DustEM User's Guide

October 12, 2010

DustEM is a numerical tool in fortran 95 jointly developed by IAS and CESR to compute the extinction and the emission of interstellar dust grains heated by photons. The dust emission is calculated in the optically thin limit (no radiative transfer) and the default spectral range is 40 to 10^8 nm. The code has been designed so that dust properties can easily be changed and mixed and to allow for the inclusion of new grain physics (Compiègne *et al.* 2010). The grain temperature distribution dP/dT is derived with the formalism of Désert *et al.* (1986) and assuming that the grain cooling is fully continuous. To correctly describe the long wavelength emission of dust, the initial algorithm has been revised to cover the low temperature part of dP/dT. The current stable version of `dustem` is available at `http://www.ias.u-psud.fr/DUSTEM`. The fortran files are in `/src`, data files in `/oprop, /hcap, /data` and output files are in `/out`. All data files have headers (where each line begins with #) that document their content. IDL routines useful to generate or visualize DustEM files can be found in `/idl_dustem`[1].
After a few words on how to set up DustEM, we detail below the way to control DustEM as well as the files it needs or produces.

1 Getting started

After downloading the tar ball from the DustEM website:

1. unpack the archive:

   ```
   tar xvjf dustem.tar.gz
   ```

   this will generate the `dustem` dir which should contain the following subdirs: `data`, `idl_dustem`, `hcap`, `oprop`, `src` and `out`.

2. Go in `src` and edit the file `DM_constants.f90`. Then define the string `data_path` as the absolute path to the DustEM dir you just created. If you are running DustEM coupled to the Meudon PDR code, you must also define the string `dir_PDR` as the absolute path to `/dustem/out`.

3. Edit `Makefile` and set up the standard commands corresponding to your fortran compiler (`gfortran`, `g95` and `ifc` have been tested[2]), *e.g.*:

   ```
   FC = gfortran
   FFLAGS = -O2 -fno-second-underscore
   ```

4. Type `make` to compile possibly preceded by `make clean`, if a former attempt went wrong. If succesful compilation will end with `That's it !`. The executable is called `dustem`

5. Running: `./src/dustem`.

6. Check your run, *e.g.*, using `show_dustem` in `idl_dustem`.

[1] Documentation on the routine `rname` is usually obtained by typing `rname` (`tt=rname()`) in case of a procedure (function) respectively.

[2] `ifc` is about twice as fast as the others.

2 Input files

The main parameters and keywords for the dust model to be computed are read from the control input file `GRAIN.DAT` (in `/data`). This control file must be located in the `/data` directory. Each *dust population* is defined by the *grain type* and the *size distribution*. We note `tt` the string containing the name of a grain type. For each grain type of mass density ρ, the radius a is defined as that of an equivalent sphere of same mass $m = \rho\, 4\pi a^3/3$. `DustEM` can handle an arbitrary number of dust populations.

2.1 Defining the dust model

We show below a typical example of the `GRAIN.DAT` (as in Compiègne *et al.* 2010).

```
# DUSTEM: definition of grain populations
#
# run keywords
# G0 scaling factor for radiation field
# grain type, nsize, type keywords, Mdust/MH, rho, amin, amax, alpha/a0 [, at, ac, gamma (ED)] [, au, zeta, eta (CV)]
# cgs units
#
 sdist
 1.00
 PAH0        10 logn-mix    7.80E-04 2.24E+00 3.50E-08 1.20E-07  6.40E-08   1.00E-01
 PAH1        10 logn-mix    7.80E-04 2.24E+00 3.50E-08 1.20E-07  6.40E-08   1.00E-01
 amCBEx      15 logn        1.65E-04 1.81E+00 6.00E-08 2.00E-06  2.00E-07   3.50E-01
 amCBEx      25 plaw-ed     1.45E-03 1.81E+00 4.00E-07 2.00E-04 -2.80E+00 1.50E-05 1.50E-05 2.00E+00
 aSil        25 plaw-ed     7.80E-03 3.50E+00 4.00E-07 2.00E-04 -3.40E+00 2.00E-05 2.00E-05 2.00E+00
```

