

DOCUMENTATION

The Meudon PDR Code (1.5.4)

August 9, 2021



1	Overview	5
2	Installation on local computers	9
3	Input parameters	11
3.1	Input files	11
3.2	Geometry & Incident radiation field	11
3.2.1	Interstellar Standard Radiation Field (ISRF)	13
3.2.2	Stellar spectrum	15
3.3	Equation of state: temperature, density and pressure profiles	16
3.3.1	Isochoric, isobaric and user defined density profiles	16
3.3.2	Thermal balance and isothermal models	17
3.4	Convergence and number of global iterations	17
3.5	Cosmic rays ionization rate and secondary UV photon flux	18
3.6	Grains properties and physics	18
3.6.1	Extinction curve and R_V	19
3.6.2	Dust-to-gas mass ratio	19
3.6.3	Grain size distribution	19
3.6.4	DustEM : detailed computation of grains IR emission	20
3.7	Chemistry, species and elementary abundances	20
3.7.1	Species in the chemistry file	21
3.7.2	Elementary abundances	21
3.7.3	Chemical reactions	22
3.7.4	Photo-reactions & UV radiative transfer	22
3.7.5	Note on some chemical reactions	24
3.8	Metallicity	25
3.9	Turbulent velocity	26
3.10	H ₂ formation and excitation	26
3.11	Summary of input parameters	27
4	Outputs and results analysis	31
4.1	Output files	31
4.2	Result extraction and visualization with IDAT	31

4.3	Chemistry analysis	33
4.4	Spherical clouds	33
5	Known problems	35
5.1	Atomic cloud core & H ₂ formation	35
A	List of output files	37

Overview

This document explains how to use the **Meudon PDR code version 1.5.4** (published in 2020 with last update in August 2021). It does not describe all the details of the physics implemented in the code but only the main elements. The physical and chemical processes, and their implementations, are presented in several papers. References are provided below.

The PDR code can be downloaded at <http://ism.obspm.fr>.

The Meudon PDR code computes the atomic and molecular structure of an interstellar cloud. This computation requires to estimate at each position in the cloud a large number of quantities, such as the gas and grains temperatures, heating and cooling rates, formation and destruction rates of chemical species, level excitation for the main chemical species, specific intensity of the radiation field at each wavelength, etc ... Once species densities and their level excitation have been computed at each position, the code computes line intensities and column densities that can be compared to observations.

The code simulates a stationary plane-parallel slab of gas and dust illuminated on one side or on two sides by an external radiation field. For grains, the code considers a distribution of grain sizes, with a power-law MRN distribution for graphites and silicates and a log-normal distribution for PAHs.

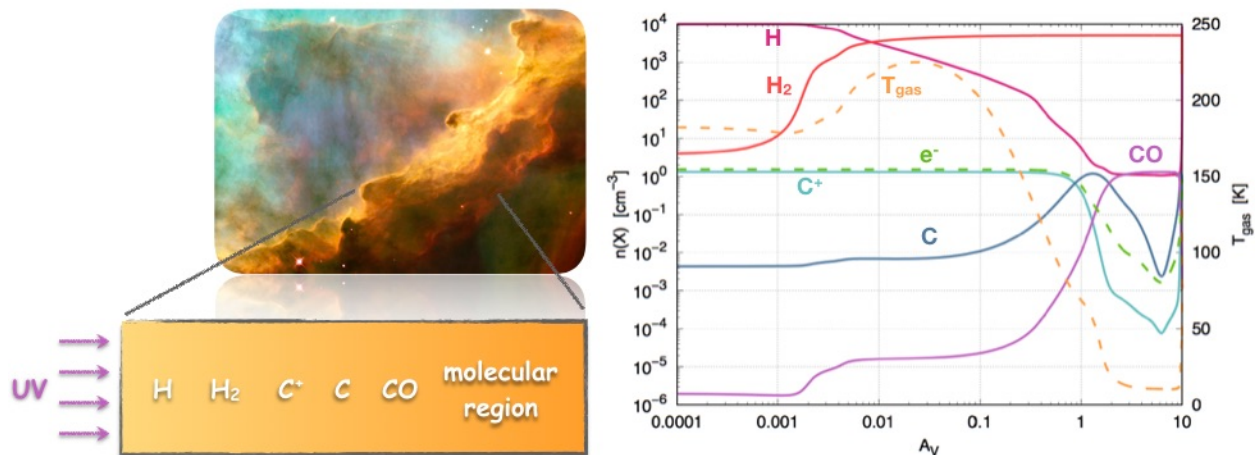


Figure 1.1: Example of density and temperature profiles computed by the PDR code (version 1.5.2) for a model with a constant proton density 10^4 cm^{-3} and a UV radiation field 100 times the ISRF in Mathis units.

The main physical processes solved at each position in the cloud are:

- the **chemistry**. At the initialization step, the PDR code reads a chemistry file that contains the list of chemical species and reactions. At each position, the code computes the densities of these species. This computation is strongly coupled to radiative transfer and thermal balance, since reaction rates depend on the gas and grain temperatures, the UV flux at each position, the charge of grains, etc ... Default chemistry files provided with the code include

several hundred species and thousands of chemical reactions but the code can consider any number of chemical species and reactions. These default chemistry files consider H_2 formation on grains with Langmuir-Hinshelwood and Eley-Rideal mechanisms¹. In PDR models, photo-reactions are important processes. The Meudon PDR code can compute these rates precisely considering the wavelength dependent efficiency of these reactions and the overlapping of absorbing lines at each position in the cloud.

- the **radiative transfer**. The radiative transfer equation is solved at each wavelength from the far-UV to the sub-millimeter domain. The resulting wavelength radiation field is used to 1) estimate photo-dissociation and photo-ionization rates, 2) determine the rate of the photo-electric effect, and 3) determine level excitation of main species as well as their cooling rates. The PDR code considers wavelength dependent absorption, scattering and emission by dust. It also considers several thousands atomic and molecular lines in the whole spectral domain. Details on the radiative transfer methods are presented in Le Petit et al. (2006), Goicoechea et al. (2007) and Gonzalez Garcia et al. (2008).
- the **thermal balance**. At stationary state, temperature is determined assuming balance between heating and cooling. Considered heating mechanisms include the photoelectric effect on grains, cosmic ray ionizations, exothermal chemical reactions, H_2 formation on grains. Concerning cooling, the code considers atomic and molecular line emission, free-free emission and H_2 dissociation. The computation of non-LTE level populations (and so of line intensities) is done for the most important species and takes into account collisional and radiative processes (absorption, spontaneous and induced emission) as well as chemical excitation. Radiative excitation includes non local effect such as IR pumping by dust emission. The code also considers in thermal processes, H_2 vibrational (de-)excitation and gas-grains collisions that can be heating or cooling terms depending on the physical conditions.

These processes are coupled together and the code searches for the solution in an iterative way. A run of the PDR code usually requires several hours². When the most detailed algorithms for the radiative transfer are activated, a run can last several days.

After a run, the code produces several output files in the `out` directory. Various tools provided with the PDR code (see in the "AnalysisTools" subdirectory and documentation therein) must be used to read and use these files. They give access to all quantities computed by the code :

- densities and level populations profiles
- temperature of the gas and of grains for each grain size
- heating and cooling rates by individual processes at each positions
- line intensities for different observation angles
- column densities, (total or per level)
- spectral energy distribution from UV to sub-mm at each position in the cloud
- analysis of the chemical network at each position for any species
- wrap the plan-parallel results in a spherical geometry

The PDR code can be used for the preparation and the interpretation of observations or it can be

¹A version of the PDR code with surface chemistry for other species, with formation of mantles, also exists. If you need it, please contact the developing team of the PDR code. Concerning H_2 , if you need our best detailed treatment that considers grain temperature fluctuations for H_2 formation and ortho/para conversion presented in Bron et al. (2014 & 2016), feel free to contact us.

²Estimations of computing times are provided for standard laptops or cluster nodes

used as a theoretical tool to study the effects of physical and chemical processes in interstellar clouds. It has been used by several teams for very different problems: diffuse clouds, classical PDRs, circumstellar regions, extragalactic and high redshift interstellar gas, ...

Its main limitations are the two hypothesis: stationary state and plane-parallel geometry. Post-processing allows to wrap results in 1D spherical model but assumes the incident radiation field is uniform around the sphere. So it is not possible to model a spherical cloud close to a star.

ISMDB: Interstellar Medium Data Base.

Several thousands PDR 1.5.2 and 1.5.4 models have been pre-computed and are available in ISMDB at <http://ismdb.obspm.fr>. Instead of running the code, you can directly have access to typical models in the database. The database gives access to many quantities online and allow the download of output files that contain all quantities computed by the PDR code. The file format of models downloaded from the database is the same than the one produced by the code. The IDAT and ChemistryAnalyser python tools (provided with the code) are required to explore and extract the content of these files.

Citation:

If you publish results produced by the Meudon PDR code, we would appreciate if you could mention it in your publications, in order for us to keep track of usage statistics for the code. A typical sentence could be:

PDR models published in this paper have been produced with the Meudon PDR code (Le Petit et al. 2006, <http://ism.obspm.fr>).

Contact:

For any question, do not hesitate to contact the team responsible for the PDR code: Emeric Bron, Jacques Le Bourlot, Franck Le Petit and Evelyne Roueff. The generic email address is support.pdr.ism@obspm.fr.

The PDR database and the tools to manipulate the outputs of the code are developed by David Languignon and Nicolas Moreau.

References:

Articles that describe the physical processes in the Meudon PDR code are:

- | | |
|--|---|
| • Le Petit et al., ApJ, 2006 | Main presentation of the Meudon PDR code with results of grids of model in PDR-like conditions |
| • Goicoechea and Le Bourlot, A&A, 2007 | Upgrade the UV radiative transfer considering exact computation of line shielding |
| • Gonzalez-Garcia et al., A&A, 2008 | Upgrade the radiative transfer with consideration of non local effects for pumping, as for example the excitation of H ₂ O by photons produced by dust at the edge of PDRs |
| • Le Petit et al., A&A, 2009 | Surface chemistry using moment equation formalism applied to H ₂ and HD formation on grains |
| • Le Bourlot et al., A&A, 2012 | H ₂ formation on grains by Langmuir-Hinshelwood and Eley-Rideal mechanisms |
| • Bron et al., A&A, 2014 | H ₂ formation on grains considering grain temperature fluctuations |
| • Bron et al., A&A, 2016 | H ₂ ortho-para conversion on grains considering grain temperature fluctuations |

Installation on local computers

Requirements

All versions of the PDR code require a recent Fortran 90 compiler with LAPACK and BLAS libraries. The code is tested with gfortran and ifort compilers.

The latest version of the PDR code, 1.5.4, also requires Python 3.6 and some Python modules (numpy, h5py) that are used to produce output files in HDF5 format. Two tools, IDAT and ChemistryAnalyzer, are also provided to read these HDF5 files and analyze the PDR code results. All the python modules necessary for these tools can be installed simply with the `install_ismpy.sh` script (see IDAT documentation).

Installation of Python 3.6 and required modules with MacPorts on Mac or with apt-get on Ubuntu:

Installation on Ubuntu with apt-get:

```
sudo apt-get install gfortran libblas3gf libblas-doc  
libblas-dev liblapack3gf liblapack-doc liblapack-dev  
python3.6 python-numpy python-h5py
```

On Mac with MacPorts (python modules only) :

```
sudo port install python36 py36-numpy py36-h5py  
To install gfortran on Mac OS, see :  
https://gcc.gnu.org/wiki/GFortranBinaries#MacOS
```

The procedure to install the code is:

- 1- download the source code at <http://pdr.obspm.fr>
- 2- edit and configure `Makefile` in the `src` directory
- 3- compile the code with `make`

This creates the executable `PDR`. Typing in a terminal `./PDR` will run the PDR code with the default input file `data/pdr.in`. Outputs are stored in the `out` directory.

