COMTAILS tutorial

Fernando Moreno (IAA-CSIC)

May 5, 2025

The code generates a tail brightness image for a given target at a given epoch, given the physical parameters described in the input files below. At each time step between the start of the activity and the observation date, the dust is considered to be uniformly ejected from the selected surface area of the assumed spherical target. At present, there are two versions of `COMTAILS` in FORTRAN available, one is a `FORTRAN` serial, available in branch `FORTRAN_SERIAL`, and another is a `FORTRAN` Message Passing Interface (MPI) parallel version, available in branch `FORTRAN_PARALLEL`. In addition, a parallel python version is also available at the repository: `https://github.com/FernandoMorenoDanvila/py_COMTAILS`

There are two main input files to `COMTAILS`:

(A) File `TAIL_INPUTS.dat`

Lines 1-25 correspond to different parameters as follows:

(1) Target identification, rec # in JPL Horizons, refer to target id, and specific orbit. Check that the output gives the desired target name, as sometimes this id changes from time to time for some reason.

(2) Particle density assumed, in kg m^{-3}.

(3) Particle geometric albedo, and linear phase coefficient. Note that this coefficient is not used in the present version of the code, where the Schleicher phase function is used instead.

(4) Particle ejection regime. This parameter has the following optional values:

`IEJEC_MODE`=1 for isotropic ejection

`IEJEC_MODE`=2 for ejection only from the sunward hemisphere

`IEJEC_MODE`=3 for ejection from a selected latitude-longitude region on a rotating nucleus characterized by certain obliquity and argument of subsolar meridian at perihelion.

(5) Obliquity, argument of subsolar meridian at perihelion (both in degree), and rotational period (in days). Ignored if values in (4) are 1 or 2. The spherical nucleus is assumed to be rotating in simple mode, i.e., constant direction of the spin axis and constant angular velocity.

(6) Latitude region of activity (minimum latitude, maximum latitude, in degree).

(7) Longitude region of activity: (minimum longitude, maximum longitude, in degree).

The code perform a pre-sampling on the selected area to calculate the actual fraction of the region which is illuminated within the selected latitude-longitude box, using the Monte Carlo number of events selected (line (18) below). The assumed dust loss rate is then reduced according to the fraction of the area which is actually illuminated at each time step.

(8) This parameter is equal to 1 if the ejection is controlled by solar illumination. Otherwise it is set to 0 (e.g. active asteroids where activity is triggered by an impact or by rotational disruption). In that case, the pre-sampling above is not performed.

Note that lines (5) to (8) are ignored unless `IEJEC_MODE`=3.

(9) Particle speed parameters V_0, γ, and κ in $V = V_0 \beta^\gamma r_h^\kappa \cos(z)^x$, where V_0 is expressed in km s^{-1} and r_h is in au.

(10) Exponent x in the above expression, $\cos(z)^x$, normally x=0.5

(11) Apparent magnitude of the sun in the selected band

(12) Julian date when activity starts

(13) Julian date of observation

(14) Output image (fits file) dimensions ($NX \times NY$). Nucleus is at fixed position given by coordinates $(NX/2, NY/2)$.

(15) Image scale in arcsec px^{-1}

(16) Monte Carlo number of time bins

(17) Monte Carlo number of particle size bins

(18) Monte Carlo number of directional events

(19) Aperture photometry, giving Afρ and magnitude: The first parameter indicates whether the aperture used in given in arcsec (set the parameter to 1) or km (set it to 2). The second parameter is the radius of the aperture (in arcsec or km, depending on the value of the first parameter).

(20) Nucleus radius in meters.

(21) Nucleus geometric albedo and linear phase coefficient.

(22) Spatial convolution of the output image: 1=Yes/0=No, and FWHM of 'seeing' in pixel units.

(23) If this parameter equals 1, then the code plots a background stellar field, useful for calibration, plate scale check, and stellar extinction calculations. In the present version, the stellar sources refer to R-Cousins band. For other photometric bands, or star catalog, the user should provide the corresponding magnitudes in those specific filters, and the code will have to be modified. The second input is the limiting magnitude for the plotted stars in the image.

(24) Set this parameter to 1 if you want the output file of particle coordinates (N,M,L), nm.dat.

(25) Set this parameter to 1 if you want active graphics output on your screen of (N,M) coordinates of the particles ejected as a function of time, and then the percent number of total particles ejected in the Monte Carlo procedure. Be aware that the total Monte Carlo events should be usually very large, so it would take a long CPU time if a large percentage is set.