After some doc lines come the run keywords, then a scaling factor on the intensity of the radiation field. The next lines define the dust populations: grain type, number of sizes (`nsize`), type keywords, the mass ratio M_i/M_H, the grain mass density, a_{min} and a_{max} the minimum (maximum) sizes in cm and other parameters for the size distribution.
Keywords are important to control `DustEM`. They can be written in uppercase or lowercase characters. The *run keywords* control the way `DustEM` is executed and the output files it produces. They are

- **quiet:** sets the verbose mode off at runtime,
- **sdist:** for each grain size and type, writes the size distribution in the file `/out/SDIST.DAT`,
- **res_a:** writes output for each grain size (see sec. 3)
- **temp:** for each grain size and type, writes the temperature distribution in the file `/out/TEMP.DAT`,
- **pdr:** writes a summary of `DustEM` output in `/out/EXTINCTION_DUSTEM.DAT`. To be set when `DustEM` is used coupled to the Meudon PDR code.

If multiple run keywords are used, they must be separated by blanks. The *type keywords* define the size distribution and the physical processes included in `DustEM` for each grain type `tt`. Type keywords can also be multiple, then separated by a dash.

2.2 Template dust models

Several dust models are widely used and constitute *templates*. We give below a list of templates along with file names containing model parameters:

- **GRAIN_DBP90.DAT:** for the model of Désert *et al.* (1990),

- **GRAIN_DL01.DAT:** for the model of Li & Draine (2001),
- **GRAIN_DL07.DAT:** for the model of Draine & Li (2007) and
- **GRAIN_MC10:** for the model of Compiègne *et al.* (2010).

2.3 The size distribution

Unless the type keyword `size` is set, the size grid has `nsize` points and is defined over the interval $[a_{min}, a_{max}]$ with a constant log step. The shape of the size distribution dn/da (the number of grains whose radius is included in $[a,\, a+da]$) is then defined by the following type keywords:

- **logn:** defines a log-normal size distribution with $dn/d\log a \propto \exp(-\log(a/a_0)^2)/\sigma$. The centroid a_0 (in cm) and then width σ are given after a_{max} in `GRAIN.DAT`.
- **plaw:** defines a power-law size distribution $dn/da \propto a^{\alpha}$. The index α is given after a_{max} in `GRAIN.DAT`.
- **ed:** applies an exponential decay to a power law size distribution multiplying dn/da by the function

$$D(a) = \begin{array}{ll} 1 & a \le a_t \\ \exp(-\left[(a-a_t)/a_c\right]^{\gamma}) & a > a_t. \end{array} \qquad (1)$$

  The parameters a_t, a_c and γ are given in this order after α in `GRAIN.DAT`.
- **cv:** applies a curvature term to a power law size distribution multiplying dn/da by the function

$$C(a) = \left[\, 1 + |\zeta|\,(a/a_u)^{\eta}\,\right]^{\mathrm{sgn}(\zeta)} \qquad (2)$$

  If `ed` is set with `cv` the parameters a_u, ζ and η are given in this order after γ in `GRAIN.DAT`. Otherwise they are given after α.

The keywords `ed` and `cv` allow to use a size distribution similar to that of Weingartner & Draine (2001).

When the type keyword **size** is set the size distribution and the size grid are read from the file `/data/SIZE_tt.DAT` which must have the following format (not including the #-lines):

```
# Size distribution of grain species
#
# shape (t-matrix code, 0:spheres, -1:spheroids) eps (b/a, eps>1:oblate, eps<1: prolate)
# Nbr of bulk materials
# Name of bulks
# Bulk densities in g/cm3
# Mass fractions for each bulk
# Nbr of size bins
# [ a (cm), dloga, a^4*dn/da, rho_eff, fv ]
# fv: volume fraction of bulks, rho_eff: volume mean density
#
0 1.00E+00
1
Gra
2.2500E+00
1.0000E+00
```

46

8.0876E-08 2.0939E-01 7.1858E-02 2.2500E+00 1.0000E+00

....

