

Documentation for `rac2dm`

Fujun Du
fujun.du@gmail.com

October 15, 2018

Contents

1	Guideline	1
2	Description of modules	2
2.1	Radiative transfer and bulk radiation field solution: <code>rtrf</code>	2
2.1.1	Overall configuration	2
2.1.2	Specification of matter distribution: <code>struct_file</code>	2
2.1.3	Specification of optical properties: <code>opti_files</code>	2
2.1.4	Specification of radiation sources: <code>rad_source_files</code>	3
2.1.5	Output	4
2.2	Disk structure adjustment: <code>structadj</code>	4
2.2.1	Overall configuration	4
2.2.2	Output	4
2.3	Thermochemical calculation: <code>thermochem</code>	4
2.3.1	Overall configuration	4
2.3.2	Output	5
2.4	Make images: <code>makeimage</code>	5
2.4.1	Overall configuration	5
2.4.2	Output	6

1 Guideline

The `rac2dm` package is designed to model a static disk structure and its thermochemistry. It is derived from the `rac2d` package. The letter “m” in the name stands for “modularized”. Different modules of this package are designed to be as independent to each other as possible. For the ease of implementation, the interface to each module is implemented with files. File IO is slow, but calculations usually take more time, hence IO should not be a bottleneck. Using script languages such as `Python`, different modules can be chained and iterated to achieve a specific goal.

2 Description of modules

2.1 Radiative transfer and bulk radiation field solution: rtrf

2.1.1 Overall configuration

```
1 &conf_nml_rt
2   conf_rt%struct_file = "struct.dat"
3   conf_rt%opti_files = "sil.dat"  "gra.dat"
4   conf_rt%rad_source_files = "s1.dat"  "s2.dat"
5   conf_rt%star_mass_Msun = 0.6D0
6   conf_rt%nphoton = 1000000
7   conf_rt%nmax_encounter = 1999999999
8   conf_rt%randnumber_save_file = "rd_saved.dat"
9   conf_rt%escaped_photons_file = "escaped_ph.dat"
10  conf_rt%collect_photon   = .true.
11  conf_rt%collect_lam_min_AA = 1D0
12  conf_rt%collect_lam_max_AA = 1D8
13  conf_rt%collect_nmu      = 3
14  conf_rt%collect_ang_mins_deg = 0D0 4D0 80D0
15  conf_rt%collect_ang_maxs_deg = 3D0 10D0 90D0
16  conf_rt%nlen_lut         = 4096
17  conf_rt%Tdust_min_K      = 1D0
18  conf_rt%Tdust_max_K      = 5D3
19  conf_rt%dist_to_Earth_pc = 51D0
20 /
```

Notes:

- Each type of dust grain has a separate entry in the `opti_files` list.

2.1.2 Specification of matter distribution: `struct_file`

```
1 format ! Default to 1
2 number_of_species  number_of_data_points
3 r1  z1  n11  n21  n31  ...
4 r2  z2  n12  n22  n32  ...
5 ...
```

Here n_{ij} means the density of the i th dust species at the j th location.

2.1.3 Specification of optical properties: `opti_files`

```
1 number_of_data_points
2 angstrom1  Q_abs1  Q_sca1  g1
```

```

3 angstrom2  Q_abs2  Q_sca2  g2
4 ...

```

2.1.4 Specification of radiation sources: rad_source_files

Multiple radiation sources are allowed. Each source can have a different type. At present the following types are supported:

- **point:** point source
- **sphere:** photons are emitted from the surface of a sphere
- **isotropic:** photons are emitted from an isotropic background
- **directional:** photons are emitted from a single direction

The following are example radiation source files:

```

1 point
2 x_au  y_au  z_au
3 number_of_data_points_of_spectrum  total_luminosity_cgs
4 angstrom1  F1_cgs
5 angstrom2  F2_cgs
6 ...

