ISMServices

DOCUMENTATION
Extractor

N. MOREAU, B. GODARD, F. LE PETIT

February 24, 2016

1 Presentation . . . 3
2 Overview . . . 5
2.1 Installation . . . 5
2.2 Graphical user interface . . . 5
2.3 Opening a HDF5 file . . . 5
2.4 Extraction of data . . . 5
3 Tree structure . . . 7
3.1 Local quantities . . . 7
3.2 Integrated quantities . . . 7
4 Scripts . . . 9
4.1 Command line interface . . . 9
4.2 Build a script . . . 9
4.3 Apply a script . . . 9
5 VO Interoperability . . . 11
5.1 Communication with VO tools . . . 11

Presentation

A single run of the Meudon PDR code provides a large number of physical quantities that fully describe the chemical and thermal state of the simulated interstellar cloud: the temperatures of gas and dust particles, the abundances of hundreds of species, the populations of hundreds of excited states, the heating and cooling rates due to several processes, the emissivities of thousands of lines, and so on. With the PDRLight_x.x versions of the code, these results and their metadata are stored in a single HDF5 file.

The Extractor tool is a software designed to open the HDF5 file produced by the code and to facilite the navigation among the outputs stored. The Extractor presents in a graphical user interface all the quantities computed by the code with the information necessary to understand their meaning (description, units, ...). Any quantity or group of quantities can then be selected and extracted in an ASCII file or a VO-Table.

The tool also supports a script mode to facilitate the extraction of quantities from a large number of models. It has been developed using Virtual Observatory standards ; any data can thus be sent to VO softwares. For most of the plots, we recommend to use the Extractor tool in combination with TOPCAT. Because it can produce ASCII files, any other graphing utility software can be used to plot the data.

Overview

2.1 Installation

The Extractor requires python 2.7 and the h5py, numpy, and PyQt4 librairies. To install these

- on linux distributions (e.g. ubuntu, debian), we recommend to use the `apt-get` software

```
sudo apt-get install python-h5py
sudo apt-get install python-numpy
sudo apt-get instlal python-qwt5-qt4
```

- on Mac OS, we recommend `macports` (https://www.macports.org/install.php)

```
sudo port selfupdate
sudo port install python27 py27-numpy py27-h5py py27-pyqt4
```

2.2 Graphical user interface

To start the GUI from a terminal window, go in the directory where the Extractor is located and type

```
python2.7 extractor.py
```

2.3 Opening a HDF5 file

To open the result of a PDR simulation (e.g. those found on the PDR database), select the menu

```
File > Open HDF5 > Model.hdf5
```

2.4 Extraction of data

The results of the code can be understood as a serie of tables, i.e. as quantities that vary as functions of other quantities (e.g. visual extinction, grain size, ...). The Extractor presents all the

data stored in the HDF5 file as a tree with several branches and sub-branches (Fig. 2.1). To select any data, the user can

- either type the name of the quantity in the search bar and click on the `confirm` button
- or explore the tree structure and directly click on the quantity to extract.

The data selected appear on the right side panel (Fig. 2.1). Those data can then be exported either in an ASCII file with the `Export as Text` button or as a VO table with the `Export as VO Table` button. Only compatible quantities can be extracted. For instance, it is impossible to simultaneously extract abundances, which are functions of position, and the size of the cloud, which is parameters of the simulation.

For an extraction in an ASCII file, the user can also choose the column separator with the menu

```
Output > Separator
```