This file can be generated with the IDL routine `create_vdist.pro` or taken from the `dustev` ouptut (Guillet *et al.* 2011, in preparation). Size dependent composite grains can be handled this way.

2.4 Grain physics

Physical processes affecting the properties of grains can be plugged in `DustEM` with type keywords often associated with data files where parameters are passed. Below is the list of type keyword for the processes included in dustem:

- **mix:** performs a size dependent mixing of the extinction and emission of several grain types. Requires a file `/data/MIX_tt.DAT` for each type which contains a single column with mixing values f_{mix} at each point of the size grid. For each size, the mixing must verify $\sum_{tt} f_{mix}(tt) = 1$. In the example `GRAIN.DAT` shown above ionized and neutral PAHs are mixed size by size in the model. In a coarser way, species can be mixed with the mass fraction.

- **beta:** applies a correction to Q_{abs} to introduce a temperature dependence, namely:

$$Q_{abs}(a,\nu) = Q_0(a,\nu)\ (\nu/\nu_t)^{\delta(T)\,H(\nu_t/\nu)} \tag{3}$$

  where Q_0 is the grain emissivity without the temperature dependence, $\nu_t = c/\lambda_t$ is the frequency threshold and $\delta(T) = \beta(T) - \beta_0$ is the index correction with respect to β_0 some reference value. The threshold function in frequency is $H(x) = 0.5\,(1+4\tanh(x)/s)$ where $x = \log(\lambda/\lambda_t)$ and s controls the stiffness of the transition (the width Δx is proportional to s) from 0 to 1: for instance for $s = 1$ $H(x)$ rises from 0 to 1 for x between 0.5 to 1.5 and $\Delta x = 1$. Parameters for this correction are in `/data/BETA_tt.DAT`:

  # DUSTEM: parameters for BETA(T) behaviour of Qabs

  # first DBETA correction to standard index beta0

  # then lambda threshold to apply BETA(T)

  #

  # beta0 a g bmax

  # lthresh(microns) lstiff (dlambda=lthresh/lstiff)

  # nbetav (nr of beta values if 0 uses formula)

  # tbeta betav (if nbetav ¿= 0)

  #

  2.11E+00 1.15E+01 -6.60E-01 5.00E+00

  3.00E+01 1.00E+00

  13

  1.00E+00 1.55E+00

  ...

  where $lthresh$ and $lstiff$ stand for the above λ_t and s. If $nbetav = 0$ then $\beta(T) = a\,T^g$ (Désert *et al.* 2008) and we take $\beta = \min(\beta, bmax)$. If $nbetav > 0$ the $\beta(T)$-values must be given below $nbetav$ in two columns T and β.

2.5 Grain data

For a given type of grain `tt`, dust grain properties are provided depending on the size a in data files, namely: `Q_tt.DAT` files for the absorption and scattering efficiencies, $Q(a,\lambda)$, and `C_tt.DAT` files for the heat capacity per unit volume, $C(a,T)$. The Q and C values are provided over the

broadest possible size range (usually 0.3 to 10^4 nm). A grain type can be used once these Q and C-files exist in the `DustEM` data file set, in the optical properties `/oprop` and heat capacity `/hcap` directories respectively. Requested sizes have to fit into those of the above files (i.e. extrapolation is not allowed). When `DustEM` is coupled to radiative transfer codes, g-factors[3] are required at all wavelengths and sizes: they are given in n in `/oprop/G_tt.DAT`.
We briefly outline below the format of these files.

