

TPCI - quick start

Michal Salz

January 16, 2015

1 Introduction

The PLUTO CLOUDY Interface (TPCI) is a interface between the two simulation codes PLUTO¹ and CLOUDY². It can be used to simulate hydrodynamic evolution under irradiation from a source. For more information see (paper).

WARNING! Any user of TPCI should be familiar with PLUTO and CLOUDY individually. Both codes provide test problems, which are also included in this package. These problems are described in the userguides PLUTO/Doc/userguide.pdf and TPCI/c13.01/docs/hazy2_c13.pdf.

2 Terms of use

TPCI is distributed freely under the GNU general public license. Users should reference the PLUTO and CLOUDY codes (see the web pages). Additionally the method paper should be referenced (paper).

3 Getting started

The download package contains a packed version of PLUTO v. 4.0 and Cloudy 13.01. Minor changes were applied to both source codes, so the Interface only runs with the provided versions of PLUTO and CLOUDY. In the directory `test_prob/`, the package further contains:

- the interface `call_cloudy.cpp`
- the `local_make` file
- versions of the local makefile for `g++` and `mpic++`
- two shell scripts to apply changes to the `makefile` (see below)
- `earth_spec.ini` a solar EUV spectrum
- the `main.cpp` replacement of the standard PLUTO `main.c` file
- PLUTO start files: `definitions.h`, `init.c`, `pluto.ini`
- `userdef_output.c` a changed PLUTO file needed to run TPCI

The start files are initialized to run the TPCI test problem, which you should be able to do after the following steps.

3.1 Compile CLOUDY

Go into the directory `c13.01/source/` and compile CLOUDY (type `<make>`). In case of any problems see the StepByStep instruction on (<http://www.nublado.org/>). Compiling CLOUDY takes a couple of minutes, but less than one hour.

Check if the code runs correctly: Go to the directory `c13.01/` and run the test problem `<source/cloudy.exe -r test>`. Open the `test.out` file and see, if the last line says that CLOUDY exited OK. If so, you are good to go. (For a complete test you have to run the TestSuite, also described on the CLOUDY web page.)

¹<http://plutocode.ph.unito.it/>

²<http://www.nublado.org/>

3.2 Export the paths

Export the paths to the source files. For the bash:

```
export PLUTO_DIR=../TPCI/PLUTO/  
export CLOUDY_DIR=../TPCI/c13.01/source/
```

For the dots please insert the full path to the TPCI directory.

3.3 PLUTO setup

Go in the `test_prob/` directory. Run the Python PLUTO setup script:

```
python $PLUTO_DIR/setup.py
```

and select the setup of PLUTO. The start screen of the setup looks as follows:

```
Working dir: ../TPCI  
PLUTO dir  : ../TPCI/PLUTO/
```

```
Setup problem  
Change makefile  
Auto-update  
Save Setup  
Quit
```

Items are selected with the arrow keys. To set up the first test problem, just hit `enter` until you reach the start screen again. Go one down and select a makefile, e.g., `Linux.gcc.defs`. Then quit the setup.

You now have to edit the `makefile`, because the interface must be compiled with the `g++` compiler, while PLUTO is compiled with a `c`-compiler. You can try to run the `INS.MAKE.g++.sh` shell script. If that does not work, run the Python PLUTO setup again and subsequently apply the changes manually. In the `makefile` change this:

```
# -----  
#   PLUTO target rule  
# -----
```

```
pluto: $(OBJ)  
$(CC) $(OBJ) $(LDFLAGS) -o $@
```

to this:

```
# -----  
#   PLUTO target rule  
# -----
```

```
pluto: $(OBJ)  
g++ $(OBJ) $(CLOUDY_OBJ) $(LDFLAGS) -o $@
```

3.4 Make PLUTO

You are ready to compile the interface. Type `make` inside the `test_prob/` directory. This should be pretty fast (less than one minute).

3.5 Run the test problem

Execute `./pluto`. TPCI needs about 20 minutes to run the problem, depending on your workstation.

The test problem is a simple hydrogen only simulation of a density clump irradiated at a distance of 0.001 AU, so just above the solar surface. This is not intended as a physically correct setup. The spectral energy distribution is defined in the file `earth_spec.ini`, a solar EUV spectrum extended with a black body. The test case will create 50 tabulated output files for PLUTO and about 10 for CLOUDY. They can be displayed as the individual userguides describe. E.g., the PLUTO output can be viewed with `gnuplot`:

```
plot "data.0000.tab" u 1:3
```

gives the density (4: velocity, 5: pressure, 6: temperature (K), 7: mean molecular weight, 8: heating/cooling (erg cm⁻³ s⁻¹), 9: radiative acceleration (cm s⁻¹), 10: heating efficiency, 11: electron density (cm⁻³), 12: hydrodynamic timescale (s), 13: hydrogen recombination timescale (s), 14: molecular timescale (s), 13: time). Density, velocity, pressure and time are in dimensionless form and the unit variables defined in `init.c` can be used to convert them (see the PLUTO userguide).

The evolution of the test problem shows, that the density clump is heated and expands. It is pushed away from the source due to slightly higher pressure at the source facing boundary and due to radiation pressure. The density is on the order of 10⁹ cm⁻³ and the temperature 10⁴ K.

4 Setup an own problem

To set up your own problem, copy the `test_prob/` directory to anywhere on your system. Basically everything can be changed in the test problem (density, pressure, irradiation, composition), but one has to be careful to choose plausible start parameters. Any changes of PLUTO are done in the usual way, see the userguide. Adapting the CLOUDY setup is done in the file `call_cloudy.cpp` in the function `CloudyInputScript` just as described by the userguide Hazy2 of CLOUDY in the section "Example Call as a Subroutine".

Restrictions for the interface:

- PLUTO grid: Cartesian 1D-3D or 1D spherical
- CLOUDY: do not use any CLOUDY commands changing the density, pressure, or temperature handling (e.g., `constant density`, `constant pressure`, `constant temperature`).

CLOUDY print frequency, the use of the advective solver, and the `change_factor` are adapted in the header of `call_cloudy.cpp`. The use of CLOUDY can also be turned off, then PLUTO runs independently.

To run your test problem you have to redo the steps 2.2 to 2.5. Step 2.1 does not have to be repeated; the CLOUDY source files are compiled only once.

5 MPI runs

TBD

6 Questions?

For questions: msalz@hs.uni-hamburg.de. Good luck.