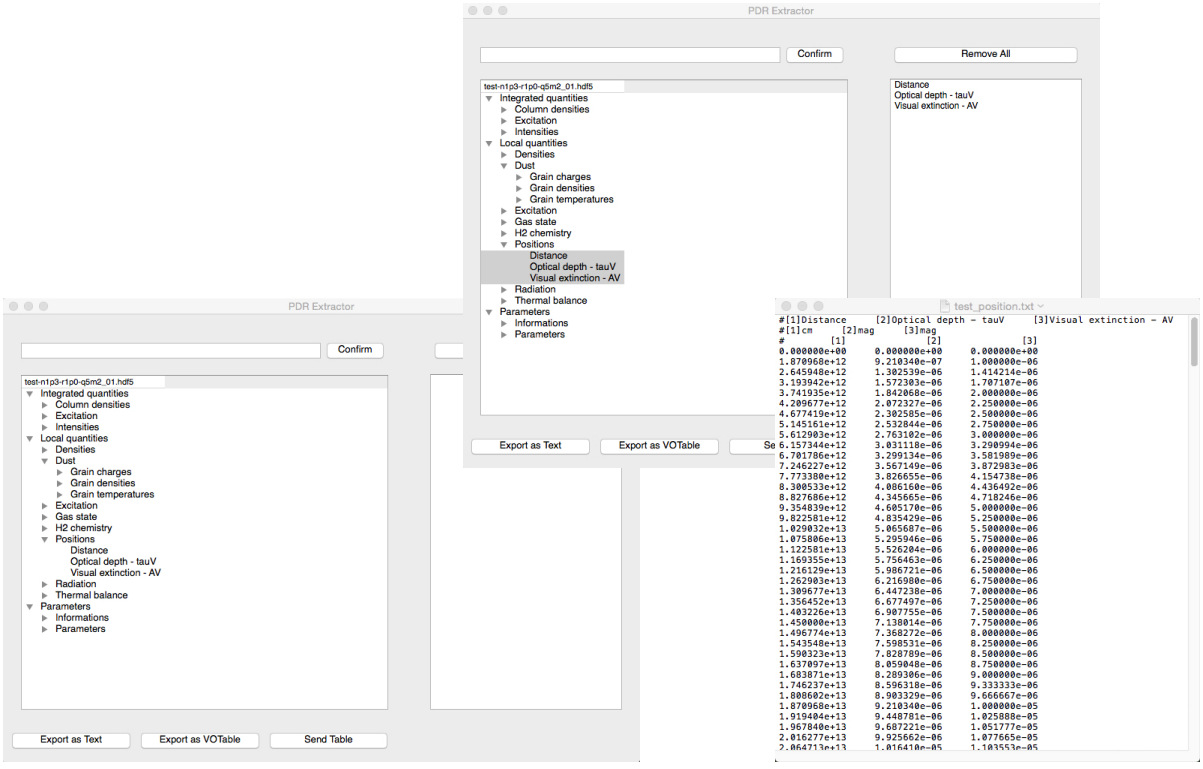

Figure 2.1: Exploration of the HDF5 tree structure and extraction of data in an ASCII file.

Tree structure

The Extractor displays all the branches, sub-branches, and datasets in alphabetical order. The data contained in a HDF5 file can be organized in three different categories.

- `Integrated quantities`: contains physical quantities integrated over the entire cloud (e.g. column densities, line intensities) which can be directly compared to the observations.
- `Local quantities`: contains local variables, i.e. physical quantities (e.g. chemical abundances, temperature) that vary as functions of the distance from the ionization front.
- `Parameters`: contains the main parameters and constant quantities used in the simulation, including the sizes of the grains, the elemental abundances, and the isotopic ratios.

3.1 Local quantities

Distance from the ionization front

All local quantities are computed at different positions in the cloud. The list of positions, i.e. the tables containing the distance from the ionization front (expressed either in cm or in units of optical depth or visual extinction) can be found in

`Local quantities / Positions`

Level populations and emissivities

The PDR code computes the populations of the excited levels of several species – including the major coolants of the ISM – and the emissivities of thousands of lines. All these data are in

`Local quantities / Excitation`

Note : the code estimates itself if a line has to be taken into account or not. For instance, it computes more levels and lines for hot PDRs than for diffuse clouds, because the excitation conditions are stronger.

3.2 Integrated quantities

The column densities of chemical species and of their excited states are computed along a direction perpendicular to the cloud. A simple saling can be done to infer these quantities for different inclination angles.

In contrast, integrated intensities are computed and given for a finite number of inclination angles: 0, 10, 20, 30, 40, and 60°. The Meudon PDR code assumes a plan parallel geometry. We thus consider that intensities computed for an angle above 60° should not be used because those are integrated over path lengths that tend towards infinity when the the angle tends towards 90°.

Scripts

The Extractor includes a script mode and a command line interface which allow to rapidly extract the results of a large number of simulations in ASCII files without using the graphical user interface.