The Q-files are as:

```
# QABS and QSCA to be used by DUSTEM
#
# nsize (number of sampled sizes in this file)
# sizes (microns)
50
3.0000E-04 3.7104E-04 4.5891E-04 5.6759E-04 7.0200E-04 8.6824E-04 1.0738E-03 1.3281E-03 ...
#
#### QABS ####
1.3461E-03 1.6642E-03 2.0571E-03 2.5419E-03 3.1398E-03 3.8762E-03 4.7819E-03 5.8952E-03 ...
...
#### QSCA ####
5.0179E-07 1.1650E-06 2.6935E-06 6.1884E-06 1.4083E-05 3.1594E-05 6.9380E-05 1.4766E-04 ...
```

G-files are as Q-files with the QABS field replaced by G and no QSCA field. The Q and G-files are all generated with a common wavelength grid which can be found in `/oprop/LAMBDA.DAT`.

For the C-files:

```
# DUSTEM heat capacity
#
# nr of sizes ns
# sizes (microns)
# nr of T-values
# log T(K) log C_1...log C_ns (erg/K/cm3)
50
3.0000E-04 3.7104E-04 4.5891E-04 5.6759E-04 7.0200E-04 8.6824E-04 1.0738E-03 1.3281E-03 ...
30
-1.0000E+00 4.8983E-01 4.8983E-01 4.8983E-01 4.8983E-01 4.8983E-01 4.8983E-01 4.8983E-01 ...
```

The same temperature grid (30 points from 0.1 to 5,000 K with a constant log step) is used for all C-files and is in `TEMP.DAT`.

The grain data can be batch generated with the `dustprop` IDL package (not included in `idl_dustem`). We list below the grain types for which Q and C files are available in the current `DustEM` distribution:

- **PAH0, PAH1:** as defined in Compiègne *et al.* (2010) for neutral and singly charged PAHs respectively,
- **Gra:** graphite grains. The Q-values have been generated with the Mie method. The refractive index and heat capacity are as in Li & Draine (2001).
- **amC-BEx:** amorphous carbon grains. The Q-values have been generated with the Mie method. The refractive index is from Zubko *et al.* (2004) and the Q_{abs} have been extrap-

[3]The anisotropy factor for scattering defined as $< \cos\theta >$ where θ is the scattering angle.

olated above 0.9 mm (see Compiègne *et al.* 2010). The heat capacity is the same as that of graphite.

- **aSil:** silicate (olivine type) grains. The Q-values have been generated with the Mie method. The refractive index and heat capacity are as in Li & Draine (2001).

Data files for the Désert *et al.* (1990) model are also available corresponding to the grain types **PAH0_DBP90, PAH1_DBP90, VSG_DBP90** and **BG_DBP90**. PAHs properties defined in Li & Draine (2001) and Draine & Li (2007) are also available as grain types **PAH_DL01** and **PAH_DL07**.

2.6 Astrophysical data

The radiation field used by `DustEM` is in `/data/ISRF.DAT`. This file contains 2 columns, the wavelength (microns) and the flux $4\pi I_\nu$ in erg/s/cm^2/Hz. The wavelength range must be included in that of the Q-file. It can be generated with the `create_rfield.pro` routine in `/idl_dustem`. The current distribution includes `ISRF_MATHIS.DAT`, the standard radiation field of Mathis *et al.* (1983) and `ISRF_CMB.DAT` where the CMB contribution has been added to the Mathis field.

3 Output files

Output files produced by `DustEM` are stored in `/out`. They give essentially the emission and extinction of dust grains. Each file begins with documentation lines (first character is #). Any `DustEM` run will produce the 2 following files:

- **SED.RES:** contains the emission per proton of each dust population $4\pi\nu I_\nu/N_H$ in erg/s/H as a function of wavelength (N_H is the proton column density). The last column is the total of all previous columns.
- **EXT.RES:** contains the extinction cross-section of the grain populations per gram of dust type.

If the run keywords `temp` or `sdist` are used, `TEMP.DAT` or `SDIST.RES` will be written containing respectively the temperature or the size distribution of the dust populations.

Using the run keyword `res_a` will generate the file `SED_A.RES` which contains the emission per grain in size bin $[a, a+da]$ or $4\pi\nu I_\nu/N_d(a)$ in erg/s/grain ($N_d(a)$ is the column density of grains of size a).