RadmcGrid doc

Amira Bouikni

July 2025

Contents

1 Introduction

This package provides a Python-based interface for efficiently simulating protoplanetary disks using the radiative transfer code RADMC-3D. It is designed to streamline the process of setting up, executing, and visualizing a variety of disk structures commonly observed in young stellar systems. By leveraging parallel computing, the package significantly reduces simulation run times, making it suitable for generating large grids of disk models within practical time frames.

The toolkit supports the simulation of different protoplanetary disk morphologies, including:

- **Continuous disks** — Smooth, uninterrupted disk structures.
- **Disks with a gap** — Systems with a radial discontinuity separating two distinct dust regions.
- **Disks with a cavity** — Disks featuring a cleared central region surrounded by an outer dusty ring.

This package is particularly useful for performing parameter space explorations, generating synthetic observations, and studying the impact of disk substructures such as gaps and cavities on the observable properties of protoplanetary disks. It also allows for flexible control over the dust composition, disk geometry, and radiative transfer parameters, providing a versatile platform for disk modeling studies.

2 Installation Requirements

2.1 Python Environment

This package is developed for Python 3 and is compatible with Python versions 3.7 and above.

2.2 Required Software

- **RADMC-3D:** The radiative transfer simulations are performed using the RADMC-3D code, which must be installed separately. The executable `radmc3d` must be accessible from the system's environment PATH in order for the simulations to run successfully. For installation instructions, refer to the official RADMC-3D website: RADMC-3D.
- **Python Packages:** The following Python libraries are required to run this package:
  - `radmc3dPy` – The official Python interface for preparing and managing RADMC-3D input and output files.
  - `numpy` – Essential for numerical computations and array management.

- `matplotlib` – Used for creating plots and visualizing simulation outputs.
- `astropy` – Required for handling FITS files and astronomical data formats.
- `tqdm` – Provides progress bars for tracking simulation execution.
- `seaborn` – Optional, but used for enhanced data visualization and plotting aesthetics.

2.3 Python Package Installation

To install the required Python libraries (excluding `radmc3dPy`), run the following commands in your terminal:

```
pip install numpy
pip install matplotlib
pip install astropy
pip install tqdm
pip install seaborn
```

Alternatively, you can install all packages at once:

```
pip install numpy matplotlib astropy tqdm seaborn
```

2.4 Important Notes

- **Multiprocessing:** This package uses the native `multiprocessing` module in Python. No installation is required, but ensure your system supports parallel execution.
- **RADMC-3D Setup:** Ensure the RADMC-3D executable is correctly compiled and accessible for running simulations. The Python interface must also be correctly located and importable as shown above.
- **Platform compatibility:** The package has been developed and tested on Linux-based systems. It is expected to work on MacOS with minimal adjustments. Windows users may need to handle additional path and system-specific issues.

3 Disk Model Descriptions and Function Definitions

This package provides three distinct protoplanetary disk geometries that can be simulated using the included functions. Each disk type represents different stages or structural features observed in young stellar objects. The physical differences between these disk types directly influence the disk's density distribution, temperature structure, and observable spectral energy distributions (SEDs).

3.1 Continuous Disk — `nogap()` Function

A continuous disk is a simple, fully connected disk without any internal structural breaks. It represents an early-stage protoplanetary disk where the material is uniformly distributed from the inner radius out to the outer edge.

Physical Motivation: Continuous disks are commonly used to model early-stage protoplanetary systems where the disk material is smoothly distributed and has not yet developed structural features such as gaps or cavities.

Function Definition:

```
nogap(sim_name, rin, rdisk, mdisk, output_dir, core_range,
      tstar, rstar, inc, npix, sizeau, nlmin, nlmax, nlam,
      Cdelt3, dpc, composition_name, template_dir, optional_params=None)
```

Key Features:

- supports different values of disk masses (`mdisk`) as well as disk inclinations (`inc`).
- Generates dust density, optical depth, temperature, and SEDs for a continuous structure.
- Supports optional customization of RADMC-3D parameters.

Simulations names :

The names of the simulations for the continuous disk model, are saved as follows:

```
Model_nogap_rin_{rin}_rdisk_{rdisk}_mdisk_{mdisk}_inc_{inc}
```

where :

- `rin`: inner radius of the disk.
- `rdisk`: outer radius of the disk.
- `mdisk`: mass of the disk.
- `inc`: inclination of the disk.

inner most radius:

The inner edge of the dusty disk, in all the simulations produced by the grid, is set as the dust sublimation radius, R_{sub}, where dust grains reach temperatures high enough to sublimate and thus cannot survive closer to the star. Under the assumption of grey dust opacity and radiative equilibrium with stellar irradiation, the sublimation radius is given by:

$$R_{\mathrm{sub}} = \left(\frac{L_\star}{4\pi\sigma T_{\mathrm{sub}}^4} \right)^{1/2} = R_\star \left(\frac{T_{\mathrm{eff}}}{T_{\mathrm{sub}}} \right)^2 \tag{1}$$

where $L_\star$ is the stellar luminosity, T_{sub} the dust sublimation temperature, and σ the Stefan-Boltzmann constant. Assuming a fixed sublimation temperature $T_{\rm sub} = 1500$K.