(B) File `dmdt_vel_power_rmin_rmax.dat`

This file contains the decimal log of the dust mass loss rate LOG10(dm/dt) profile (column 2), an additional velocity factor (column 3), the power index of the size distribution function (column 4), the minimum particle radius (column 5), and the maximum particle radius (column 6), all of them as a function of time to perihelion in days (column 1), negative before perihelion and positive after perihelion. Make sure to define those quantities in the time interval that includes the activity start and the observation times. External codes can be used by the user to generate this file using analytical expressions. The velocity factor is set to unity in this example file, where the time dependence of speeds is being determined by the heliocentric distance dependence.

Note: Make sure that the perihelion time of the orbit being analyzed corresponds to the date shown in the JPL ephemeris. For instance, let us assume that you want to obtain synthetic images of comet 7P/Pons-Winnecke during the year 2021. In that orbit, comet's perihelion was on May, 2021. However, if you select the latest orbit from the 7P orbits available in the JPL-Horizons (as to the date at the beginning of this document), it corresponds to the year 2015 orbit (perihelion on January 30, 2015), so that the code will not run properly. The updated perihelion date is calculated by adding the orbital period to the previous perihelion passage date, and should be checked.

Code outputs (files):

In file '`nm.dat`', the (N, M, L) coordinates (columns 1-3) (see the Finson-Probstein theory) of each launched particle, along with their radius (column 4), the time index (column 5), and the ejection time (column 6). One has to be careful with this option, as it generates a huge file if the code is run with typical parameters (e.g., $\gtrsim 10^7$ Monte Carlo events).

File '`starpos.dat`',contains the x,y positions of the stars in the image, along with their magnitudes and fluxes (columns 1, 2, 3, and 4, respectively)

File '`subsolar.dat`' contains the subsolar latitude of the nucleus as a function of time. This file is generated when `IEJEC_MODE`=3 only. Column 1 is time to perihelion in days, column 2 is the heliocentric distance in au, column 3 is the true anomaly in radians, and column 4 is the subsolar latitude in degrees.

File '`dustlossrate.dat`' contains the dust loss rate (column 3) as a function of time in days to perihelion (column 1) and heliocentric distance in au (column 2).

File `fort.99`: For the aperture size selected in input file '`TAIL_INPUTS.dat`', line 19, this file contains the time to perihelion in days (column 1), Afrho, Afrho reduced to 0 deg phase (both in metres, columns 2 and 3), apparent magnitude of target (column 3).

In addition, a few *.txt workspace files are generated, but these do not need to be examined. These might be used for internal checks.

Code outputs (images, in FITS format):

`tail_sdu.fits`: target image in solar disk intensity units

`tail_mag.fits`: target image in mag arcsec^{-2}

`OPT_DEPTH.fits`: image of tail optical depth at each pixel

All FITS images have some information of the input parameters in the header.

Compilation of the FORTRAN versions

The user has to install the corresponding `FITSIO` and `PGPLOT` libraries in his/her system in order to generate the output `FITS` images and the graphics output, respectively. If a graphics output is not required, it is enough to comment all the lines containing a call to `PGPLOT` routines (search for variable `IGRAPHO` in the code in order to find them).

For the serial version, the code is compiled in my Linux station using the command:

```
> gfortran COMTAILS.for heliorbit.for -o COMTAILS.exe -L/usr/local/pgplot64 -lpgplot
-L/usr/X11R6/lib64 -LX11 -L. -lcfitsio -lm
```

And for the MPI version:

```
> mpif77 -fallow-argument-mismatch -o COMTAILS_PAR.exe COMTAILS_PAR.for heliorbit.for
-L/usr/local/pgplot64 -lpgplot L/usr/X11R6/lib64 -LX11 -L. -lcfitsio -lm
```

Example execution

After successful compilation, copy the file `C2020F3_TAIL_INPUTS.dat` to `TAIL_INPUTS.dat` and run the code. The output images `tail_sdu.fits` and `tail_mag.fits` must match the files provided in the tarball `C2020F3_sdu.fits`, and `C2020F3_mag.fits`, respectively. Those files are for a simulated image of comet C/2020 F3 (NEOWISE) on July 17th, 2020. Similarly, your outputs should match those provided by running the code for the file `C2020F3_TAIL_INPUTS_stars_convol.dat` (by copying first this file to `TAIL_INPUTS.dat`). Running the code for the serial version is simple:

```
> ./COMTAILS.exe
```

and for the parallel MPI version:

```
> mpirun -np 8 ./COMTAILS_PAR.exe
```

The output images should match those provided as `C2020F3_sdu_stars.fits`, and `C2020F3_mag_stars.fits`. Those correspond to the same simulation as before but including the GAIA stars up to the 17th magnitude, with the final images being convolved with a PSF of 3 pixels FWHM.

NOTE: None of the inputs provided are intended to simulate an actual observed image.