Other programs are provided with the PDR code, such as a version of DustEM¹ that can be used synchronously with the PDR code to compute more precisely grain temperature (see Sect. 3.6.4). In `src/OTHER_PROG`, two small programs are provided : `read_rf_alb` and `build_FractalGrid`. The first one can be used to read `.rf` files produced by the PDR code that contains the energy density at each position and at each wavelength in the cloud. The second one, is devoted to the creation of fractal density profiles.

¹DustEM is a program computing the extinction and the emission of interstellar dust grains heated by photons. It is developed and maintained by Institut d'Astrophysique Spatiale (Orsay - France) and Institut de Recherche en Astrophysique et Planétologie (Toulouse - France). The DustEM website is: <http://www.ias.u-psud.fr/DUSTEM/>.

The PDR code folder is organized as follows:

- `src`: contains the FORTRAN source code.
- `data`: contains input parameter files, chemistry files as well as atomic, molecular and grain data.
- `out`: is the directory where output files are stored.
- `dustem40`: contains the DustEM code. Even if DustEM is not used, the PDR code reads some grain data files in this directory.
- `AnalysisTools`: contains the two Python programs `IDAT` and `ChemistryAnalyzer` to analyze PDR code results.

Note on old versions of the PDR code:

PDR 1.5.2 and PDRLight: Installation of these versions of the PDR code is similar to PDR 1.5.4 excepted that they use python 2.7 instead of python 3.6. These version are no more maintained.

PDR 1.4.4: This old version of the PDR code requires the `cfitsio` library as well as Java. They are used to produce output files in FITS format and to run the `PDRAnalyzer`. The `cfitsio` library can be found at: <http://heasarc.gsfc.nasa.gov/fitsio>. This version is not maintained anymore.

Older versions of the PDR code only require a fortran compiler with LAPACK and BLAS libraries. Output files are produced in a binary format that can be read only on the computer that produced them. The fortran PREP program provided with the PDR code can read these files and extract computed quantities in ASCII format. These versions are not maintained anymore.

First run - test model

The Meudon PDR code is not difficult to run. A good starting point is to run the code with the default input file, `data/pdr.in`, and to play with the analysis tools, `IDAT` and `ChemistryAnalyzer` located in the `AnalysisTools` directory, to read the results, extract and plot some quantities. The default parameters in `data/pdr.in` correspond to a diffuse cloud model so the physical and chemical processes that take place are more limited than in dense and bright PDRs.

- 1- Compile the code in the `src` directory with the command `make`.
- 2- Run the code: in the `src` directory type `./PDR`

The execution time should be of a few hours after what output files can be found in the `/out/ExampleDiffuse` directory. If everything ran fine, a file named `ExampleDiffuse_s_20.hdf5` should be present. The most important quantities computed by the code are stored in this file.

The `QuickTest.in` model is a un-physical case that runs even faster. It does only the first iteration corresponding mostly to initialization of variables. It can be used to check that the code runs and produce output files properly.

To extract quantities from this file, use the `IDAT` python tool. Go in the `AnalysisTools/IDAT` directory and run it with the command: `./idat da`, a tab will open in your browser prompting you to load a HDF5 file. Load the file `ExampleDiffuse_s_20.hdf5` and start exploring its content interactively (a user guide for `IDAT` can be found in the `AnalysisTools/IDAT` folder).

Input parameters

When ready to run a real model, it is time to modify the default input parameter file.

Run the PDR code

By default, the PDR code reads input parameters in the `data/pdr.in` file:
`./PDR` (run the code with `./data/pdr.in`)

It is possible to provide the name of a specific input file:
`./PDR ../data/MyInputParameters.in`

Before launching a model, several questions have to be asked:

- **Which geometry to choose ?** A 1 side or a 2 sides model ?
In most "real" cases, 2 sides models are better choices. Even to study a bright PDR, it is better to simulate a 2 sides cloud with a large size and a small radiation field on the back side than to use a 1 side model.
- **What kind of incident radiation field should be used ?**
In the PDR code it is possible to submit the cloud to the ISRF scaled by a chosen factor or to use a stellar spectrum (or the combination of both, star + ISRF). In the literature, models of PDRs often simulate the radiation field of a close star by the ISRF scaled by a large factor. An important difference between the two kinds of radiation fields (apart from the fact they do not have the same spectral shape) in the PDR code is that the ISRF illuminates the cloud isotropically whereas stellar spectra are assumed to be beamed incident radiation fields and so penetrate deeper in the cloud. For PDR modeling, we recommend, when possible, to use a stellar spectrum to illuminate the cloud plus the ISRF with a scaling factor of 1.0 instead of multiplying the ISRF by strong factors. To simulate a cloud far from any star (diffuse clouds, dark clouds), it is fine to use only the ISRF.
- **Which equation of state to use ?**
The PDR code can simulate constant density models and isobaric models (it can also use a pre-defined profile in density or pressure, and temperature). Models with constant density are easier to analyze than constant pressure models but for comparison with observations, isobaric models may be more realistic. By experience, we recommend to use isobaric models to interpret observations in PDRs. But when starting with the code, use constant density models; understanding code results will be easier.

Then, it is time to think about more specific parameters such as the grains properties (size distribution, quantity, minimum and maximum radii, ...), cosmic rays ionization rates, etc. as well as the list of chemical species to consider, their elementary abundances and their chemical network. These quantities must be specified in several input files.

3.1 Input files

Input parameters to control a run are provided in files located in the `data` directory. The two most important ones are the `data/pdr.in` file and chemistry files identified by the extension `.chi` and located in the `data/Chemistry` directory.

The list of input parameters found in `pdr.in` is presented at the end of this chapter (Tab. 3.3). The sections below explain how to modify these parameters to simulate specific interstellar objects. At the start of each section, the names of the parameters that have to be modified in order to control some aspects of the model are listed. A summary is presented in Tab. 3.1.

3.2 Geometry & Incident radiation field

Parameters: `Avmax`, `F_ISRF`, `radm`, `radp`, `srcpp`, `d_sour`

Table 3.1: Input parameters and their relation to model aspects

To control:	Parameters to modify:
Geometry and Incident radiation field	Avmax, F_ISRF, radm, radp, srcpp, d_sour
Equation of state (isochoric, isobaric, isothermal):	densh, iegth, tgas, ifisob, fprofil, presse
Convergence of the code / Iterations	ifafm
Grains properties	los_ext, rrr, cdunit, alb, gg, gratio, rhogr, alpgr, rgrmin, rgrmax, F_dustem
Cosmic rays ionization rate	fmrc
List of species & elementary abundances	chemistry and modify the chemistry file (data/Chemistry/*.chi)
Metallicity	metal
Photo-dissociation & UV radiative transfer	itrfer, jfgkh2
H ₂ sticking and excitation	istic, iforh2

The size of the cloud is controlled by the A_{vmax} parameter, the total visual extinction. For a constant density model, the conversion to a size in centimeters is simple:

$$L = \frac{N_H}{n_H} = \frac{C_D}{n_H} \times \frac{A_V}{R_V}$$

where N_H is the column density of protons in cm^{-2} , n_H is the volume density of protons in cm^{-3} , C_D is the ratio between the column density of protons and the reddening and R_V is the ratio of the visual extinction to the reddening (C_D and R_V are two parameters of the `pdr.in` file and are described in the grain section).

In a non constant density case, the physical size is related to A_v as

$$L = \frac{C_D}{R_V} \times \int_0^{A_v} \frac{dA'_V}{n_H(A'_V)}$$

The proton density in the PDR code is defined as:

$$n_H = n(\text{H}) + 2n(\text{H}_2) + n(\text{H}^+) + \dots$$

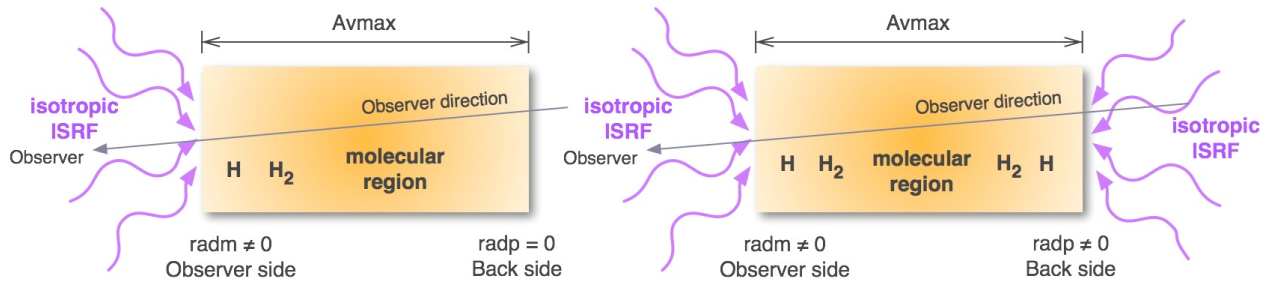


Figure 3.1: One-side and two-sides models. In the one-side case, $radp = 0$ to not add the ISRF on the back side.

Concerning the external radiation field, it is possible to illuminate the cloud on one side or on both sides. To produce a two-sides model, external radiation fields scaling factors have to be non zero on both sides of the cloud ($radm$ and $radp$), whereas to produce a one-side model, no external radiation field must be declared on the back side ($radp = 0$).

Definition: The left side is always the observer side. On the left side, at the first cell, the visual extinction (or optical depth or position in centimeter) is 0.0. The right side is called the back side. This identification between left side and observer side is important when dealing with line intensities. Line intensities provided by the PDR code are computed assuming they emerge from the left side.

Two kinds of radiation fields, that can be combined, can illuminate each side of the cloud:

- the **interstellar standard radiation field (ISRF)** eventually scaled by multiplicative factors $radm$ and $radp$ corresponding to respectively the observer side and the back side of the cloud. This scaling factor corresponds to χ or G_0 that can be found in the literature (in this document, we will use χ as the scaling factor of the Draine or Mathis ISRF and G_0 for the scaling factor to the Habing radiation field). By definition this radiation field is isotropic.
- a **stellar spectrum**. Stellar spectra are *beamed* radiation fields. They penetrate deeper in the cloud than ISRF radiation field. To use a stellar spectrum, one has first to specify the distance in parsecs between the star and the cloud thanks to the d_{sour} parameter, then, one has to specify the shape of the stellar spectrum with the $srcpp$ parameter. The stellar spectrum can be either a black body or a specific spectrum provided by the user (see Sect. 3.2.2).

Thanks to these parameters, it is possible to define several configurations. Figure 3.2 presents a 2 sides model in which a cloud of size $A_V = 20$ is bathed in the ISRF and is illuminated by a B 0 V star at 0.4 pc on the observer side.

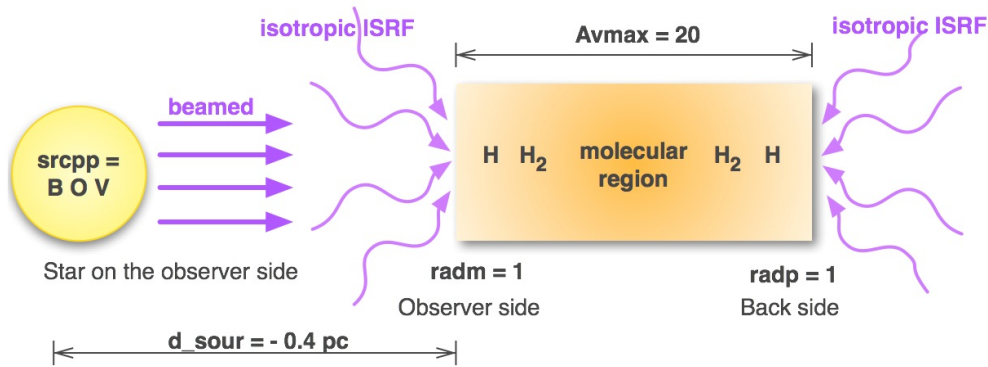


Figure 3.2: Example of a 2 sides models on which both sides of the cloud are illuminated by the ISRF. The observer side of the cloud is also illuminated by a B 0 star located at 0.4 pc.

Excepted for specific tests, isotropic ISRF should always be present even if there is a star.

