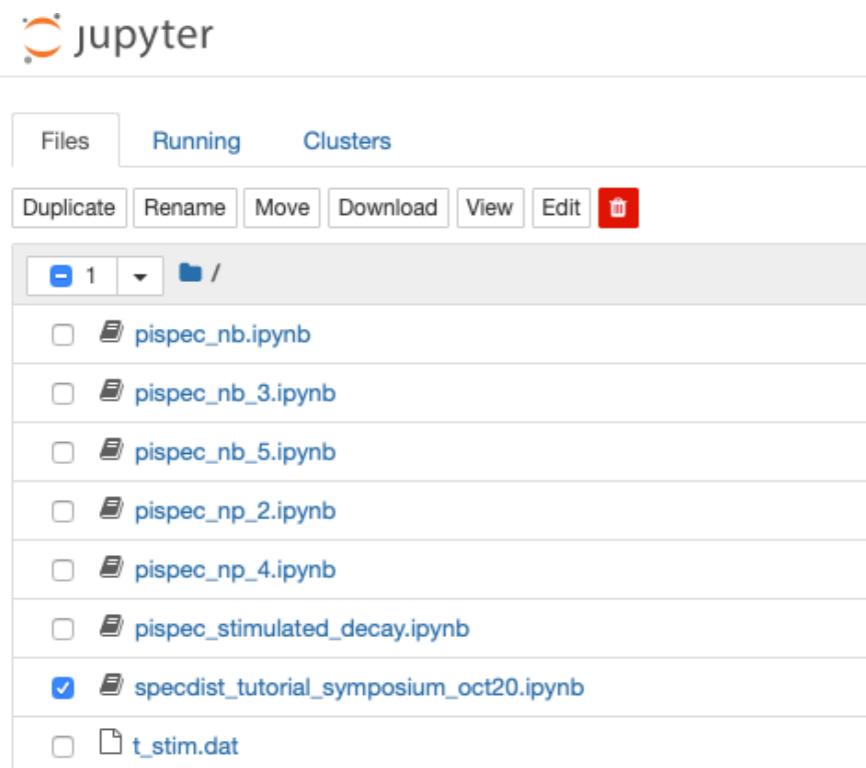
#the path to the Recfast binary file
path_to_recfast = path_to_sd_projects + "recfast-.vx"

#the path to save the output from cosmothemr
path_to_recfast_results = path_to_sd_projects + "specdist/specdist/recfast_outputs"
subprocess.call(['mkdir', '-p', path_to_recfast_results])
```

- \$ cd SPECDISTDIR/specdist/
- \$ python setup.py install
- \$ python
  - >> import specdist as sd
  - >> exit()
- Just for later on we need to copy a param file for recfast:
