The scripts are in IDL 7.0 and Python 3.6; they have not been tested for different versions of the same programming languages.

Contents

The scripts perform four different functions overall, so they can be split in as many categories (in the following, scripts are in IDL 7.0 unless otherwise specified):

- *Opacity reprocessing:* Taking the opacity files recovered from the various laboratories and rewriting them in a coherent fashion (i.e. with the same wavelength grid, naming convention, etc.). The only script in this category is `opacity_reprocess.pro`.

- *SED creation:* The main script is `create_FIR_SED.pro`. It requires to compile `grams_synthphot.pro` and `physconst.pro` before running. It calls, or otherwise needs for running, the following scripts: `makesed.pro`, `make_opac.pro`, `uncphot.pro`, `makesynthphot.pro`, `synthphot_alma.pro`, `ss_bbfunc.pro` (percentiles.pro?). The script creates two types of output: dust continuum spectra and its related photometry.

- *Fitting:* The Python script `photometry_fit.py` makes a full fit to the synthetic photometric bands using MCMC; it outputs a file of the results (normally: dust mass, temperature, and opacity index beta), plus walker and corner plots for the MCMC process. The IDL script `fit2bands.pro` applies a χ^2 fit to two-band photometry to simulate the (typically data-poor) analysis of high-redshift observations.

- *Plotting:* Scripts that turn the result files into the plots of the article (Not yet added to the new Git repository).

The data files used can be separated into six categories, spread through five subfolders:

- *"Original" opacities:* In the `MAC_files_original/` folder. These are the opacity files as they are immediately after download (see "Obtaining the original opacity files" below).

- *Reprocessed opacities:* In the `MAC_files_reprocessed/` folder. These are the opacities files as reprocessed by `opacity_reprocess.pro`.

- *Filter profiles:* These are not in any subfolder, but in the main folder with the scripts. They are needed to obtain photometry from the dust emission spectra. `grams_synthphot.pro` provides non-ALMA filters for the IDL scripts; a series of `.dat` files includes non-ALMA filters for the Python script and ALMA filters for all scripts.

- *Synthetic SEDs:* In the `synthetic_SEDs/` folder, forther subdivided in the `Spectra/` and `Photometry/` subfolders. In addition to selected-band photometry, full photometry (including low-quality bands) is included in `Photometry/Redundant_bands/`, although the fit does not use it.

- *Fit results:* In the `fit_result_files/` folder: `.dat` files for dust mass, temperature, and beta plus asymmetric error bars.

- *Plots:* In the `plots/` folder (Coming soon).

Requirements

You will need the following to run the IDL codes (all are already included in the repository, but I may have to take out the code that's not Sundar's, or figure out the proper citations to add):

- Sundar Srinivasan's synthetic photometry code, which includes `grams_synthphot.pro`, `physconst.pro`, `madm.pro`, `ss_bbfunc.pro` and `GRAMS_filters.fits`;

- Chris Beaumont's `medabsdev.pro` to calculate median absolute deviations, downloadable from `https://www.ifa.hawaii.edu/users/beaumont/code/medabsdev.html`

- `percentiles.pro`, downloadable from `http://www.heliodocs.com/xdoc/xdoc_print.php?file=$SSW/packages/s3drs/idl/util/percentiles.pro`

Obtaining the original opacity files

Download the Demyk 2017 data from here: `https://www.sshade.eu/db/stopcoda` and save them in the `MAC_files_original` folder. To get the Mennella data you need to ask him by mail. (I included it in the Git repository for now, but I'll take it out of the final version)

Running the scripts

IDL scripts: `samplesession.pro` shows how you are supposed to run the thing: reprocess the data (you should need to only do this once), set up the parameters for the SED creation (but the script can use default values if you don't) and run `create_FIR_SED.pro`. If the script does create the photometry (synthetic_SEDs folder) please check that results are consistent with the previous version: go to the older Github repository `https://github.com/ICSM/Synthetic_photometry_and_fits` and compare the files with the like-names ones in the synthSED_v08 folder (small differences are OK since we're using Monte-Carlo).

Python script: To use the Python fitting code `photometry_fit.py`, start by editing the parameters on lines 323 to 369 to determine the type of fit you want:

- Number of parameters (3 = free-beta fit; 2 = fixed-beta fit).

- Sed type to fit: single-temperature or two-temperatures. (Also possible but not used: single temperature with CMB heating correction)

- Priors (standards if flat in T and β, but one can choose Gaussian for T and/or M31 distribution for β).

- Composition of SED to fit (standard: E30R-70.0+BE-30.0)

- Length of MCMC chain (two choices: "fast" or "long"; latter is $\sim 30\times$ slower).

The fits used in the article can be obtained by running four combinations: sedtype = '1T' / '2T'; comp = "MBBtest" / "E30R-70.0+BE-30.0". This is by far the longest-running script; on my machine (specs) each single fit requires ~ 4 minutes ("fast" run), for a total of ~ 10 hours for a full run of 160 fits (8 redshift $\times$ 10 temperatures, repeated for reduced opacity).

To be added: IDL scripts for two-band fit, plots.