Typical values

To study the chemistry of dark clouds, one can simulate a 1-side or 2-sides model with $A_V = 20$ or more and analyze results once the UV radiation field has been absorbed. For diffuse clouds, a typical value of A_V is of the order of 0.5 with $\chi = 1$ on both sides. For dense and bright PDRs, one can simulate a 2-sides models with a large A_V and with the observer side illuminated by a strong radiation field, for example $\chi \approx 10^3$, or better, keep $\chi = 1$ and add a stellar spectrum. The back side would then be illuminated by the ISRF, $radp = 1$.

3.2.1 Interstellar Standard Radiation Field (ISRF)

The ISRF used in the Meudon PDR code goes from far-UV (912 Å- the Lyman limit) to the millimeter domain. It is the sum of 4 components.

- **Far-UV to Near-UV** : The prescription for this component can be chosen between 2 expressions: Mathis et al. (1983) or Draine (1978). This is controlled by the F_ISRF flag in the input parameter file. We recommend to use Mathis expression for the ISRF since it takes into account more precisely the near-UV to near-IR components of the ISRF.
 - $F_ISRF = 1$: Mathis ISRF
 - $F_ISRF = 2$: Draine ISRF

- **Near-UV - Visible - Near IR** : Cold stars are responsible for this part of the ISRF spectrum. Our expression is an update of Mathis et al. (1983). It is the combination of 3 black bodies at 6184, 6123 and 2539 K.
- **Dust emission (IR)** : The external IR component produced by dust (in the surrounding environment) has been estimated by the DustEM code. The resulting specific intensity is the sum of the emission by PAHs, very small grains and big grains. Data are provided in the file :
data/Astrodata/IR_field_dustem.dat.
- **Cosmic microwave background** assumed to be a black body at the temperature of the CMB, by default 2.73 K. This temperature can be modified in the source file `PXDR_INITIAL.f90`.

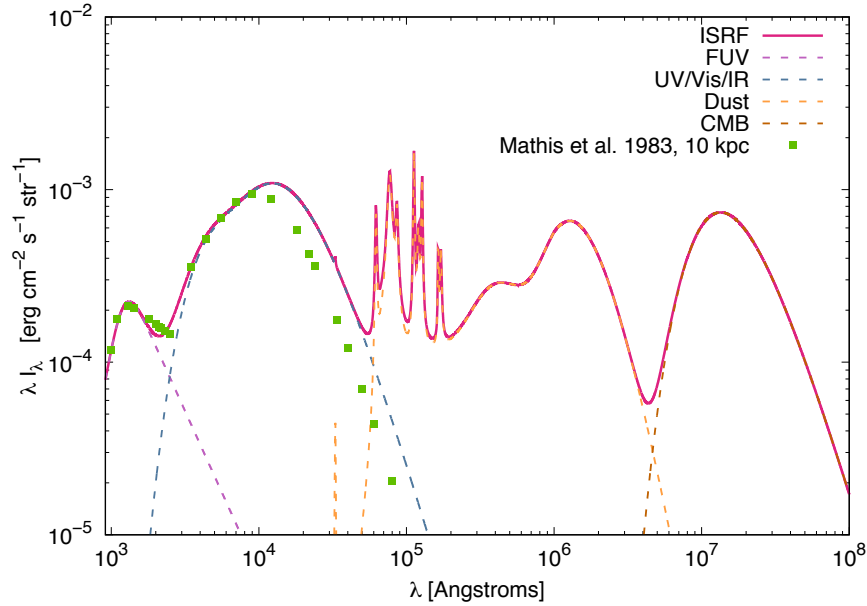


Figure 3.3: ISRF in the Meudon PDR code

The scaling factors `radm` and `radp` are applied only to the first component. An interpolation is done to merge this component with the other ones. In the case of strong radiation fields ($\chi > 100$) this may produce a discontinuity in the ISRF. Usually, this discontinuity does not affect results but it is not physical. Indeed, strong values of χ indicate the presence of stars in the neighborhood. So a better model than high values of `radm` and `radp` is to set them to 1 and to add a stellar spectrum.



The chemistry file used for the model must consistent with the selected ISRF formalism (Mathis or Draine). Indeed, photo-reaction rates depend on the shape of the ISRF. Several chemistry files are provided corresponding to Mathis and Draine's radiation fields. It is indicated in their filename.

Mathis' prescription

The expression of the far UV radiation field based on Mathis et al. (1983) and Black (1994) and fitted by Jacques Le Bourlot is :

$$\lambda \leq 8000 \text{ \AA}, \quad I(\lambda) = [\tanh(4.07 \cdot 10^{-3} \times \lambda - 4.5991) + 1.0] \times 107.192 \times \lambda^{-2.89}$$

In this expression, the wavelength is in Angstroms and the specific intensity, I , in $\text{erg cm}^{-2} \text{ s}^{-1} \text{ \AA}^{-1} \text{ sr}^{-1}$. The intensity of this component can be scaled by the `radm` and `radp` parameters in the input data file.

Draine's prescription

As noted by Mathieu Kopp during his PhD thesis, different expressions of Draine's ISRF can be found in the literature. We use the expression by Sternberg and Dalgarno (1995) because of its good precision. Draine's ISRF is only provided

up to 2000 Å. Nevertheless, in the latest versions of the PDR code, we use it up to 10 000 Å. Above 2000 Å, the Near-UV/Visible/Near-IR component, provided by Mathis et al. dominates the extrapolation of Draine's ISRF.

$$I(\lambda)_{\text{Draine}} = \frac{1}{4\pi} \left[\frac{6.3600 \cdot 10^7}{\lambda^4} - \frac{1.0237 \cdot 10^{11}}{\lambda^5} + \frac{4.0812 \cdot 10^{13}}{\lambda^6} \right] \quad \lambda \leq 2000 \text{ Å}$$

Important point: By definition, the ISRF as defined in Habing (1968); Draine (1978) and Mathis et al. (1983) is the specific intensity produced by a typical distribution of stars at a point of the ISM far from any object. This point sees photons coming from 4π sr. A point at the edge of an opaque cloud sees only photons from 2π sr. Such corrections is done automatically by the PDR code. Moreover, the PDR code considers backscattering by dust and, for 2-sides models, the UV photons coming from the other side that are not absorbed. So, in the output, one must not be surprised that the density of energy at the edge of the cloud be roughly half of the initial input value.

Comparison of the expressions of the ISRF

In the following table, we provide the energy densities for various expressions of the ISRF. Often, the Habing (1968) radiation field is used as reference. We remind that in this paper, H. J. Habing provides estimations of the ISRF for 3 wavelengths (1000, 1400 and 2200 Å) with large uncertainties. Moreover, in the literature, one finds scaling factors between the various expressions of the ISRF. These scaling factors depends on the domain of integration. Habing's paper title is "The interstellar radiation field density between 912 and 2400 Å" but the commonly used conversion factors are based on Draine & Bertoldi (1996) who consider the interval 912 - 1110 Å because H₂ has only weak absorptions in $v = 0$ for larger wavelengths. In the table below values for the Habing radiation field are obtained with the fit presented in Draine and Bertoldi (1996):

$$\lambda u_{\lambda} = \left[-\frac{25}{6} \left(\frac{\lambda}{1000} \right)^3 + \frac{25}{2} \left(\frac{\lambda}{1000} \right)^2 - \frac{13}{3} \frac{\lambda}{1000} \right] \times 10^{-14} \text{ ergs cm}^{-3}$$

with λ in Angstroms.

	$\int u_{\lambda} d\lambda$ [ergs cm ⁻³]	$\int u_{\lambda} \times c d\lambda$ [ergs cm ⁻² s ⁻¹]	$\int u_{\lambda}/E_{\lambda} \times c d\lambda$ [photons cm ⁻² s ⁻¹]	Ratio relative to Habing	Ratio relative to Draine
Integration between 912 and 1110 Å					
Habing	$7.98 \cdot 10^{-15}$	$2.39 \cdot 10^{-4}$	$1.22 \cdot 10^7$	1	0.59
Draine	$1.35 \cdot 10^{-14}$	$4.05 \cdot 10^{-4}$	$2.10 \cdot 10^7$	1.7	1
Mathis	$1.01 \cdot 10^{-14}$	$3.03 \cdot 10^{-4}$	$1.56 \cdot 10^7$	1.3	0.75
Integration between 912 and 2400 Å					
Habing	$6.21 \cdot 10^{-14}$	$1.86 \cdot 10^{-3}$	$1.50 \cdot 10^8$	1	0.61
Draine	$1.01 \cdot 10^{-13}$	$3.03 \cdot 10^{-3}$	$2.33 \cdot 10^8$	1.6	1
Mathis	$6.42 \cdot 10^{-14}$	$1.92 \cdot 10^{-3}$	$1.45 \cdot 10^8$	1.0	0.64

3.2.2 Stellar spectrum

To add a stellar spectrum, one has to specify in the `pdr.in` file:

- 1) the **distance in parsecs between the star and the cloud** in the variable `d_sour`. The star can be located on the observer side or on the back side of the cloud.
 - `d_sour < 0` : the star is on the observer side of the cloud
 - `d_sour > 0` : the star is on the back side of the cloud
 - `d_sour = 0` : no stellar spectrum is added
- 2) a stellar spectrum with the `srcpp` parameter. You can choose either to select a spectral type type, in this case the code will use a black body, or to provide the name an ASCII file containing a specific spectrum (Kurucz spectrum for example). The list of recognized spectral types can be found in `data/Astrodata/star.dat`.

Important: To not add a stellar spectrum, do not forget to fix the `d_sour` parameter to 0.0.

To provide a specific stellar spectrum, build a file containing the flux as a function of wavelength. This file has to be stored in the `data/Astrodata` directory. The name of this file must begin with `F_`. Then provide the name of this file in the `pdr.in` file in the `srcpp` parameter. The format for this file is:

- first line : radius of the star in solar radius
- second line : effective temperature in K
- third line : number of points in the spectrum
- forth line : comment
- then the file must contain the spectrum in two columns with, on the first column, wavelengths in nm and on the second one the flux in $\text{erg cm}^{-2} \text{s}^{-1} \text{nm}^{-1} \text{sr}^{-1}$.

Example of a stellar spectrum file:

```
2.26      # Star radius in solar radius
10500.0   # Effective temperature [K]
1170      # Number of points in wavelength in the spectrum
#nm       Flux (erg cm-2 s-1 nm-1 sr-1)
90.500    26.7754
91.500    820.921
92.500    5083.97
93.500    8428.85
...
```

3.3 Equation of state: temperature, density and pressure profiles

Parameters: `densh`, `ieqth`, `tgas`, `ifisob`, `fprofil`, `presse`

Temperature, density and pressure¹ profiles can be fixed or computed in different ways. The code can deal with :

- isothermal models
- constant proton density models
- isobaric models
- user defined density, pressure, and temperature profiles

Thanks to the user defined profiles, it is possible to model clumpy or fractal media.

3.3.1 Isochoric, isobaric and user defined density profiles

The PDR code accepts different state equations to control the temperature and proton density profiles. State equation is controlled by the `ifisob` and the `fprofil` parameters.

<code>ifisob</code>	Description
0	Constant density model: the proton volume density, n_{H} , is fixed to the value provided by the <code>densh</code> parameter in <code>pdr.in</code> . The unit of <code>densh</code> is cm^{-3} .
1	specific density and temperature profiles provided in an ASCII file by the user is adopted. In this case, if thermal balance is solved, temperature read in the file is not used. See below for how to build this input file.
2	Isobaric model: the thermal pressure, $P = n_{\text{tot}} \times T$ is constant and fixed to the value of the <code>press</code> parameter. The unit of <code>press</code> is K cm^{-3} . With thermal balance computation activated, the temperature decrease in the core will produce a rise of the density.
4	specific pressure and temperature profile provided in an ASCII file by the user is adopted. In this case, if thermal balance is solved, temperature read in the file is not used. See below for how to build this input file.

¹We use the term pressure for thermal pressure ($n_{\text{tot}} \times T$). That is the only pressure considered by the code.

To simulate PDRs, by experience, we recommend to use isobaric models (see. Joblin et al., 2018) - or user defined profiles. Note that isobaric models are usually longer to run. They may also be difficult to interpret because of the variation of both density and temperature.

