

Tutorials for Modeling Sessions

This document provides step-by-step guide on how to set up fitting programs used in the hands-on sessions.

l) sfit

This program uses multi-dimensional Newton's method to fit data from multiple sites to a single-lens microlensing model. The following offers instructions on a simple single-lens fitting procedure without exhausting all features of the code.

(0) The following sub-directories are under the directory *sfit*:

code_sfit: Fortran source code
input : configuration files that feed in input parameters to the program
output : error and debug log files
plot : files related to generating light curve and model plots
data : photometric data files

The default working directory is assumed to be *sfit* in the following discussion.

(1) Preparing data

Photometric data files can be copied to the *./data* directory. (The first three columns in the data file have to 1. time (usually in HJD' = HJD - 2450000), 2. magnitude, and 3. error in magnitude.)

Next the filename and nickname should be specified in the *./input/data.ctrl* file. If one has more than one data set to model, one can just add more lines in the *./input/data.ctrl* file.

Optional parameters for *data.ctrl* file are described in the *data.ctrl.info* file.

(2) Adjusting input parameters

(a) Modify the initial values for all parameters in *./input/parm.init*. The three columns are parameter names (*t0*, *u0*, *tE* and *rho*), flags that control whether the parameters are fixed during the run (0: free, 1: fixed) and their initial values. Note that for point-lens point-source fit, one usually keep *rho* fixed at a small value (e.g., 0.0001).

(b) There are two configurations files in the *./input* directory that affect the plot generation. One can specify the event name in the *./input/event.ctrl* (please ignore other options in the file, which are related to microlens parallax fitting, which is disabled in the present version). In the *./input/modeldat.ctrl*, one may specify the time range and step size (in the units of day) for which the microlensing model plot is generated. One can specify several segments with various step

sizes.

(3) To compile the code, issue the following command in the *sfit* directory:

```
make
```

This will generate an executable called *sfit* in the current directory.

There are two modes of operations -- interactive (default) and automatic.

a) Simply execute

```
./sfit
```

will enter interactive mode, and user will be prompted to enter the scaling factor (<1) for suggested step sizes and number of iterations until a successful fit is obtained. The current fitting parameters (besides t_0 , u_0 , t_E , ρ , the linear limb darkening parameters, source and blend fluxes for each observatory are also displayed; please ignore F_{see} , which is not activated in this version), suggested step sizes, calculated error bars, X^2 for each observatory and total X^2 are displayed each time a set of iterations are done. A useful strategy to do the fitting is to enter a small scaling factor first (such as 0.01), try a few iterations until the suggested step sizes become reasonably small to try a larger scaling factor (such as 0.1, 1.0). Enter 0 for the iteration number to stop the program.

b) For automatic mode, run

```
./sfit a
```

In the automatic mode, the program automatically makes these choices.

(4) After a successful fit, one can generate a plot with data and the best-fit model curve. To do so, first do *cd ./plot* and then specify boundaries for the plot (the minimum, maximum values for the HJD time, magnitude, residuals) in the *sm.ctrl* file. Execute

```
make lc
```

to generate a postscript plot *plotlc.eps*, which can be viewed by *gv (gv plotlc.eps &)*. Adjust *sm.ctrl* and execute *make lc* again to generate plots with a different setup.

Detailed instructions of the single-lens and binary-lens MCMC programs can found in the “ulens mcmc Manual”. The following gives the step-by-step instructions.

II) single_lens_mcmc

(0) The single-lens MCMC fitting program is located at `~/mcmc/single_lens_mcmc/`. The sub-directory structure is the same with that of the *sfit* program:

code_single_lens: source code
input : configuration files that feed in input parameters to the program
output : file that contains best-fit parameter, MCMC chain files
plot : files related to generating light curve and model plots
data : photometric data files

(1) Data preparation

This step is the same with *sfit*. Photometric data files can be copied to the `./data` directory. (The first three columns in the data file have to 1. time (in HJD' = HJD - 2450000), 2. magnitude, and 3. error in magnitude.)