```

```

1 sphere
2 x_cen_au  y_cen_au  z_cen_au  r_sphere_Rsun
3 number_of_data_points_of_spectrum  total_luminosity_cgs
4 angstrom1  F1_cgs
5 angstrom2  F2_cgs
6 ...

```

```

1 isotropic
2 ! Empty line
3 number_of_data_points_of_spectrum  energy_density_cgs
4 angstrom1  F1_cgs
5 angstrom2  F2_cgs
6 ...

```

```

1 directional
2 v_x  v_y  v_z
3 number_of_data_points_of_spectrum  flux_density_cgs
4 angstrom1  F1_cgs
5 angstrom2  F2_cgs
6 ...

```

2.1.5 Output

- Temperature distribution
- Internal radiation field
- Escaped radiation field

2.2 Disk structure adjustment: structadj

2.2.1 Overall configuration

```
1 &conf_nml_struct_adj
2   conf_struct_adj%density_file = "dens.dat"
3   conf_struct_adj%density_file_new = "dens_new.dat"
4   conf_struct_adj%temperature_file = "temperature.dat"
5 /
```

2.2.2 Output

- New density distribution

2.3 Thermochemical calculation: thermochem

Here we assume the thermochemical calculation is only done for a single cell. To calculate for the whole disk just call the module in a loop or use the `map` function in Python.

2.3.1 Overall configuration

```
1 &conf_nml_thermochem
2   conf_thermochem%network_file = "rate12.dat"
3   conf_thermochem%init_file = "init.dat"
4   conf_thermochem%enthalpy_file = "enthalpy.dat"
5   conf_thermochem%Tgas = 20D0
6   conf_thermochem%Tdust = 20D0
7   conf_thermochem%sigdust_cm2 = 20D0
8   conf_thermochem%G0 = 1D0
9   conf_thermochem%Xray = 0D0
10  conf_thermochem%zeta_cosmicray_H2 = 1.36D-17
11  conf_thermochem%PAH_abundance = 1.6D-9
12  conf_thermochem%Av = 10D0
13  conf_thermochem%filename_CII = ""
14  conf_thermochem%filename_OI = ""
15  conf_thermochem%filename_NII = ""
16  conf_thermochem%filename_SiIII = ""
```

```

17 conf_thermochem%filename_FeII = ""
18 conf_thermochem%dt_first_step = 1D-8
19 conf_thermochem%t_max = 1D6
20 conf_thermochem%ratio_tstep = 1.1D0
21 conf_thermochem%max_runtime_allowed = 60D0
22 conf_thermochem%RTOL = 1D-6
23 conf_thermochem%ATOL = 1D-30
24 /

```

2.3.2 Output

- For a single location, output the gas temperature and chemical abundances as a function of time.
- Optional output

2.4 Make images: makeimage

2.4.1 Overall configuration

```

1 &conf_nml_image
2 conf_img%molecule_density_file = "mol_dens.dat"
3 conf_img%Tgas_file = "Tgas.dat"
4 conf_img%continuum_file = "cont.dat"
5 conf_img%transition_file = "12C16O_H2.dat"
6 conf_img%database_name = "lamda"
7 conf_img%molecule_name = "CO"
8 conf_img%useLTE = .true.
9 conf_img%image_type = "cube"
10 conf_img%out_dir = "images"
11 conf_img%maxx = 200D0
12 conf_img%maxy = 200D0
13 conf_img%nx = 401
14 conf_img%ny = 401
15 conf_img%nf = 201
16 conf_img%nfreq_window = 3
17 conf_img%freq_mins = 0.9D11 3.4D11 6.90D11
18 conf_img%freq_maxs = 3.0D11 3.6D11 6.92D11
19 conf_img%E_min = 0D0
20 conf_img%E_max = 3D3
21 conf_img%n_theta = 1
22 conf_img%view_thetas = 7D0
23 conf_img%dist_to_Earth_pc = 51D0
24 conf_img%vel_width_kms = 30D0
25 /

```

2.4.2 Output

- 3D data cube, 2D image, or 1D spectrum