Typical values:

- for proton volume densities, typical values are $n_H = 1\text{-}200\text{ cm}^{-3}$ for diffuse ISM and $10^4\text{-}10^6\text{ cm}^{-3}$ for dense PDRs or dark clouds.
- for thermal pressure, typical values are $P = 10^3\text{-}10^4\text{ K cm}^{-3}$ for diffuse ISM and $10^5\text{-}10^8\text{ K cm}^{-3}$ for PDRs.

User defined density, pressure and temperature profiles. To use a fixed density-temperature profile (or pressure-temperature profile), an ASCII file with the gas conditions at each position must be prepared. The file should have a name with the extension `.pfl` and it should be stored in the `data/Astrodata` directory. The name of this file should be provided in the `fprofil` parameter in `pdr.in`. This parameter is not used if `ifisob` is not 1 or 4. The format of `.pfl` files is:

- First line : number of points in the profile.
- Following lines, on three columns : visual extinction, temperature in K, proton density in cm^{-3} (or pressure in K cm^{-3}).

Data must start at a visual extinction of 0.0. The PDR code computes the structure of the cloud up to `Avmax` and not up to the maximum visual extinction in the `.pfl` file. So, data in `pfl` files must be provided at least up to `Avmax`.

When using user-defined profiles, attention should be paid to discontinuities. Consider to have enough points to avoid discontinuities. Avoid steep transitions between low density medium and high density medium (for example, instead of a steep step, use instead a hyperbolic tangent variation).

3.3.2 Thermal balance and isothermal models

The parameter `ieqth` allows to switch between isothermal models or models in which thermal balance is solved.

- `ieqth = 0`: isothermal model. The gas temperature is fixed at the value provided in the `tgas` parameter.
- `ieqth = 1`: isothermal model. Thermal balance is solved and the code searches at each point the temperature so that the total heating rate equals the total cooling rate. The value provided in the `tgas` parameter is not used.

In the case of models with user defined density profile (`ifisob = 1`), the `.pfl` file containing the proton density at each position in the cloud must also have a column providing the gas temperature. If `ieqth = 0`, then the temperature in the file is used. If `ieqth = 1`, then thermal balance is solved. Temperatures in the profile file are then read but not used.

3.4 Convergence and number of global iterations

Parameter : `ifafm`

The parameter `ifafm` is an important technical parameter.

The PDR code converges towards a solution after several iterations on the whole cloud. Indeed, each point of the slab sees a radiation field coming from both sides (even if there is no radiation field on the back side, backscattering by dust induces a radiation field from the back side towards the observer side). Determining shielding efficiencies in both directions require to estimate the atomic and molecular composition everywhere in the cloud and this is done after several iterations on the whole cloud. The number of iterations the code must do has to be provided in the `ifafm` parameter.

By default, the code writes output files for the last and the second last iterations. The index of iterations is written in output filenames before the file extension (Ex: `MyModel_s_20.hdf5` is the 20th iteration).

Typical values:

- `ifafm` should never be lower than 10 (some physical processes are activated progressively).
- A typical value is 20 but sometimes a larger value is required.



After a run, check if the model has converged. By default, the code writes output files for the last and the second last iterations. Extract the quantities you are interested in from the last iteration as well as the second last one and check if they give the same results.

3.5 Cosmic rays ionization rate and secondary UV photon flux

Parameters: `fmrc`

Cosmic rays ionization rate, ζ , is controlled by `fmrc` with, as unit, 10^{-17}H_2 ionization per second.

$$\zeta = f_{\text{mrc}} \times 10^{-17} \text{ s}^{-1}$$

The actual ionization rate by cosmic rays in the model is determined by this parameter and by some rates in the chemistry file. Default chemistry files have for cosmic-ray ionization:

					gamma	beta	alpha
h	crp	h+	electr		4.60E-01	.00	.00 1
h2	crp	h+	h	electr	4.00E-02	.00	.00 1
h2	crp	h2+	electr		9.60E-01	.00	.00 1

The code computes cosmic-ray ionization rates with expression:

$$\left. \frac{dn(X)}{dt} \right|_{\text{c.r. ioniz.}} = \pm \gamma f_{\text{mrc}} 10^{-17} n(X) \quad [\text{cm}^{-3} \text{ s}^{-1}]$$

The total ionization rate by cosmic rays per H_2 molecule is nearly twice the one of atomic hydrogen. Branching ratios for the H_2 ionization comes from private communication with Alex Dalgarno.

The flux of secondary UV photons is also controlled by `fmrc` and is:

$$F_{\text{sec. UV}} = 10^3 \frac{\zeta}{10^{-17}} = 10^3 \times f_{\text{mrc}} \quad [\text{photon cm}^{-2} \text{ s}^{-1}]$$

Typical values:

- Diffuse clouds: `fmrc` = 10 (so $\zeta = 10^{-16}$).
- Dense clouds: `fmrc` = 1-5 (so $\zeta = 10^{-17}$ - 5×10^{-17}).

3.6 Grains properties and physics

Parameters: `los_ext`, `rrr`, `cdunit`, `alb`, `gg`, `gratio`, `alpgr`, `rgrmin`, `rgrmax`, `F_dustem`

The PDR code considers grain distributions with a MRN (Mathis et al. 1977) law for amorphous carbons and silicates and a log-normal law for PAHs.

Grains properties are involved in 3 important physical aspects:

- They determine the extinction curve used in the UV radiative transfer
- They catalyze H_2 formation and some other chemical reactions.
- They contribute to thermal balance through photo-electric effect and collisions with the gas. This last process can contribute either to heating or cooling of the gas depending on the difference of temperature between the gas and grains.

3.6.1 Extinction curve and R_V

The optical depth due to dust absorption writes:

$$\tau_\lambda = \left[1 + \frac{1}{R_V} \frac{E(\lambda - V)}{E(B - V)} \right] \frac{A_V}{2.5 \log(e)}$$

The Meudon PDR code uses Fitzpatrick and Massa (1990) formalism to parametrize the extinction curve. This formalism provides the term: $E(\lambda - V)/E(B - V)$. Parameters for several lines of sight are implemented in the code. They can be found in the `line_of_sight.dat` file stored in the `data/Astrodata` directory. New lines of sight can be introduced adding Fitzpatrick and Massa parameters in this file. The R_V parameter must be modified accordingly to the choice of the line of sight extinction curve.

To select an extinction curve, provide the name of a line of sight in the `los_ext` parameter and a value for R_V in the `rrr` parameter. The `line_of_sight.dat` file lists possible values.

Typical values:

- mean Galactic extinction curve: `los_ext = Galaxy` and `rrr = 3.1`

3.6.2 Dust-to-gas mass ratio

The volume density of grains is controlled by the dust-to-gas mass ratio that should be provided in the `gratio` parameter. The amount of PAHs compared to dust, is controlled by the PAH-to-dust (`q_pah`).

The relation between proton column density and reddening must also be provided:

$$C_D = \frac{N(\text{H}) + 2 \times N(\text{H}_2)}{E(B - V)}$$

This quantity is controlled by the `cdunit` parameter.

Mean Galactic values:

- `cdunit` = $5.8 \cdot 10^{21} \text{ cm}^{-2} \text{ mag}^{-1}$ (Bohlin et al., 1974, Rachford et al., 2002, ApJ, 577, 221).
- For the dust-to-gas mass ratio, a typical value is `gratio` = 0.01.
- If PAHs are introduced, a typical value for `q_pah` is $4.6 \cdot 10^{-2}$.



These three parameters are scaled by the metallicity parameter. At the initialization phase, the PDR code multiplies the grain-to-dust mass ratio by the metallicity Z and divide C_D by Z .

3.6.3 Grain size distribution

It is possible to modify the power-law grain size distribution in the input file. This power-law is controlled by 3 parameters: the minimum and the maximum radius of grains (`rgrmin` and `rgrmax`, both in centimeter) and α , the slope of the power-law (`alpgr`). So the volume density of grains is:

$$n_{gr} \propto \int_{rgrmin}^{rgrmax} r^{-alpgr} dr$$

Mean Galactic values:

- `rgrmin` = $1 \cdot 10^{-7} \text{ cm}$ & `rgrmax` = $3 \cdot 10^{-5} \text{ cm}$
- `alpgr` = 3.5

Available grain surface is highly dependent on the minimum grain radius. So `rgrmin` can have an important impact on surface chemistry and on the H_2 formation rate.

The dust-size distribution is discretized in the code with all computations being done for a set of grain size values. In the output quantities, you can find these sizes labelled as `Grain radius(r01)`, `Grain radius(r02)`, etc., and the cor-

responding grain temperatures (`Grain temperature(r01)`, etc.), individual grain charges (`Grain charge(r01)`, etc.), and grain number density per unit volume for each size bin (`Grain density(r01)`, etc.).

3.6.4 DustEM : detailed computation of grains IR emission

DustEM (<http://www.ias.u-psud.fr/DUSTEM/>) is a code developed at Institut d'Astrophysique in Orsay devoted to the study of the emission of interstellar grains. It assumes a distribution of grains (different components and sizes) and, for a given incident radiation field, computes the IR emission of the grains.

A version of DustEM is provided with the PDR code. It is located in the `dustem40` directory and has to be compiled separately.

The PDR code computes grain temperatures and photo-electric effect following the Bakes & Tielens (2009) prescription. Grain IR emission is then computed. It is possible to replace this part of the code by a call to DustEM. In this case, at selected positions in the cloud, the UV radiation field computed by the PDR code is sent to DustEM, which sends back the IR contribution by dust. This permits to get a more precise IR radiation field. Note that using DustEM significantly increases the computation time.

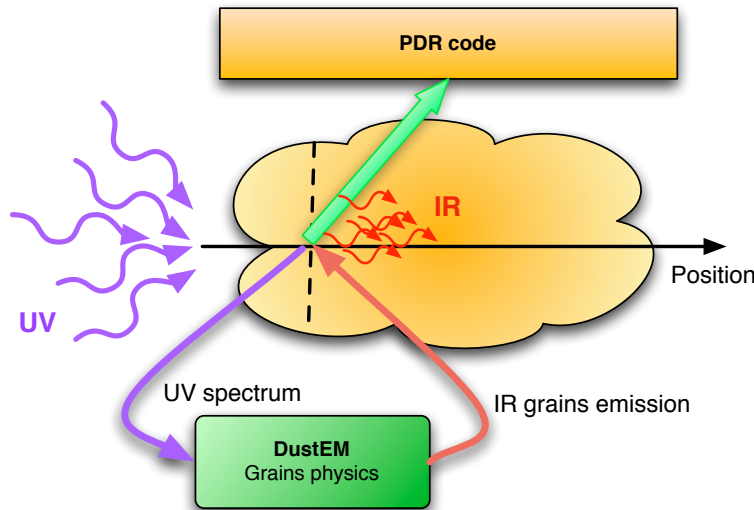


Figure 3.4: Coupling the PDR code and DustEM

The use DustEM is controlled by the flag `F_dustem` :

- `F_dustem = 0` : DustEM is not used
- `F_dustem = 1` : DustEM is activated

3.7 Chemistry, species and elementary abundances

Species used in a model and the corresponding chemical network have to be provided in a specific file located in `data/Chimie`. Two default chemistry files are provided with the code. These files are identified by the `.chi` extension. One of them must be used with Mathis ISRF and the other one with the Draine ISRF. It is indicated in their name.

The name of the chemistry file to use is one of the parameters in the input file `pdr.in`.

The PDR code computes the density at each position of all species present in the chemistry file. For some of these species, the PDR code can compute their densities in their quantum levels and the corresponding line intensities. The list of species for which it is possible to compute level populations is in the file `data/spectre.flag`. Each lines corresponds to a species and the first column is either 0 (excitation not computed for this species) or 1 (excitation computed for this species).

3.7.1 Species in the chemistry file

Chemistry files are divided in two parts. The first part corresponds to the list of species to consider in the model. It ends with the number 888. The list of species must be provided in a specific format and in a specific order.

The order to provide species is: neutral species, cations, anions, chemisorbed species, physisorbed species. Chemisorbed species are identified with the symbol `::` after the name and physisorbed species are identified with `·`. For example, chemisorbed H is written `h::` and physisorbed H atoms are written `h·`.