Next the filename and nickname should be specified in the `./input/data.ctrl` file. If one has more than one data sets to model, one can just add more lines in the `./input/data.ctrl` file.

Optional parameters for `data.ctrl` file are described in the `data.ctrl.info` file.

(2) Parameter setup

(a) Modify `./input/mcmc.init` file to input the initial parameters (column 3), the allowed limits (column 4 and 5) they may vary within and initial step sizes (column 6) for a given parameter (parameter name in column 1). One also needs to specify whether a parameter is fixed (value 0) or allowed to vary (value 1) in column 2 (note that for *sfit*, 0 is free while 1 is fixed). Again, for a single-lens point-source fitting, one usually keep rho fixed to be a small value (e.g., 0.0001). During our exercises, we fix the parallax parameters (π_{EN} and π_{EE}) to be zero.

(b) Same with the *sfit* program, there are two configurations files in the `./input` directory that will affect the plot generation. One can specify the event name in the `./input/event.ctrl` (please ignore other options in the file, which are related to microlens parallax fitting, which is disabled in the present version). In the `./input/model.dat.ctrl`, one may specify the time range and step size (in the units of day) for which the microlensing model plot is generated. One can specify several segments with various step sizes.

(c) [Optional] `./input/mcmc.ctrl` file configures parameters for the MCMC procedure. One can use the default values for most of the exercises. These parameters are explained in the header of the file. `./input/corr.matrix` and `./input/corr.sigma` can be used to specify the correlation matrix used to generate initial transitional probability for the MCMC. To apply the initial correlation matrix, one need to assign `READ_CORR` parameter to be 1 in the `./input/mcmc.ctrl` file.

(3) To compile the code, issue the following command in the `single_lens_mcmc` directory:

make

To execute the program, simply do:

```
./run
```

(4) Output and Plots

During the run, the current chain parameters and X^2 values are updated in realtime in */output/chain.best*. The chain points in the burning phase are stored in */output/chain.burn* file, and the points after the burning phase is finished are in */output/chain.stable* file. One can inspect the status of the run by checking these files. After the run is finished, the best-fit solution is stored in */output/chain.best* file. One can refer to */plot/column.info* for the name of each column of these chain files. For every ~30 sec, the status of the MCMC procedure is summarized in */output/status* file.

One can also inspect the status of the run during the plot tools in the *plot* directory. The current solution is printed in the form of light curves (*plot/plot.** are the light curves and residuals for each observatory) and model curves (*plot/model.dat*) every few minutes to *plot* directory. One can look at the light curve and model plotted with caustics for the current solution by:

```
cd plot  
make lc
```

To see the light curve plots:

```
gv plotlc.eps
```

Use *plot/sm.ctrl* file to adjust the box sizes for the light curve panel, residual panel and the source trajectory plot.

Additionally, one can also plot the MCMC links stored in *chain.stable* or *chain.burn* to inspect the correlations and uncertainties of the parameters. For example, if one is already in the *plot* directory, do the following to generate such a plot from the *output/chain.stable* file:

```
cp ../output/chain.stable ./  
sm -m plotmcmc.sm chain.stable  
gv plotmcmc.eps
```

III) binary_lens_mcmc

The binary-lens fitting code is located at *~/mcmc/binary_lens_mcmc*. The setup and execution of the *binary_lens_mcmc* program is essentially the same as the *single_lens_mcmc*, except that one needs to configure *input/fs.ctrl* file. This file configures the light curve sections in which finite-source calculations should be performed, and point-source calculations will be performed for the rest of the light curve. This setup is to save runtime as finite-source calculations are

orders of magnitude more expensive than the point-source for binary lens. Every line in the file defines a region with the start and end time in HJD' (= HJD - 2450000). And in the /plot directory for the binary_lens_mcmc program, one can also generate the source trajectory along with the caustics of the best-fit solution by executing:

```
make tj
```

and the trajectory is generated as *plottj.eps*.