CosmoPMC — a short guide

Versions

This short guide corresponds to CosmoPMC version 1.2, and pmclib version 1.01.

Download and install pmclib

The package pmclib can be downloaded from the CosmoPMC site http://www.cosmopmc.info.

After downloading, unpack the gzipped tar archive

```
> tar xzf pmclib_x.y.tar.gz
```

This creates the pmclib root directory pmclib_x.y. pmclib uses waf[1] instead of configure/make to compile and build the software. Change to that directory and type

```
> ./waf --local configure
```

See ./waf --help for options. The packages lua, hdf5 and lapack are optionally linked with pmclib but are not necessary to run CosmoPMC. Corresponding warnings of missing files can be ignored. Instead of a local installation (indicated by --local), a install prefix can be specified with --prefix=PREFIX (default /usr/local).

Patch pmclib

For CosmoPMC v1.2 and pmclib v1.x, a patch of the latter is necessary. From http://www.cosmopmc.info , download patch_pmclib_1.x_1.2.tar.gz and follow the instructions in the readme file readme_patch_pmclib_1.x_1.2.txt.

Download and install CosmoPMC

The newest version of CosmoPMC can be downloaded from the site http://www.cosmopmc.info.

First, unpack the gzipped tar archive

```
> tar xzf CosmoPMC_v1.2.tar.gz
```

This creates the the CosmoPMC root directory CosmoPMC_v1.2. Change to that directory and run

```
> [python] ./configure.py
```

This (poor man's) configure script copies the file Makefile.no_host to Makefile.host and sets host-specific variables and flags as given by the command-line arguments. For a complete list, see 'configure.py --help'.

[1] http://code.google.com/p/waf

Alternatively, you can copy by hand the file Makefile.no_host to Makefile.host and edit it. If the flags in this file are not sufficient to successfully compile the code, you can add more flags by rerunning configure.py, or by manually editing Makefile.main. Note that a flag in Makefile.main is overwritten if the same flag is present in Makefile.host.

To compile the code, run

```
> make; make clean
```

On success, symbolic links to the binary executables (in ./exec) will be set in ./bin.

It is convenient to define the environment variable COSMOPMC and to set it to the main CosmoPMC directory. For example, in the C-shell:

```
> setenv COSMOPMC /path/to/CosmoPMC_v1.2
```

This command can be placed into the startup file (e.g. ~/.cshrc for the C-shell). One can also add $COSMOPMC/bin to the PATH environment variable.

Quick reference guide

Examples

To get familiar with CosmoPMC, use the examples which are contained in the package. Simply change to one of the subdirectories in $COSMOPMC/Demo/MC_Demo and proceed on to the point **Run** below.

User-defined runs

To run different likelihood combinations, or your own data, the following two steps are necessary to set up a CosmoPMC run.

1. **Data and parameter files**

   Create new directory with newdir_pmc.sh. When asked, enter the likelihood/data type. More than one type can be chosen by adding the corresponding (bit-coded) type id's. Symbolic links to corresponding files in $COSMOPMC/data are set, and parameter files from $COSMOPMC/par_files are copied to the new directory on request.

   If necessary, copy different or additional data and/or parameter files to the present directory.

2. **Configuration file**

   Create the PMC configuration file config_pmc. Examples for existing data modules can be found in $COSMOPMC/Demo/MC_Demo.

   In some cases, information about the galaxy redshift distribution(s) have to be provided, and the corresponding files copied (see $COSMOPMC/Demo for example files 'nofz*').

Run

Type

```
> $COSMOPMC/bin/cosmo_pmc.pl -n NCPU
```

to run CosmoPMC on NCPU CPUs. See '`cosmo_pmc.pl -h`' for more options. Depending on the type of initial proposal, a maximum-search is started followed by a Fisher matrix calculation. After that, PMC is started. Fig. 1 shows a flow chart of the script's actions.

Diagnostics

Check the files `perplexity` and `enc`. If the perplexity reaches values of 0.8 or larger, and if the effective number of components (ENC) is not smaller than 1.5, the posterior has very likely been explored sufficiently. Those and other files are updated during run-time and can be monitored while PMC is running.