For each species, on one line, the chemistry file must contain:

- index (not used by the code)
- name with a maximum of 8 characters
- the elementary composition of atoms and molecules (several integer columns)
- the charge
- the initial abundance relative to protons
- the enthalpy of formation in kcal mol^{-1}
- INCHI key (not mandatory - use `NOINCHIFOUND` if not provided)
- comment (not mandatory)

The elementary composition is used to check the conservation of elements. The order of atoms is, on two digits H and C, and then on one digit, N, O, He, D, ^{13}C , ^{15}N , ^{18}O , Ar, F, Na, Mg, Al, Si, Ne, S, Cl, Ca, Fe².

3.7.2 Elementary abundances

The initial abundance column controls the elementary abundances. These abundances are relative to protons.

For example, in the first part of the chemistry file, one finds:

```
4  c          0 1000 0000 00000 000000 0  .000E+00 169.978
69 c+         0 1000 0000 00000 000000 1  1.320E-04 429.646
```

This means the C/H ratio in the gas phase is 1.32×10^{-4} and that, as initial condition, all carbon atoms are assumed to be in C^+ and nothing in C.



Elementary abundances are additive. In the example above, if a non zero value is given on the C line *and* on the C^+ line, then the elementary abundance for carbon is $\text{C}/\text{H} + \text{C}^+/\text{H}$. It is not recommended to split elementary abundances on several atoms or molecules. Usually, for convergence reasons, it is recommended to start with initial conditions in which elements are in the form they are found at the edge of PDRs, so: H in H, C in C^+ , O in O, etc.

Hydrogen elementary abundance must be exactly 1. For initial conditions, we recommend to spread H atoms between atomic H and H_2 with ratios $\text{H}/\text{H} = 0.800$ and $\text{H}_2/\text{H} = 0.100$.

Elementary abundances are scaled by the metallicity parameter. If for a low metallicity medium, the metallicity parameter is fixed to 0.5 and elementary abundances are also reduced by a factor 2, then final elementary abundances 4 times lower than the local Galactic values. So, be careful to not scale elementary abundances twice.

Check elementary abundances. After a run, the actual elementary abundances used in the model and the chemical network are written in the log file `.def`. Check in this file the proper elementary abundances have been used.

²Fe is not really iron but a pseudo-species simulating all heavy elements but the ones present in the chemistry file. It is used for charge exchange.

3.7.3 Chemical reactions

The chemical network starts after the line 888 in the chemistry file and ends with the line 9999. All lines after 9999 are considered as comments.

Detailed explanation about how to modify the chemical network will be provided in an appendix in a next version of this documentation. Nevertheless, a few important points about how chemical reactions are written in this file are:

- chemical reactions can have 2 reactants and 4 products³
- reaction rates are computed using the three floats provided on each line, called respectively γ , α and β
- the integer after the 3 floats is a flag to identify the type of the reaction. It controls how γ , α and β are used to compute chemical reaction rates

The fortran code corresponding to the computation of reactions rates is in `PXDR_CHEMISTRY.f90`.

For example, type 4 corresponds to gas phase reactions. The reaction:

```
jun16 h      ho2      o2      h2      2.06E-11      .84      277.00      4 ! UMIST
```

has for reaction rate:

$$k = 2.06 \cdot 10^{-11} \left(\frac{T}{300} \right)^{0.84} e^{-277/T}$$

Type	Reactions	rate
0	formation of H ₂ and HD on grains (outdated)	$R \propto 3 \cdot 10^{-17} \text{ cm}^3 \text{ s}^{-1}$
1	cosmic ray processes	$k = \gamma \times \zeta \text{ s}^{-1}$
2	secondary photon processes	$k = \gamma \times f_{H_2} \times \zeta \times (T/300)^\alpha$
3	radiative association	$k = \gamma \times (T/300)^\alpha \times \exp(-\beta/T)$
4	gas phase reactions	$k = \gamma \times (T/300)^\alpha \times \exp(-\beta/T)$
6	endothermal reactions with H ₂	k is a function of the internal energy in H ₂
5	photoreactions with the classical approximation	$P = G_0 \times \gamma \times \exp(-\beta A_V)$
7	photoreactions with integration of cross sections	$P \propto \int \sigma(\lambda) \times I(\lambda) d\lambda$
8	3 body reactions (X + 2 hydrogen atoms)	$k = \gamma \times (T/300)^\alpha \times \exp(-\beta/T)$
10x	exceptions - special expressions	special - see <code>PXDR_CHEMISTRY.f90</code>
11	reactions on grain surface	see Le Bourlot et al. (2012)
12	photoreactions on grain surface	$k = G_0 \times \gamma \times \exp(-\beta A_V)$
13	adsorption on grains	see Le Bourlot et al. (2012)
14	neutralizations with grains	special - see <code>PXDR_CHEMISTRY.f90</code>
17	photodesorption from grains	$k = \gamma \times (4 \times F_{sec,ph} \times f_{H_2} + F_{ph} \times d_{site}^2/4)$
18	thermal evaporation from grains	see Le Bourlot et al. (2012)
19	chemisorption on grains	see Le Bourlot et al. (2012)
20	Eley-Rideal process	see Le Bourlot et al. (2012)
40	radiative and dielectronic recombination of atomic ions	special - see <code>PXDR_CHEMISTRY.f90</code>

Table 3.2: List of reaction types in the PDR code.

3.7.4 Photo-reactions & UV radiative transfer

Parameters: `itrfer`, `jfgkh2`

³It is possible to have 3 reactants if two of them are hydrogen. In that case, the two hydrogen reactants must be introduced with the name 2h and the type for gas phase chemical reactions must be set to 8.

H and H₂ absorption lines in the UV - H₂ shielding

PDR models rely on a detailed treatment of the UV radiative transfer and its interaction with the gas and grains. The radiation field is absorbed in lines and in the continuum by atoms, molecules and by dust.

Rates of photo-reactions depend directly on the UV radiation field. The Meudon PDR code can solve the UV radiative transfer by two methods:

- **FGK approximation:** in this method described in Federman, Glassgold & Kwan (1979), self-shielding of H and H₂ is treated approximatively. Line self-shielding is taken into account but not mutual shielding (neither for lines of the same species nor between lines of different species).
- **Exact method:** What we call the exact method is described in Goicoechea & Le Boulart (2008). Overlapping of H, H₂ and CO UV absorption lines can be taken into account, at the cost of an increased computation time.

The exact method is CPU time consuming but depending on the goal of the study, it may be mandatory to use it. It is often the case for models of diffuse clouds in which the computation of the H/H₂ transition require a detailed treatment of shielding processes. To study complex chemistry in dark clouds, this method is not required (at least for H₂). The FGK approximation may not compute a precise position of the H/H₂ transition but this will not affect what happens in the shielded parts of the cloud.

Parameter `itrfer` is used to select the method :

- `itrfer = 0`: FGK approximation
- `itrfer = 1`: Exact computation of self- and mutual shielding for H and H₂. Another parameter has to be set for H₂: `jfgkh2`. Only lines with initial level $J < \text{jfgkh2}$ are treated exactly, the rest is treated with FGK approximation.
- `itrfer = 3`: Exact computation of self- and mutual shielding for H, H₂ and ¹²CO (for ¹²CO, only lines from the first 30 levels are treated exactly, the rest is treated with FGK approximation).
- `itrfer = 4`: Exact computation of self- and mutual shielding for H, H₂, ¹²CO and ¹³CO (for ¹³CO, only lines from the first 30 levels are treated exactly, the rest is treated with FGK approximation).
- `itrfer = 5`: Exact computation of self- and mutual shielding for H, H₂, ¹²CO, ¹³CO, C¹⁸O (for C¹⁸O, only lines from the first 30 levels are treated exactly, the rest is treated with FGK approximation) and HD (for HD, only lines from the first 3 levels are treated exactly, the rest is treated with FGK approximation).

Typical values

- Because of the computing time, first runs should be done with `itrfer = 0`.
- H₂ absorption lines are narrow above $J = 2$, so a value of `jfgkh2` of 3 is often enough to take into account detailed shielding effects. Most of the time, higher values are only useful to produce nice absorption spectra with H₂ absorption lines.

Output files provide the absorption spectra through the slab of gas as well as the density of energy at each position as a function of the wavelength. With FGK approximation, these spectra do not contain H, H₂ and CO absorption lines whereas in the case of exact radiative transfer, they do (see. Fig. 3.5).

Computation of photo-reaction rates

By default and for species other than H, H₂, C and CO, photo-reaction rates are computed with the expression :

$$\mathcal{P}(A_V) = \mathcal{P}_0 \times e^{-\beta A_V} \quad \left[s^{-1} \right] \quad (3.1)$$

with $\mathcal{P}_0$ the probability of ionization (dissociation) at the edge of the cloud and β a factor computed for a specific radiation field (Mathis or Draine). Both parameters are provided in the chemistry file in type 5 reactions.

With the Meudon PDR code, it is possible to compute more precisely photo-reaction rates. Indeed, for species which ionization and dissociation cross sections, σ_λ , are known, the code can compute photo-reaction probabilities by an integration of these cross sections times the specific density of energy at each position in the cloud:

$$\mathcal{P}(A_V) = \int_{912}^{\lambda_{lim}} \sigma_\lambda \times u_\lambda(A_V) d\lambda \quad (3.2)$$

The file `data/photodest.flag` contains a list of species and photo-reactions paths for which exact photo-reaction rates can be computed. The corresponding data (σ_λ) are located in the `data/Sections` directory. To activate detailed

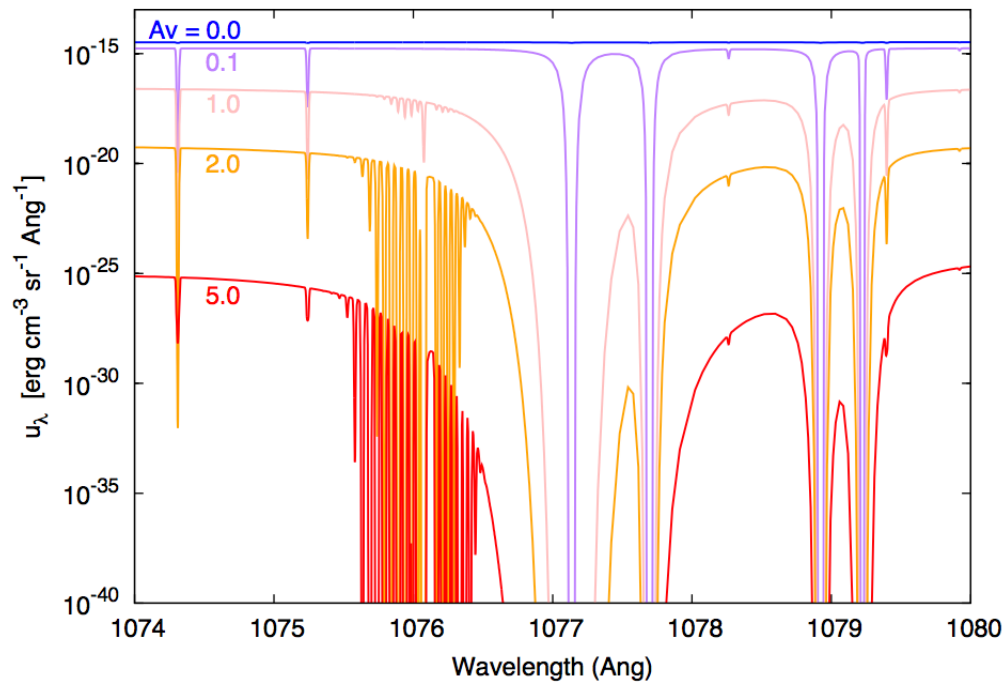


Figure 3.5: Density of energy computed by the PDR code at several A_V with the exact radiative transfer method including H, H₂ and CO absorption lines. Parameters of the model are $n_H = 10^4 \text{ cm}^{-3}$, $\chi_m = 100$.

computation of photo-reaction rates, one has to set to 1 the flag in front of the species names.