Usage exemple:

Simulate a continuous disk in the L band:

```
nogap("Model_nogap_rin_0.1_rdisk_100_mdisk_1e-4_inc_30", 0.1, 100, 1e-4,
"output", "0-3",4000, 2.0, 30, 200, 60, 3, 4, 11, 0.1, 145,
      "full_astrosil_amira", "/data/dust_templates")
```

3.2 Gapped Disk — `gap()` Function

A gapped disk is characterized by a depletion region, where the dust density is significantly reduced, separating the inner and outer disks. The gap can result from a variety of physical processes, such as planet formation, which carves out material along its orbit.

Physical Motivation: Gapped disks are commonly observed in transition disks and systems where embedded planets may be actively clearing material. These gaps typically show lower dust densities, but may still contain gas and some residual dust. In this package, the depletion region is modeled explicitly: the dust surface density inside the gap is reduced by several orders of magnitude to simulate this dust depletion.

This approach allows the user to set independent properties for the inner and outer disks, such as:

- **Different disk masses** (`mdisk1`, `mdisk2`)
- **Distinct density profiles** (power-law indices `plsig1` and `plh`)
- **Independent outer radii** for each disk segment

The gap region itself is empty and is treated as a dust-depleted zone to reflect the observed physical scenarios.

Function Definition:

```
gap(sim_name, rin, rdisk1, rout, rdisk2, mdisk1, mdisk2, output_dir,
    core_range, tstar, rstar, inc, npix, sizeau, nlmin, nlmax, nlam, Cdelt3, dpc,
    composition_name, template_dir, optional_params_inner=None,
    optional_params_outer=None)
```

Key Features:

- Supports different parameter sets for the inner and outer disks, as well as inner disk mass (`mdisk1` and disk inclinations (`inc`)).
- Models the gap as a depletion region with reduced dust surface density.
- Enables fine control of the geometry and dust distribution on both sides of the gap.

- Supports independent power-law density profiles for each disk section.

Usage Example:

```
gap("Model_gap_rin_0.1_rdisk1_30_rout_50_mdisk1_1e-8_inc_30", 0.1, 30, 50, 100,
1e-8, 1e-4, "output", "0-3", 4000, 2.0, 30, 200, 60, 3, 4, 11, 0.1, 145,
    "full_astrosil_amira", "/data/dust_templates",
    optional_params_inner={'plsig1': '-1', 'plh': '0'},
    optional_params_outer={'plsig1': '-0.5', 'plh': '0.1'})
```

3.3 Cavity Disk — `cavity()` Function

A cavity disk represents a disk with a cleared inner region and a single outer dust ring. These structures are observed in evolved protoplanetary disks where photoevaporation, planet formation, or disk winds may have removed most of the material from the inner disk.

Physical Motivation: Cavity disks are often used to model transition disks that exhibit central clearings in millimeter-wavelength observations. In this setup, the inner disk is entirely removed, and the remaining dust forms an outer ring starting at a larger radius.

Function Definition:

```
cavity(sim_name, rin, rout, rdisk, mdisk, output_dir, core_range,
       tstar, rstar, inc, npix, sizeau, nlmin, nlmax, nlam, Cdelt3, dpc,
       composition_name, template_dir, optional_params=None)
```

Key Features:

- Inner cavity is fully evacuated of dust.
- Outer disk parameters are fully configurable.
- supports different values of disk masses (`mdisk`) as well as disk inclinations (`inc`).
- Suitable for modeling disks with significant dust removal in the inner region.

Usage Example:

```
cavity("Model_cavity_rout_10_rdisk_100_mdisk_1e-6_inc_30", 0.1, 10, 100,1e-6,"output",
"0-3", 4000, 2.0, 30, 200, 60, 3, 4, 11, 0.1, 145,"full_astrosil_amira",
"/data/dust_templates")
```

important: In the cavity model, the function still accepts `rin` as an input parameter, even though the dusty disk does not begin at that radius. This is intentional: `rin` defines the starting radius of the computational grid in RADMC, which may differ from the actual inner edge of the dusty disk. This allows for the inclusion of an inner cavity—an entirely depleted region within the grid—between `rin` and the onset of the dusty disk.

4 Running the models

All disk models require access to dust opacity files. These are provided with the package in the `opacities/` directory. Each opacity set is stored in the `dustkappa.inp` format, generated using **Optool**[1] and designed to be directly readable by **RADMC-3D**. To ensure portability across machines, you should always pass this directory to the `template_dir` argument using a relative path rather than an absolute one.