Results

The text file `iter_{niter-1}/mean` contains mean and confidence levels. The file `iter_{niter-1}/all_contour2d.pdf` shows the 1d- and 2d-marginals. Plots can be redone or refined, or created from other than the last iteration with `plot_contour2d.pl`. Note that in the default setting, the posterior plots are not smoothed. or more details, and for information on the alternative script `plot_confidence.R`.

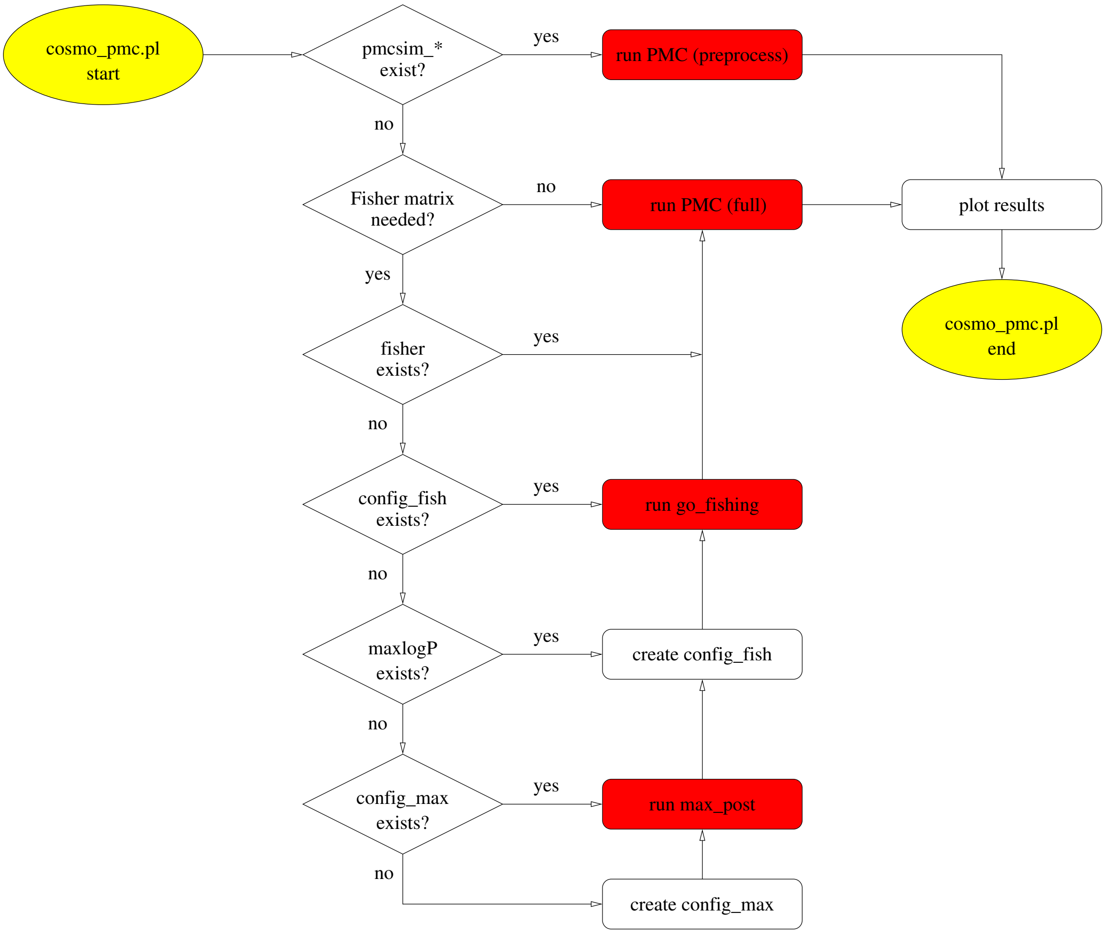

Figure 1: Flow chart for cosmo_pmc.pl.