For both cases (approximation or detailed computation), photo-reactions must be introduced in the chemistry file with the type 5 that corresponds to the approximated photo-reaction rate (3.1). If detailed computation is activated in the `photodest.flag`, the code automatically modifies the type of the reaction 7 corresponding to (3.1). For historical reasons, the photo-dissociation reactions of H₂ and CO should not be introduced in the chemistry file. These reactions are automatically introduced by the code.

When detailed computation of photo-reaction rates is activated, two cases have to be considered:

- `itrfer = 0`: if the exact radiative transfer method is not activated, then detailed photo-reaction rates are computed using a radiation field extinguished only by dust and atoms absorbing in the continuum such as C.
- `itrfer > 0`: if the exact radiative transfer method is activated, then detailed photo-reaction rates are computed using a radiation field where extinction in lines of H₂ (and other molecules such as CO depending on the value of `itrfer`) is also accounted for. Fig. 3.6 shows an example of the overlapping between resonances of the sulfur ionization rate and H₂ Lyman and Werner absorption lines.

Most of the photo-reaction cross sections and rates used in the Meudon PDR code come from Ewine van Dishoeck website (<http://home.strw.leidenuniv.nl/~ewine/photo>).

3.7.5 Note on some chemical reactions

We try to provide accurate chemical reaction rates in the default chemistry file. Nevertheless, new rates are frequently determined and, for some reactions, discrepancies can be found. Some references for chemical reaction rates are:

- the KIDA database (<http://kida.obs.u-bordeaux1.fr>)
- the UMIST database (<http://udfa.ajmarkwick.net>)

H₂ collisional dissociation rate: In hot gas, H₂ can be dissociated by collisions with H atoms and H₂ molecules. This process is an important cooling process in hot gas. Important discrepancies are found in the literature about this rate. Presently, two rates are implemented in the PDR code, Janev et al. (2008) and Glover & Mac Low (2007). The rate to use must be selected commenting and de-commenting the proper lines in `PDR_CHEMISTRY.f90` (reaction type

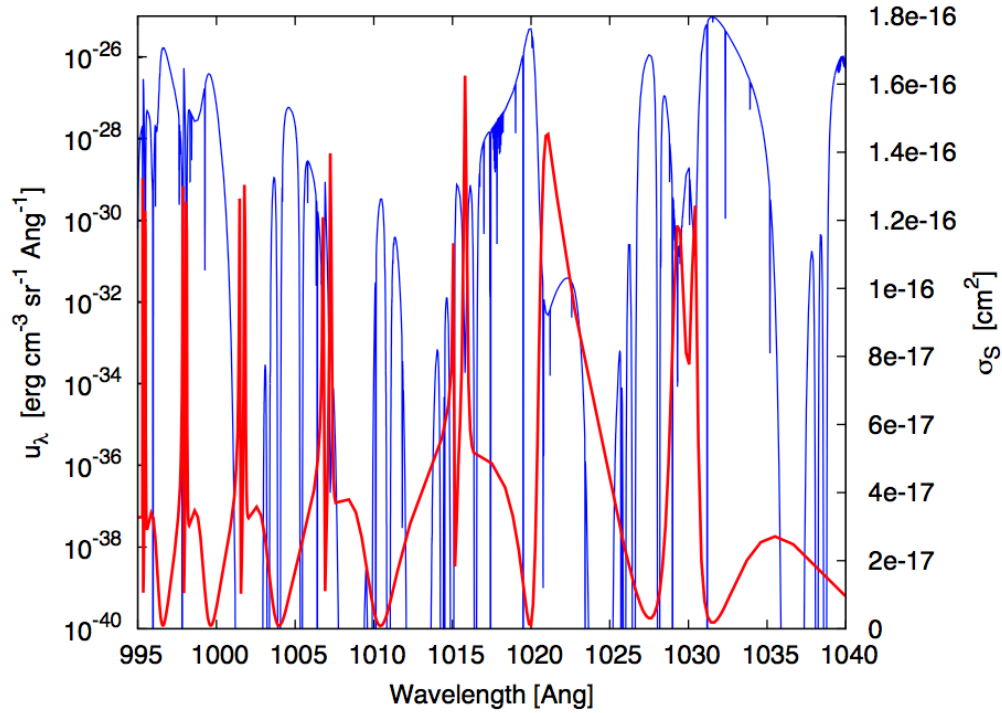


Figure 3.6: Density of energy (blue) at $A_V = 5$ for a model $n_H = 10^4 \text{ cm}^{-3}$, $\chi_m = 100$ and sulfur ionization cross section (red). The absorption lines in the density of energy correspond to Lyman and Werner H_2 transitions.

110). By default, Glover & Mac Low (2007) is selected. The temperature profile in the hot atomic layer of PDRs can be significantly modified depending on the selected rate. We are investigating the impact of this rate (Le Bourlot et al. in prep) and we may change the default value in the future.

3.8 Metallicity

Parameters: `metal`

The metallicity, Z , can be modified in the `pdr.in` input file with the `metal` parameter. Its default value is 1.

It is used in the code to scale several quantities:

- the mass of grains on the mass of gas is multiplied by Z
- the ratio $N_H / E(B-V)$ is divided by Z
- elementary abundances in the chemistry file are multiplied by Z

Example:

```
0.50E+00 ! metal      : metallicity
5.80E+21 ! cdunit    : NH / E(B-V) for Z = 1
1.00E-02 ! gratio    : Mass grains / mass gas for Z = 1
```

and

```
69 c+      0 1000 0000 00000 000000 1 1.320E-04 429.646 ....
```

Then the code scales some parameters and will run with a $m_{\text{grain}}/m_{\text{gas}} = 5 \times 10^{-3}$, $N_H/E(B-V) = 5.8 \times 10^{21}/0.5$ and a $C/H = 1.32 \times 10^{-4} \times 0.5$.

3.9 Turbulent velocity

Parameters: `vturb`

Turbulent velocity is used by the code to compute Doppler line broadening, b :

$$b = \sqrt{\frac{2kT}{m} + v_{turb}^2}$$

It is controlled by the `vturb` parameter and is in km s^{-1} . Large values of the turbulent velocity may increase H_2 self-shielding.

Typical values

- `vturb` = 2 to 6 km s^{-1} .

3.10 H_2 formation and excitation

Parameters: `istic`, `iforh2` and `ichh2`

It is possible to modify some aspects of H_2 processes (formation, excitation, collision rates) in the input files. Most of the time, these parameters should not be modified.

H_2 formation mechanisms are defined in the chemistry file but the formalism of H sticking on grains can be modified in `pdr.in`. In default chemistry files, H_2 is formed following the Langmuir-Hinshelwood (LH) and Eley-Rideal (ER) mechanisms as described in Le Bourlot et al. (2012). The efficiency of these two processes depends on several energy thresholds, which themselves depend on dust composition. It is also possible to fix H_2 formation rate to a constant value (or depending on the gas temperature) in the chemistry file.

H sticking on grains

The formation rate of H_2 (by the Langmuir-Hinshelwood mechanism) on grains also depends on the H sticking coefficient on grains (for physisorption). Several prescriptions can be found in the literature. Parameter `istic` allows to change the prescription.

- `istic` = 1: Sticking factor is the γ factor in the chemistry file for the $\text{H} + \text{grain} \rightarrow \text{H}$: reaction.
- `istic` = 2: Prescription by Sternberg & Dalgarno (1995).
- `istic` = 3: Prescription by Aderson & Vanier (1993).
- `istic` = 4: Prescription from Flower & Pineau des Forêts where $s = \sqrt{10/T_{gas}}$.

Note that chemisorption (for the Eley-Rideal H_2 formation mechanism) uses a separate sticking coefficient (Le Bourlot et al. 2012):

$$s = \frac{e^{-\frac{300\text{K}}{T_{gas}}}}{1 + \left(\frac{T_{gas}}{464\text{K}}\right)^{3/2}}$$

H_2 excitation after formation on grains

H_2 formation on grains releases about 4.5 eV. This energy is spread between the grain, kinetic energy of the molecule and internal energy. Since the repartition between these three possibilities is unknown, we assume equipartition. Several options, controlled by the `iforh2` parameter, are possible to control how internal energy is spread in the levels:

- `iforh2` = 0: Boltzmann distribution with exactly 1.478 eV in internal energy (recommended)
- `iforh2` = 1: Boltzmann distribution with an excitation energy equals to 1/3 of formation energy. This does not correspond to a total internal energy of 4.5/3 eV
- `iforh2` = 2: Formation in $v = 0$, $J = 0$ and $J = 1$.
- `iforh2` = 3: H_2 formed in $v = 14$ and $J = 0$ and 1
- `iforh2` = 4: H_2 formed in $v = 6$ with $T = 65\text{ K}$ for rotational levels

H₂-H collision rates

It is possible to select different H₂-H collision prescriptions with the `ichh2` parameter.

- `ichh2 = 0`: Martin and Mandy prescription
- `ichh2 = 1`: Flower et al. without reactive collisions
- `ichh2 = 2`: Flower et al. with reactive collisions from Schofield (recommended)
- `ichh2 = 3`: Flower et al. with reactive collisions from Schofield plus empiric laws for non-referenced transitions

Typical values

- `istic = 4`: the sticking coefficient of H on grains is $s = \sqrt{10/T}$ and 1 for T_{gas} below 10 K
- `iforh2 = 0`: assure that one third of the H₂ formation energy goes in internal energy
- `ichh2 = 2`: Flower et al. H-H₂ collision rates with reactive collisions from Schofield

3.11 Summary of input parameters

<code>modele</code>	Model name . This is a name for the model. All output files will have this name plus a file extension.
<code>chimie</code>	Chemistry file name . This is the name of the chemistry file name that must be used. This file must be located in the <code>data/Chemistry</code> directory.
<code>ifafm</code>	Number of iterations before the code stops. Default value: 20. Never use a value lower than 10 excepted for debugging.
<code>Avmax</code>	AVmax Size of the cloud expressed as a visual extinction.
<code>densh</code>	Proton density , n_H in cm ⁻³ . Gas density is fixed to this value for isochoric models. This value is not used for isobaric models in which proton density is computed by the code.
<code>F_ISRF</code>	ISRF prescription . This is a flag that can take two values: • 1: Mathis prescription • 2: Draine prescription Depending ISRF prescription, the proper chemistry file must be used.
<code>radm</code>	Scaling factor of the ISRF on the observer side of the cloud. This radiation field is isotropic. Note that the actual value of the radiation field intensity at the edge of the cloud is computed by the code considering the cloud is illuminated in 2π steradians and backscattering by dust.
<code>radp</code>	Scaling factor of the ISRF on the back side of the cloud. This radiation field is isotropic. Note that the actual value of the radiation field intensity at the edge of the cloud is computed by the code considering the cloud is illuminated in 2π steradians and backscattering by dust.
<code>srcpp</code>	Star spectral type . Used only if a star is added as external source. Stellar spectrum is modeled as a beamed radiation field whereas ISRF is modeled as an isotropic radiation field. See in the document for possible values.
<code>d_sour</code>	Distance of the star in parsecs. Syntax is: • if 0.0 : no star illuminate the cloud (default). • if < 0.0 : the star is added to the ISRF on the observer side. • if > 0.0 : the star is added on the back side.
<code>fmrc</code>	Cosmic ray ionization rate . Cosmic rays impact the chemistry, the heating rate, the flux of secondary UV photons. Unit: in 10 ⁻¹⁷ H ₂ ionization per second. Typical value: 10 (corresponding to 1E-16 s ⁻¹).
<code>ieqth</code>	Thermal balance flag . Turn on or off the computation of gas temperature. • 1: Thermal balance solved (default)