- \$ cp specdist/run\_scripts/recfast-parameters.ini ..//recfast-.vx/runfiles
- That's all good!
- Check that you see the subdirectories:
  - ✓ specdist/ct\_spectra
  - ✓ specdist/recfast\_outputs



- \$ cd SPECDISTDIR/specdist/specdist/notebooks
- \$ jupyter notebook
- Select specdist\_tutorial\_symposium\_oct20.ipynb



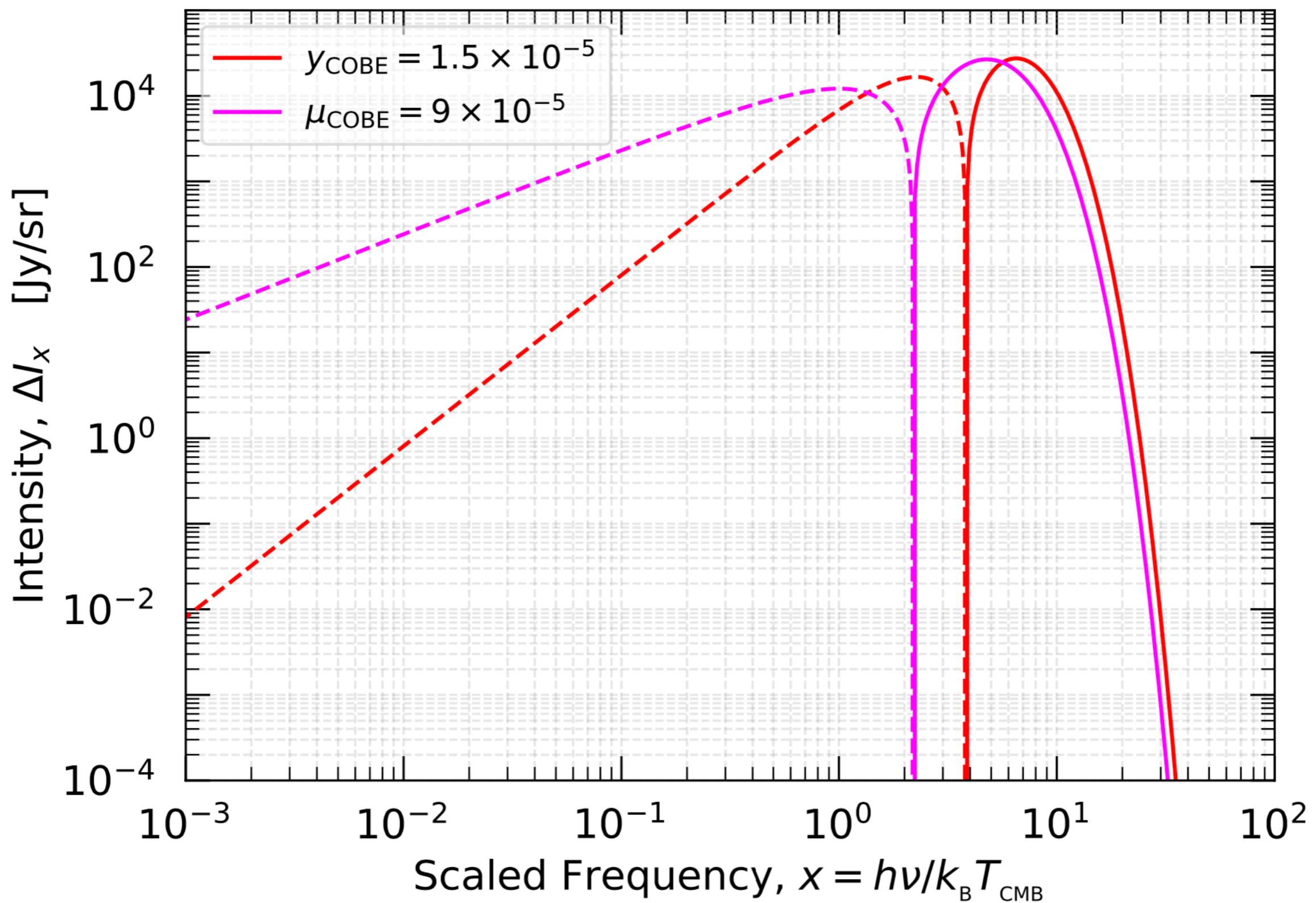
File Edit View Insert Cell Kernel Widgets Help



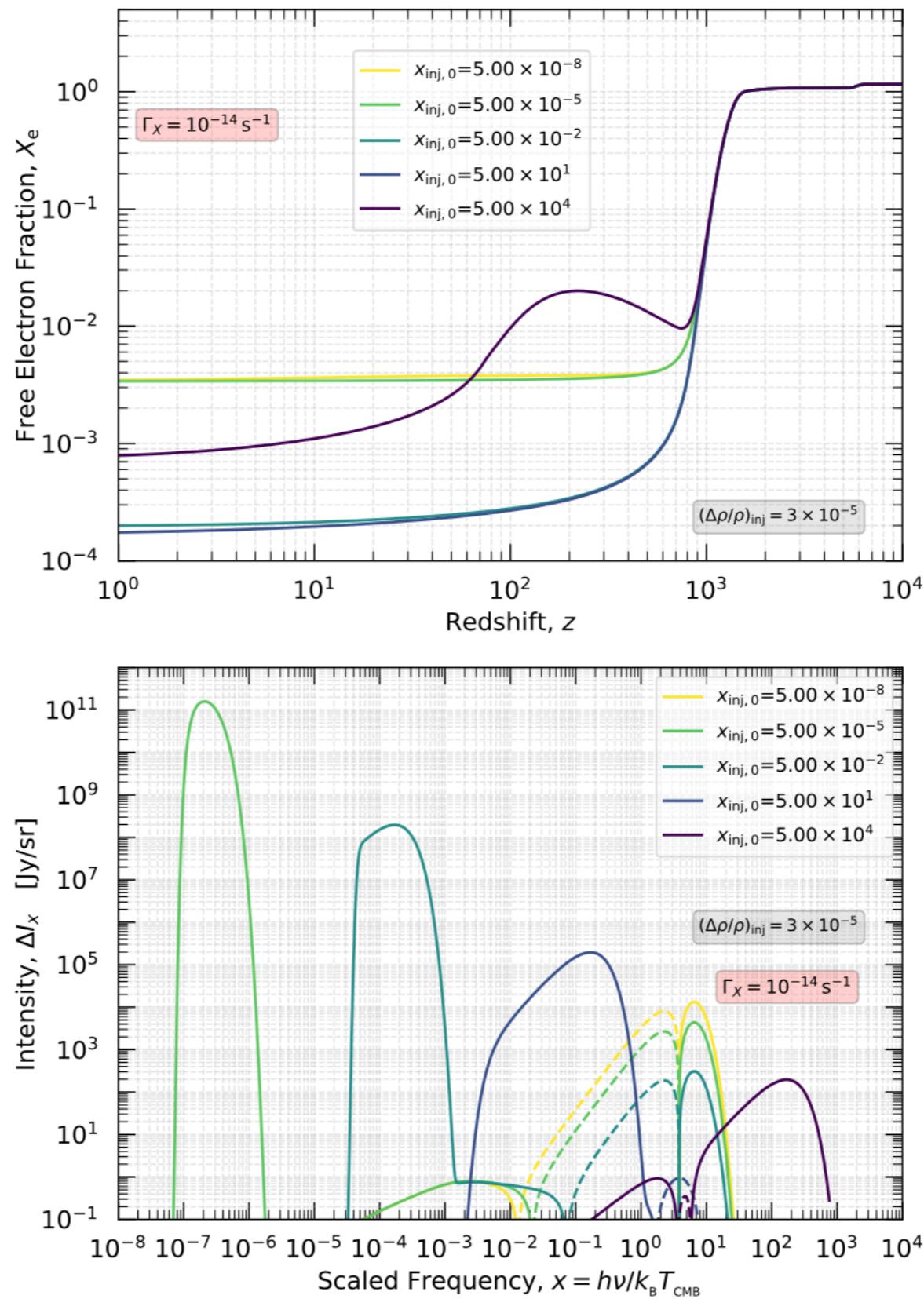
- Plot standard mu and y distortions