4.1 Command line interface

The command line interface includes several options which can be summarized as follows

`-h`	`--help`	display help
`-f`	`--file`	HDF5 file to read
`-t`	`--template`	script to apply (.esf file)
`-o`	`--output`	name of output file
`-s`	`--separator`	separator between columns

4.2 Build a script

The scripts used by the Extractor are text files which simply contains the names of all the quantities that the user want to extract. Such scripts can be edited manually or with the graphical user interface (once the data have been selected) through the menu

```
Script > Save Script
```

The name of the script can be chosen by the user. However, this name must explicitly carry a ".esf" extension in order to be recognized as a script by the graphical user interface.

4.3 Apply a script

To use a script with the graphical user interface, select the menu

```
Script > Apply Script
```

To use a script from a terminal window, go in the root directory where the Extractor is located and type the following instructions

```
python2.7 extractor.py -f PATH1/model.hdf5 -t PATH2/script.esf
                       -o PATH3/result.txt -s "   "
```

The data listed in the script `PATH2/script.esf` will be extracted from the HDF5 model `PATH1/model.hdf5` and written in the text file `PATH3/result.txt`, where the columns will be separated by 3 spaces.

VO Interoperability

5.1 Communication with VO tools

By using a SAMP hub, the Extractor can send data directly to VO tools (e.g. TOPCAT). To activate the connection between a VO tool which supports SAMP and the Extractor, follow this procedure

1. Run the Extractor and open a model.
2. Run the VO plotting tool and open a SAMP hub (for instance, in TOPCAT, select the `Interop > SAMP status > Attempt to connect to SAMP hub` menu).
3. Connect the VO plotting tool with SAMP if not done automatically (for instance, in TOPCAT, select the `Interop > SAMP status > Connect > Register with hub` menu).

Once the two applications communicate with each other, data can be send from the Extractor to the VO tool with the `Send Table` button (Fig. 5.1).

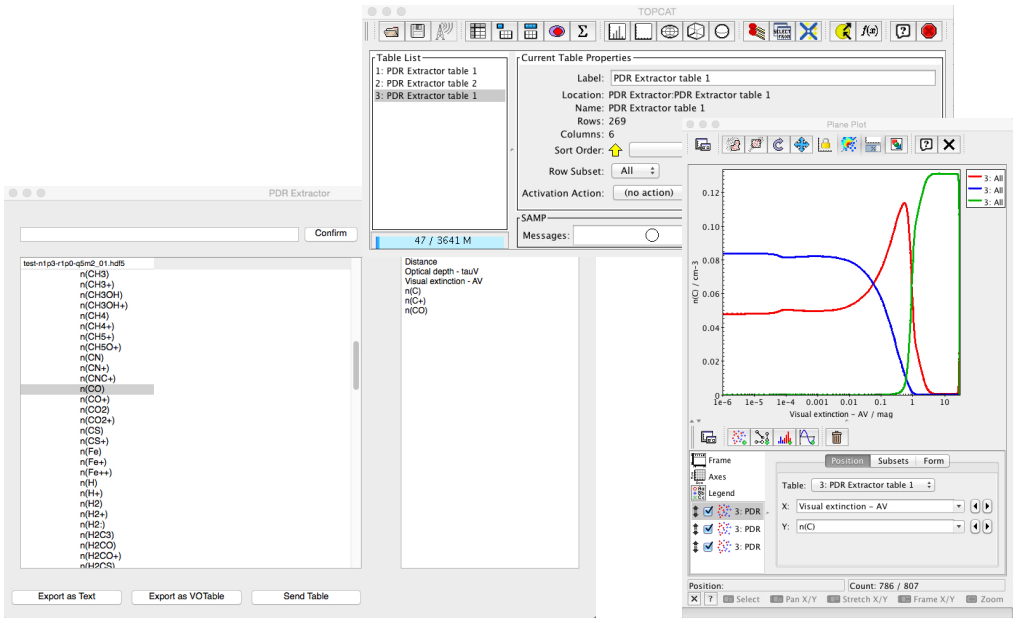

Figure 5.1: Example of a plot quickly produced by combining the Extractor and TOPCAT softwares.