<code>tgaz</code>	• 2: Isothermal model Gas temperature. Temperature of the gas in isothermal models. If thermal balance is solved, gas temperature is computed by the code. The value given in <code>tgaz</code> is then used as initial guess, and a well chosen value can help convergence. Typical value: 500 K
<code>ifisob</code>	Equation of state. Flag to choose between isochoric, isobaric or more specific models: • 0: n_H is constant at the value of <code>densh</code>. • 2: P is constant at the value of <code>presse</code> • 1: User defined n_H (and T, optional) profile. The profile must be provided in a <code>.pfl</code> file located in the <code>Astrodata</code> directory. • 4: User defined P (and T, optional) profile. The profile must be provided in a <code>.pfl</code> file located in the <code>Astrodata</code> directory.
<code>fprofil</code>	n_H-T or P-T profile. If <code>ifisob</code> is 1 or 4, the code reads a <code>.pfl</code> density-temperature or pressure-temperature profile. The name of this file must be provided in this variable.
<code>presse</code>	Thermal gas pressure. Used only for isobaric models (<code>ifisob</code> = 2). For isochoric models, thermal pressure is computed by the code. Typical values: 10^3 to 10^8 K cm⁻³.
<code>vturb</code>	Turbulent velocity. This parameter affects line broadening. Typical value: 2 km s⁻¹.
<code>itrfer</code>	UV radiative transfer method. This flag can take several values: • 0: FGK approximation for H2 self-shielding • 1: Exact computation of self- and mutual shielding for H and H₂. • 3: Exact computation of self- and mutual shielding for H, H₂ and ¹²CO. • 4: Exact computation of self- and mutual shielding for H, H₂, ¹²CO and ¹³CO. • 5: Exact computation of self- and mutual shielding for H, H₂, ¹²CO, ¹³CO, C¹⁸O and HD.
<code>jfgkh2</code>	Rotational H2 quantum number (J) below which exact UV radiative transfer is solved in the UV lines of H₂ (if <code>itrfer</code>>0).
<code>ichh2</code>	H₂-H collision rates model (default 0).
<code>los_ext</code>	Extinction curve . Used to select a Fitzpatrick and Massa grain extinction curve. Available values are listed in the file <code>line_of_sight.dat</code> located in <code>Astrodata</code>. Typical value: Galaxy
<code>rrr</code>	RV = AV / E(B-V). Total to selective extinction ratio. Typical value for the Galaxy: 3.1 If specific extinction curve is used, R_V should be modified accordingly. File <code>line_of_sight.dat</code> located in <code>Astrodata</code> provides some values.
<code>metal</code>	Metallicity, Z. Grains and elementary abundances are scaled by Z. Typical values: Z = 1.
<code>cdunit_0</code>	N(H) / E(B-V), ratio of proton column density on reddening for a metallicity Z = 1. This value is divided by metallicity in the code. Typical value for the Galaxy: $5.8 \cdot 10^{21}$ - Bohlin et al. (1978)
<code>gratio_0</code>	Dust to gas mass ratio for Z = 1. This value is multiplied by metallicity in the code. Typical value: 0.01.
<code>q_pah</code>	PAH to dust mass ratio for Z = 1. This value is multiplied by metallicity in the code. Typical value: 4.6E-2
<code>alpgr</code>	Index of grain size distribution. Grain distribution follows a MRN-like power law. This parameter allows to modify the slope of the power law. Typical value: 3.5 according to Mathis, Rumpl & Nordsieck (1977)
<code>rgrmin</code>	Minimum radius of grains (excluding PAHs)

rgrmax	Typical value: 10^{-7} cm. Maximum radius of grains (excluding PAHs)
F_DUST_P	Typical value: 3×10^{-5} cm. Grain model . Flag to activate or not the coupling of the PDR code with DustEM. • 0: PDR grain model • 1: DustEM grains model
iforh2	H ₂ excitation formation model. Default value is 0.
istic	H sticking on grains model. Default value is 4.
F_W_ALL_IFAF	Flag : write outputs at all iterations. Default value is 0 (write only at the two last iterations)

Table 3.3: List of input files. The two most important ones are pdr.in and *.chi files.

Outputs and results analysis

4.1 Output files

After a run, several output files are produced and stored in the `out` directory. The list of all output files can be found in Appendix A. Output filenames are the model name provided in `pdr.in` plus extensions.

The most important output files have a `.hdf5` extension: They contain all quantities computed by the code and must be read with dedicated python tools located in the `AnalysisTools` directory. Two tools are provided: `IDAT` and `ChemistryAnalyser`.

- `_s_XX.hdf5`: main output files that contain all profiles as a function of position (densities, temperatures, level populations) and many other results such as line intensities. These files must be opened with the `IDAT` tool.
- `_c_XX.hdf5`: output file containing informations about chemical reaction rates at each position in the cloud. These files allow to analyse the chemistry and understand the formation and destruction routes at each position. They must be opened with the `ChemistryAnalyser` tool.
- `_a_XX.hdf5`: annex output files containing line emissivities and other related quantities (optical depth, escape probabilities) at each positions. Quantities stored in these files are only useful for specific problems. These files must be opened with the `IDAT` tool.

XX in filenames correspond to the index of the global iteration. By default, the code writes output for the last and second last iterations (see `ifafm` parameter in `pdr.in`).

4.2 Result extraction and visualization with IDAT

To extract quantities from `_s_XX.hdf5` and `_a_XX.hdf5` files, you will need to run the `IDAT` tool provided in the `AnalysisTools/IDAT` directory. This folder also contains a complete user guide for this tool. We will thus provide here only basic instructions for a minimal usage of `IDAT`.

1. To install all the necessary python modules, run the script `install_ismpy.sh` located in `AnalysisTools/IDAT/scripts`.
2. To start the interactive interface of `IDAT` with the command `./idat da` from the `AnalysisTools/IDAT` directory.
3. Choose an HDF5 file to load.
4. You can then explore the list of physical quantities contained in the HDF5 file, extract some of them into ASCII data files, or interactively plot the spatial profile of any physical quantity in the HDF5 file.

The interface is presented in Fig. 4.1

IDAT: Data analysis

Local version

Opened model: NGC70J18_BestMod_s_20.hdf5

This tool allows to **extract data** produced by the ISM platform codes (as text files) and to **plot the results**.

- Search among the computed quantities with the search bar or in the data tree listing all available quantities.
- The quantities you select will appear in the boxes of section 2.
- If you already know the quantity names, you can also directly type or paste them into the boxes.
- Export as text files or plot the data with our interactive plot tool using the corresponding buttons.

1. Select quantities to extract or plot

Find quantities computed by the code with the search bar

Find quantities in the data tree

- Integrated quantities
- Local quantities
 - Auxiliary
 - Densities
 - Dust
 - Gas state
 - Positions
 - Parameters

Ionisation degree ☐ nH ☐ Pressure ☐ Temperature ☐ Total density

2. List of selected quantities

Extract the chosen quantities as datafiles or plot them with our interactive plot tool.

Direct cut/paste into this box is possible.

The selected quantities are divided between local quantities (functions of the position) on the left, and global quantities (taking one single value for the whole model) such as integrated quantities and model parameters on the right.

Local quantities (position dependant)

nH
Temperature

Actions

Export as ASCII file
Export as VOTable
Plot

Integrated quantities or parameters

Actions

Export as ASCII file
Export as VOTable

3. Plot selected quantities

graph

Title (text fields accept LaTeX expressions enclosed in \$ \$)

x axis

Quantity Distance Label d [cm] log ☐

Min Max

Min auto ☒

y1 axis

Label n_H [cm⁻³] log ☒ Min Max

☒ n_H nH
☐ Temperature K Temperature

y2 axis

Label T_{gas} [K] log ☐ Min Max

☐ nH nH
☒ T K Temperature

Plot

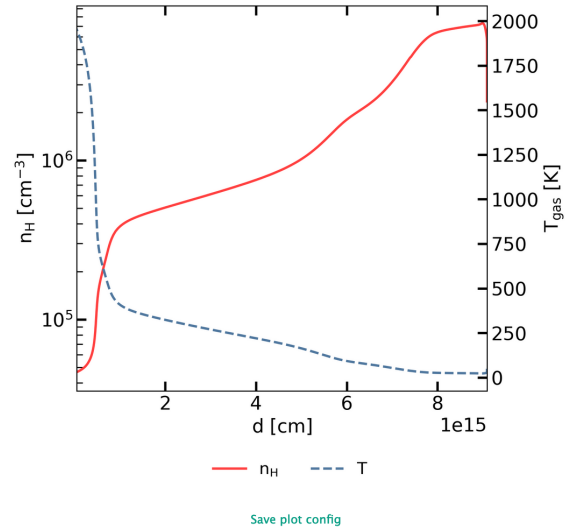


Figure 4.1: Interface of the IDAT tool.

4.3 Chemistry analysis

It is often useful to identify what are the most important chemical reactions leading to the formation and to the destruction of chemical species. This can be done easily at any position in the cloud using the chemistry analyzer python tool (located in `AnalysisTools/chemistry\_analyser`) to open the `\_c\_XX.hdf5` files. A user guide can also be found in this directory. We will thus provide here only basic instructions for a minimal usage of the Chemistry Analyser. Note that `ismpy` should be installed (see above).

How to run the chemistry analyzer tool:

- 1) copy a `\_c\_XX.hdf5` file in the directory: `AnalysisTools/chemistry\_analyser/data`
- 2) run the chemistry analyser tool typing in a terminal from the `AnalysisTools/chemistry\_analyser/server` directory: `~/ismpy/python/python3 server.py`
- 3) open the file `AnalysisTools/chemistry\_analyser/visualizer/index.html` double clicking on it. This will open your default web browser with the Chemistry Analyser interface.

The interface is presented in Fig. 4.2. The chemistry analyzer tool allows to select a position in the cloud as well as chemical species. It will then provide present the formation and destruction reactions. Detailed documentation of the tool can be found in the `chemistry\_analyser` directory.

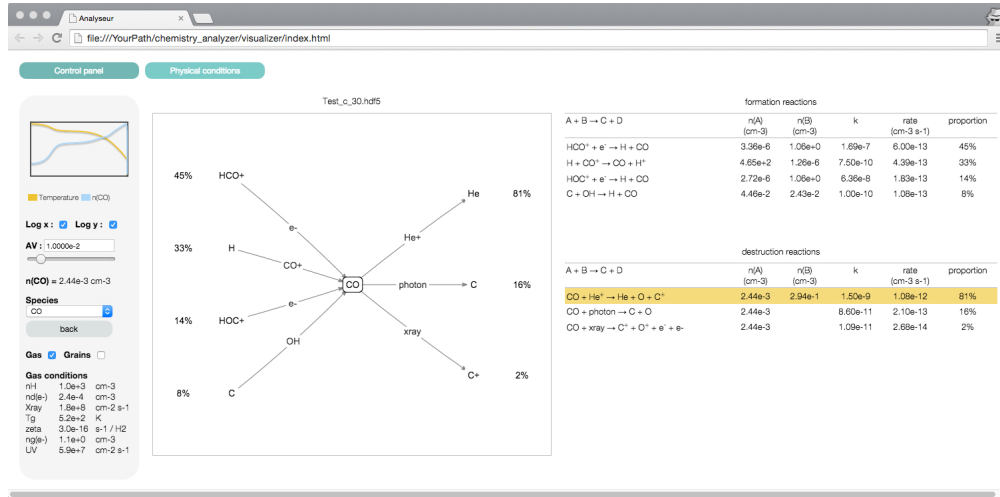


Figure 4.2: Interface of the Chemistry Analyser tool.

4.4 Spherical clouds

The Meudon PDR code is a plane-parallel model. This can be a bit restrictive to interpret observations in edge-on PDRs or objects with spherical geometries. A new features of the post-processor code, `prep`, is to wrap the structure to simulate a spherical cloud. This assumes that the radiation field illuminating the sphere is uniform. It is not possible to simulate a spherical cloud near a star (this is a 2D problem).

To use this facility, run a model with the same radiation field on both sides of the cloud then with the `prep` post-processor ask for line intensities in spherical conditions. You will have to provide the position of the line-of-sight.