In the implementation (`RadmcGrid.py`), the central `run()` function is responsible for executing the simulations. Based on the model type specified by the user (`model = 0` for continuous, `model = 1` for gapped, `model = 2` for cavity), run() automatically calls the corresponding lower-level function (`nogap()`, `gap()`, or `cavity()`) and manages parameters, file handling, and multiprocessing.

```python
import radmcgrid as grid
from pathlib import Path

# Basic shared parameters
template_dir = Path(__file__).parent.parent / "opacities"
tstar_rstar_list = [('[4000.0]', '[2.0*rs]'), ('[5000.0]', '[1.5*rs]')]
rout = 10
mdisk1 = 1e-3
mdisk2 = 1e-4
rdisk2 = 100
Inc = [30, 60]
npix = 200
sizeau = 60
nlmin = 3
nlmax = 4
nlam = 11
Cdelt3 = 0.1
dpc = 145
composition_name = "full_astrosil_amira"
total_cores = 8
cores_per_sim = 2

# Optional parameters (shared)
shared_optional_params = {
    "plsig1": "-1.0",
    "plh": "0.1",
    "nphot": "1e6",
    "scattering_mode_max": "1"
}
```

[1] https://github.com/cdominik/optool

4.1 Continuous disk model

```

grid.run(
    model = 0,
    tstar_rstar_list = tstar_rstar_list,
    rdisk = 100,
    mdisk = 1e-3,
    Inc = Inc,
    npix = npix,
    sizeau = sizeau,
    nlmin = nlmin,
    nlmax = nlmax,
    nlam  = nlam,
    Cdelt3 = Cdelt3,
    dpc = dpc,
    composition_name = "full_astrosil_amira",
    template_dir = template_dir,
    total_cores = total_cores,
    cores_per_sim= cores_per_sim,
    optional_params = shared_optional_params
)
```

4.2 Disk with gap mode

```

grid.run(
    model = 1,
    tstar_rstar_list = tstar_rstar_list,
    rout = rout,
    mdisk1 = mdisk1,
    mdisk2 = mdisk2,
    rdisk2 = rdisk2,
    Inc = Inc,
    npix = npix,
    sizeau = sizeau,
    nlmin = nlmin,
    nlmax = nlmax,
    nlam = nlam,
    Cdelt3 = Cdelt3,
    dpc = dpc,
    composition_name = composition_name,
    template_dir = template_dir,
    total_cores = total_cores,
    cores_per_sim = cores_per_sim,
    optional_params_inner ={
        "plsig1": "-1.5",
        "plh": "0.05"
    },
    optional_params_outer ={
```

```
        "plsig1": "-0.5",
        "plh": "0.2"
    }
)

```

4.3 Disk with cavity model

```
grid.run(
    model = 2,
    tstar_rstar_list = tstar_rstar_list,
    rout = rout,
    rdisk = 100,
    mdisk = 1e-6,
    Inc = Inc,
    npix = npix,
    sizeau = sizeau,
    nlmin = nlmin,
    nlmax = nlmax,
    nlam = nlam,
    Cdelt3 = Cdelt3,
    dpc = dpc,
    composition_name = composition_name,
    template_dir = template_dir,
    total_cores = total_cores,
    cores_per_sim = cores_per_sim,
    optional_params = shared_optional_params
)
```

5 Parameter reference

5.1 Required parameters

parameter	description
sim_name	simulation identifier
rdisk1	Outer edge of inner disk (AU) (only for gap model)
rdisk	outer edge of the disk (AU) (for continuous disk and cavity model)
mdisk1	Mas of inner disk ($M_\odot$) (only for gap model)
mdisk	Mas of the disk ($M_\odot$) (for continuous disk and cavity model)
rout	Outer disk radius (AU)
rdisk2	Outer edge of outer disk (AU) (only for gap model)
mdisk2	Mas of outer disk ($M_\odot$) (only for gap model)
output_dir	Output directory
core_range	Assigned CPU cores
tstar	Stellar temperature (Kelvin)
rstar	Stellar radius ($R_\odot$)
inc	Disk inclination (degrees)
npix	Number of image pixels
sizeau	Image size in AU
nlmin	Minimum wavelength (μm)
nlmax	Maximum wavelength (μm)
nlam	Number of wavelengths
Cdelt3	Image cube wavelength step
dpc	Distance to the object (Pc)
Composition_name	Dust composition file name
template_dir	Directory of dust opacity files

Table 1: Required parameters to run the grid.

5.2 Optional parameters

Parameter	Description
nx	Grid resolution in the radial direction (e.g. [100,100] or [100,100,100,100] for gapped disks).
nz	Grid resolution in the vertical direction.
xres_nstep	Number of refinement steps in the radial direction.
xres_nspan	Radial span per refinement level.
xres_nlev	Number of refinement levels in the radial direction.
sigma_type	Surface density profile type.
plsig1 (plsig1_inner, plsig1_outer)	Power-law index for surface density. Can differ for inner and outer disks in the gapped model.
plh (plh_inner, plh_outer)	Power-law index for scale height. Can differ for inner and outer disks in the gapped model.
scattering_mode_max	Scattering mode selection (e.g. 0 for no scattering).
nphot	Number of photon packets for thermal simulation.
nphot_scat	Number of photons used for scattering.
nphot_spec	Number of photons used for the SED.
binary	Boolean to enable binary dust species treatment.
nw	Number of wavelength points in each wavelength segment.
wbound	Boundaries of the wavelength grid (e.g. [0.1, 1.0, 10.0, 25.0, 1000.0]).

Table 2: Optional grid parameters. Parameters with inner/outer versions apply specifically to the gapped disk model.