- Run CosmoTherm in parallel and plot results
- Run CosmoRec/Recfast++ in parallel and plot results
- Make a movie of spectra using the photon injection library

*That's it for the slide show, everything else should happen on the notebook now!*

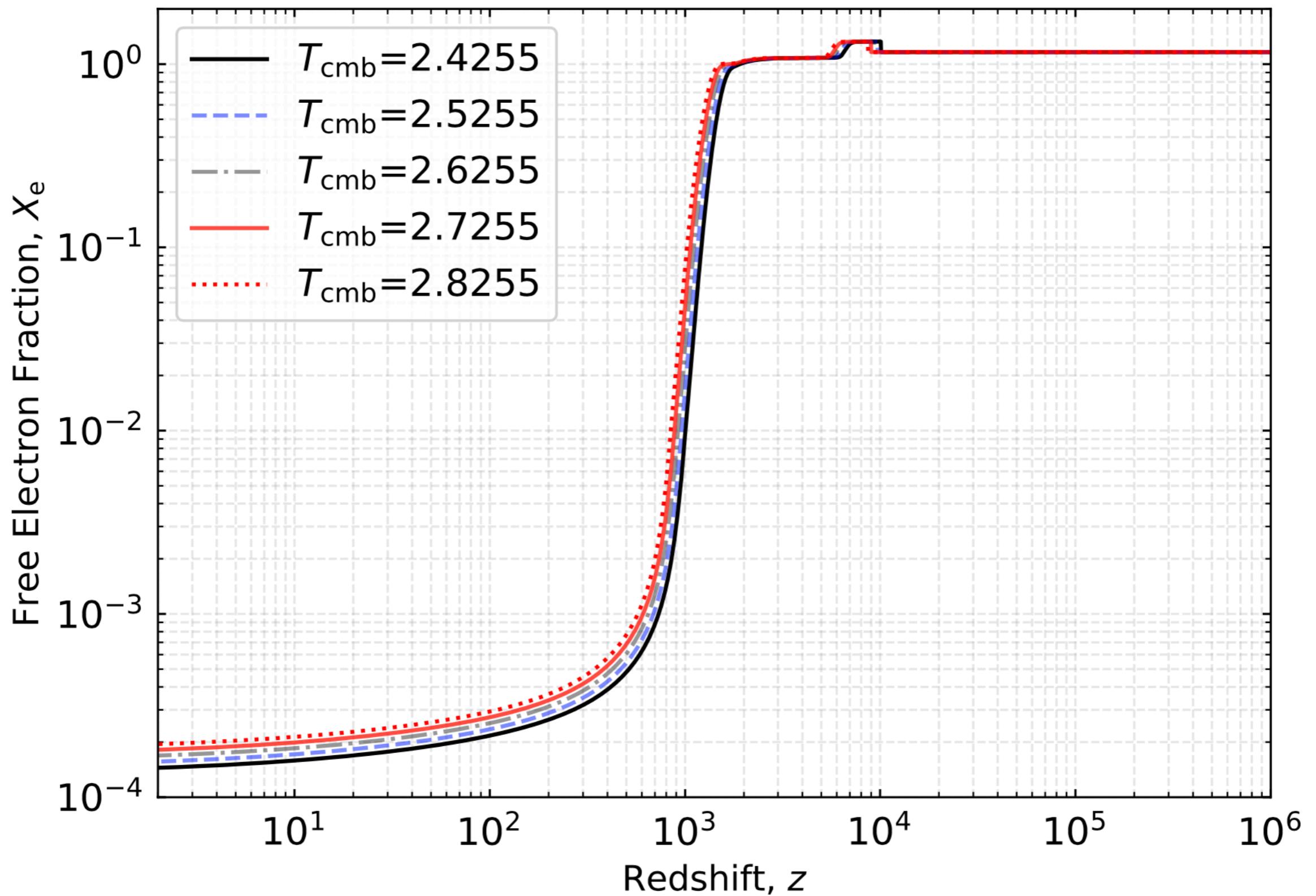
- Plot standard mu and y distortions



- Run CosmoTherm in parallel and plot results



- Run CosmoRec/Recfast++ in parallel and plot results



- Make a movie of spectra using the photon injection library