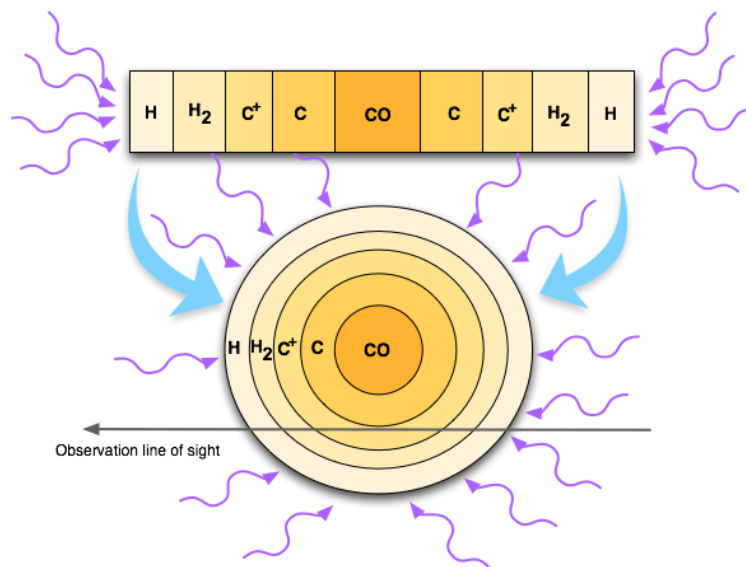


Figure 4.3: wrapping a plane-parallel model in a spherical model.

Known problems

This section discusses known problems with the code. These problems happen in specific domains of parameters where either the numerical algorithms or the physical approximation start breaking down. The problems discussed here are under investigation by the Meudon PDR team. If you notice any other abnormal results, please contact us.

5.1 Atomic cloud core & H₂ formation

In some conditions, the density atomic hydrogen increases at high A_V , sometimes leading to a central core region where hydrogen is mostly atomic. An exemple model exhibiting this behavior is shown on Fig. 5.1.

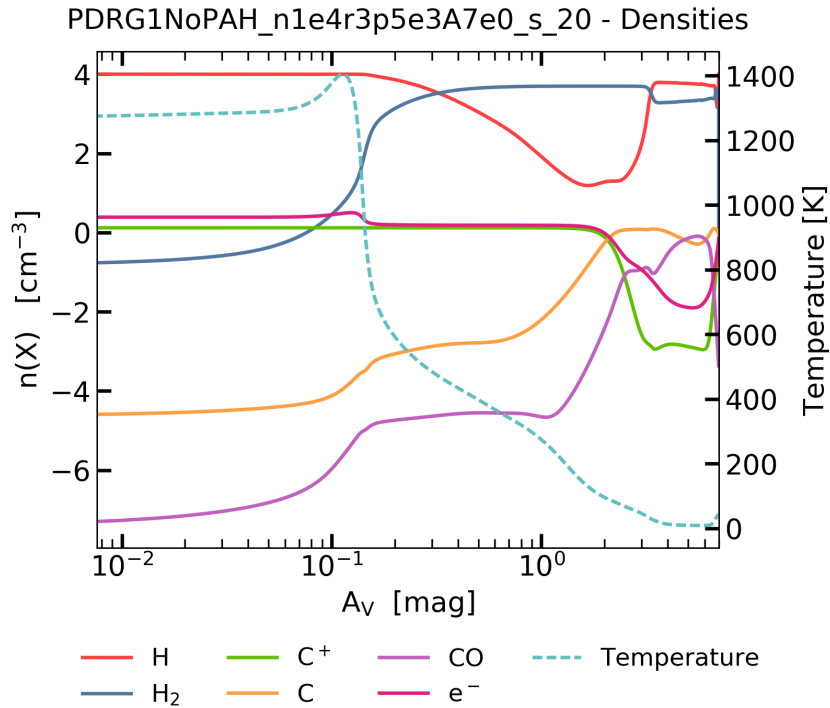


Figure 5.1: Example model exhibiting the atomic core problem for $A_V > 3$. This is a constant density model with $n_H = 10^4 \text{ cm}^{-3}$, $\chi = 3.5 \times 10^3$ and $A_V^{tot} = 7$.

This happens because of the H₂ formation formalism implemented in the code. By default the PDR code, considers Langmuir-Hinshelwood and Eley-Rideal mechanisms. At large A_V , the gas temperature is low and the Eley-Rideal mechanism is not efficient. Indeed, the Eley-Rideal mechanism efficiency depends on the gas temperature: H atoms from the gas phase must have enough kinetic energy to reach chemisorption sites. So, in low gas temperature regions, only the Langmuir-Hinshelwood mechanism should be efficient. Nevertheless, the LH mechanism is efficient only in a narrow range of grain temperatures. Grains may be too warm in the center of the cloud and adsorbed H atoms desorb from the grain before forming H₂. As a result, H₂ destroyed by cosmic-rays and other chemical reactions can not be efficiently replenished and, in the core hydrogen is found in atomic form and not in H₂ form. Grains are often warm in the center in models with high density and strong UV incident UV field. At the edge of the PDR, the UV photons are converted in IR photons that interact with grains in the center of the cloud, raising their temperature.

We consider this to be a consequence of the limitations of our physical approximations to treat the formation of H₂ and not a physical result. Indeed, there is no observation indicating fully atomic regions inside molecular clouds. We are working on this problem. In the meantime, our recommendations are:

- if you are interested in processes that take place at the edge of the PDR: do not bother with this atomic zone. Just make sure that the quantities you extract from the code are not affected by this atomic region.
- if you are interested in processes that takes place in this central region: then, you have no other choice than to de-activate the H₂ formation by Eley-Rideal and Langmuir-Hinshelwood formalisms and to adopt a cruder formalism. As explained below, it is possible to replace these mechanisms by a simple H₂ formation rate equals to $R = 3 \times 10^{-17} \text{ cm}^3 \text{ s}^{-1}$ that is independent of the grain temperature. In that case, at the edge of the PDR, the computed H₂ formation rate will be rough but the atomic core will not appear.

How to force a H₂ formation rate at $R = 3 \times 10^{-17} \text{ cm}^3 \text{ s}^{-1}$

The choice of a formalism for H₂ formation rate is done in the chemistry file. By default the Langmuir-Hinshelwood and Eley-Rideal mechanisms are used and implemented as described in Le Bourlot et al. (2012), A&A. To force a constant formation rate, for example at $3 \times 10^{-17} \text{ cm}^3 \text{ s}^{-1}$, the mean value estimated in the diffuse ISM by Jura et al. (1975) with Copernicus observations or Gry et al. (2002) with FUSE observations:

- 1- Deactivate LH and ER mechanisms by removing the two species h: (physisorbed H) and h:: (chemisorbed H) and all reactions involving them from the chemistry file.
- 2- Add the H₂ formation reaction:

```
OLDST h          h          h2          3.00E-17  0.50  1.00E2  0 !
```

This will replace the Langmuir-Hinshelwood and Eley-Rideal H₂ formation formalism presented in Le Bourlot et al. (2012) by the classical constant efficiency H₂ formation reaction with formation rate

$$\left. \frac{dn(\text{H}_2)}{dt} \right|_{\text{H}_2 \text{ formation}} = 3 \times 10^{-17} \left(\frac{T_{\text{gas}}}{100 \text{ K}} \right)^{0.5} \times n_{\text{H}} n(\text{H}) \times s(T_{\text{gas}})$$

As the Le Bourlot et al. (2012) formalism was mainly added to improve the treatment of H₂ formation in the surface PDR layers, using the old style H₂ formation rate when interested only in the colder, deeper core region is reasonable.

List of output files

Version: PDR 1.5.2

Tables below list the output files produced by the PDR code. Most of them are optional and are produced depending on internal variables in the code. Those variables are defined in the `PXDR_CONSTANTES.f90` file. The code must be recompiled to take into account modifications in these variables.

MIME	Description
<code>.def:</code>	Log file This ASCII file contains a summary of input parameters, chemical network, various flags and runtime of the model
<code>_s_XX.hdf5:</code>	Standard output file This is the main output file of the code. It contains nearly all quantities computed by the code such as density profiles, gas and grain temperature profiles, level excitation as well as column densities and line intensities. This file can be read with the IDAT program provided with the code. IDAT can be use to extract and visualize quantities contained in the HDF5 output files.
<code>_c_XX.hdf5:</code>	Chemistry output file - Chemistry analysis This file can be used to study the chemical network at any position in the cloud. It must be read with the chemistry analyzer Python tool.

Other output files:

These output files may not be produced by the code, depending on some internal options. Most of these files are useful only for debugging purposes, other ones for specific studies.

<code>.binXX:</code>	Output files in binary format This file is not produced by default anymore. To generate it, set the flag <code>F_W_BINARY</code> to 1 in <code>PXDR_CONSTANTES.f90</code> . It contains the same data as the .hdf5 files, written in binary format and can only be read on the computer that produced it, with the same compiler. It is superseded by the .hdf5 files for normal use. * To read it and produce output files in ASCII format, use the prep fortran program located in the src directory. * To read it to analyze chemical networks, use the CHEM_ANALYSER fortran program located in the src directory.
<code>.rf_alb:</code>	Radiation field and albedo This file contains the radiation field, the full albedo (gas + dust contributions) and the full absorption coefficient (gas + dust) at each position in the cloud and at each wavelength. This a voluminous file that is produced only by setting flag <code>F_W_RF_ALB</code> to 1 in <code>PXDR_CONSTANTES.f90</code> . It can be used to extract the spectrum at a given depth in the cloud, or to get the radiative energy density at a specific wavelength as a function of AV. Data are written in a binary format that can only be read on the computer that produced the file. To read this file, use the <code>read_rf_alb_abs.f90</code> program located in the <code>src/OTHER_PROG</code> directory of the PDR code.
<code>.uv:</code>	Cloud transmission This file provides the normalized absorption through the cloud at each wavelength. It can be used to multiply a given background source emission (star or external galaxy) to evaluate the spectrum after absorption

	First column: wavelength, Second column: light attenuation ($\exp(-\tau(\lambda))$) Third column: optical depth ($\tau(\lambda)$).
<code>.flin:</code>	External radiation field This file is mainly useful for debugging purposes, and should not be used in standard exploitation. It provides some informations on the external radiation field impinging the cloud. By default the incoming radiation field is isotropic, but it may include an additional normal component. This radiation field is coming from either side of the cloud, but most of the time from both (single side, although possible, is not recommended). Columns are as follow: 1 : Wavelength (Angstrom) 2 : Perpendicular contribution to left side (specific intensity) 3 : Isotropic contribution to left side (specific intensity) 4 : Perpendicular contribution to right side (specific intensity) 5 : Isotropic contribution to right side (specific intensity) 6-9 : Internal usage (do not use)
<code>.Iesc:</code>	Emission spectra This file give the specific intensity radiated by the cloud as a function of direction. The radiation field is in $\text{erg cm}^{-2} \text{s}^{-1} \text{sr}^{-1} \text{Ang}^{-1}$, and is given at all directions used for the boundary conditions of the radiative transfer part of the code. These are 0 of a Legendre polynomial of degree $L+1$, where L is the order of the approximation used. The relevant angles (in degrees) are given in the header. The first column gives the wavelength in Angstrom, followed by $2 * (L+1)$ values of the specific intensities at both surfaces of the cloud: 1 : Wavelength (Angstrom) $2-(L+1)/2+1$: Outward going I on left side $(L+1)/2+2 - (L+2)$: Incoming radiation field (boundary condition) on the left $(L+3) - 3*(L+1)/2+1$: Incoming radiation field (boundary condition) on the right $3*(L+1)/2 + 2 - 2*(L+1) + 1$: Outward going I on the right The boundary conditions are built from the data written in the previous file (<code>.flin</code>) through a rather complex condition insuring integrated energy conservation.
<code>.extinct</code>	Extinction curve This file provides the extinction curves used in the code to solve for radiative transfer and to convert from AV to size (in cm). It is computed from the internal values of the absorption and scattering coefficients at all wavelengths. These may be derived from 3 different sources: * From an "Fitzpatrick & Massa" fir to an observed curve, via the use of an ad hoc prescribed albedo * From the grain model of Draine * From data computed by the DUSTEM model of interstellar grain, directly coupled to the PDR code. Columns are: 1 : $1 / \lambda$ (μm^{-1}) 2 : $(k(\lambda) - k(V)) / (k(B) - k(V))$, where $k = \kappa + \sigma$, and B and V are standard photometric bands 3 : Extinction curve

Table A.1: List of output files