MESA Batch Runs

PDF of README

This directory contains tools to automate running multiple MESA models with different parameters. This README explains how to use these tools, their options, and the recommended workflow.

Directory Structure

```
batch_runs/
├── bin/                    # Executable scripts for batch operations
│   ├── make_batch.py       # Script to generate inlists from CSV
│   ├── make_batch.sh       # Shell script version of make_batch.py
│   ├── run_batch.py        # Script to run all inlists sequentially
│   ├── run_batch.sh        # Shell script version of run_batch.py
│   ├── dependency_check.py  # Checks required dependencies
│   ├── verify_inlists.py    # Script to verify inlist parameters
│   ├── verify_outlists.py   # Script to verify run outputs
│   └── construct_output.py  # Script to extract results into CSV
│
├── batch_inlists/          # Directory for generated inlist files
│   └── *.inp               # Generated inlist files
│
├── runs/                   # Directory for run outputs
│   └── */                  # Subdirectories for each model run
│
├── analysis/               # Analysis and visualization tools
│   ├── plot_hr.py          # Script to generate HR diagram plots
│   ├── plot_ccore_mass.py  # Script to plot core mass evolution
│   ├── plot_composition.py # Script to plot composition profiles
│   └── plot_timing.py      # Analyze runtime performance
│
├── notebooks/              # Interactive Jupyter notebooks
│   ├── run_batch.ipynb     # Notebook version for generating run script
│   └── make_batch.ipynb    # Notebook version for generating inlists
│
├── plots/                  # Output directory for generated plots
│   └── *.png               # Plot image files
│
├── MESA_Lab.csv            # Parameter combinations for batch runs
├── filled_MESA_Lab.csv     # Results from completed runs
└── run_timings.csv         # Performance data for each run
```

Workflow Overview

The typical workflow for batch runs is:

1. **Prepare a CSV** file with parameter combinations to explore
2. **Generate inlists** using `bin/make_batch.py` or `bin/make_batch.sh`
3. **Run the models** using `bin/run_batch.py` or `bin/run_batch.sh`
4. **Analyze the results** using the scripts in the `analysis/` directory or `bin/construct_output.py`

Detailed Steps

1. Use Provided CSV File

The online spreadsheet and the provided `MESA_Lab.csv` file contain the same parameter combinations.

You don't need to create your own CSV file. Use the provided `MESA_Lab.csv` file or access the online spreadsheet

This CSV file already contains the necessary columns:

- `YOUR NAME` (your name)
- `initial mass [Msol]` (stellar mass in solar masses)
- `initial metallicity` (Z value)
- `overshoot scheme` ("no overshooting", "exponential", or "step")
- `overshoot parameter (f_ov)` (overshooting parameter)
- `overshoot f0` (f0 parameter for overshooting)

2. Generate Inlists

Using Python Script:

```
python make_batch.py MESA_Lab.csv
```

Using Shell Script:

```
./make_batch.sh MESA_Lab.csv
```

This will:

1. Create the `batch_inlists` directory if it doesn't exist
2. Generate an inlist file for each parameter set in the CSV
3. Name each inlist file according to its parameters (e.g., `inlist_M2_Z0.014_exponential_fov0.01_f00.001.inp`)

Options during inlist generation:

You will be prompted to choose whether pgstar (visualization) should be enabled:

- Answer `yes` to enable visualization during runs (slower but you can see progress)
- Answer `no` to disable visualization (faster for batch processing)

Alternative: Jupyter Notebook

AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

3. Run the Models

Using Python Script:

```
python run_batch.py
```

Using Shell Script:

```
./run_batch.sh
```

This will:

1. Process each inlist in `batch_inlists` directory
2. Create a subdirectory in `runs` for each model
3. Copy the model results to its respective subdirectory
4. Record timing information in `run_timings.csv`

Alternative: Jupyter Notebook

AAAAAAAAAAAAAAAAAAAAAA

4. Analyze Results

Extract Data to CSV:

```
python construct_output.py
```

This will create a CSV file (`filled_MESA_Lab.csv`) with the results from all runs, including:

- Input parameters
- log(Teff)
- log(L)
- Core mass
- Core radius
- Age at TAMS
- Runtime

Create Plots:

```
python plot_hr.py             # Create HR diagrams
python plot_ccore_mass.py     # Plot core mass evolution
python plot_composition.py    # Plot composition profiles
```

These scripts will:

1. Read data from all models in the `runs` directory
2. Create comparison plots for all models
3. Save plots to a `plots` directory

Verification Tools

To verify that your inlists were generated correctly:

```
python verify_inlists.py MESA_Lab.csv
```

To verify that your runs completed successfully and match the expected configurations:

```
python verify_outlists.py MESA_Lab.csv
```

Compatibility Notes

- The Python scripts require Python 3.6+ and the `mesa_reader` package for analysis scripts
- The shell scripts require a UNIX-like environment (Linux, macOS, or WSL on Windows)
- The Jupyter notebooks can be run in Google Colab for platform independence

Troubleshooting

- If a run fails, check the `run.log` file in the corresponding run directory
- Verify that the MESA installation is working with a single model before attempting batch runs
- Make sure paths are set correctly for `$MESA_DIR` and `$MESASDK_ROOT`
- Ensure all inlists have valid parameters (use `verify_inlists.py` to check)

Running Individual Models

To run a specific model rather than the entire batch:

1. Copy the desired inlist file from `batch_inlists` to the main MESA directory as `inlist_project`
2. Run MESA as normal with `./rn`

Example Usage

```
# Generate inlists from the provided CSV
python make_batch.py MESA_Lab.csv

# Run a subset of models for testing
cp batch_inlists/inlist_M2_Z0.014_noovs.inp ../inlist_project
cd ..
./rn

# Run all models in batch
cd batch_runs
./run_batch.sh

# Extract results to CSV
python construct_output.py

# Generate plots
python plot_hr.py
python plot_ccore_mass.py
```