MAGRATHEA
Tutorial and Practice Problems

A **1D planet structure code** written in **C++** which considers the case of fully differentiated interiors. The code integrates the hydrostatic equation using **a shooting method**. The code returns the pressure, temperature, density, phase, and radius at **steps of enclosed mass**. The code supports **4 layers: core, mantle, hydrosphere, and atmosphere**. Each layer has a phase diagram with equations of state (EOS) chosen for each phase.

The code was developed by Chenliang Huang, David R. Rice, and Jason H. Steffen at the University of Nevada, Las Vegas starting in 2017. Paper for complete info: https://ui.adsabs.harvard.edu/abs/2022MNRAS.513.5256H.

Installation Instructions (all systems):

1. Open a Terminal/Shell.
2. Check C++ compiler is installed: `g++ --version`
3. Install the GNU Scientific Library*
    a. Check if installed already: `which gsl`
    b. Linux: `sudo apt-get install libgsl-dev`
    c. Mac/Windows/Other: Download "current stable version" tar file from https://www.gnu.org/software/gsl/
        1.) Navigate to tar file location in terminal: `cd ~/Downloads`
        2.) Unzip tar: `tar -xvxf gsl-latest.tar.gz`
        3.) Navigate into gsl: `cd gsl-2.7`
        4.) `sudo ./configure && make && make install`
4. Navigate to where you want to work. `mkdir Tutorial`
5. Clone the repository: `git clone https://github.com/Huang-CL/Magrathea.git`
6. Navigate to code: `cd Magrathea`
7. Compile the code for first time: `make -B`
8. If "error while loading shared libraries" or "gsl/gsl_odeiv2.h: No such"
    a. Record path for GSL Cflags : `gsl-config --cflags`
    And GSL LDflags: `gsl-config --libs`
    b. Open *Makefile* in text editor. Remove "#"s before -I and -L (Line 4 &5) and update the path (i.e. ~/gsl/include or ~/Downloads/gsl/include; repeat same location for ~/gsl/lib).

If something goes wrong:

- Email drice986@gmail.com for GSL tar file and Magrathea zip file.
- *Backup instructions for GSL Installation: https://gist.github.com/TysonRayJones/af7bedcdb8dc59868c7966232b4da903

To run your first planet:

1. Open *run/mode0.cfg* in a text editor (e.g. notepad, VSCode, Emacs, Vim).
2. Line 12-14: set mass of the core, mantle, hydrosphere, and atmosphere in Earth-masses.
3. Line 16-20: set surface temperature and any temperature discontinuities.
4. Line 21: set output file path and name.
5. **In terminal**, in your `Magrathea` directory, compile any changed files with: `make` (use `make -B` for the first compilation)
6. Run MAGRATHEA with: `./planet run/mode0.cfg`
7. Check planet's radius: `tail -2 result/Structurefilename.txt`

   Outputs a file with step number, radius (Earth radius), pressure (GPa), enclosed mass (Earth mass), density (g cm^{-3}), temperature (K), and composition

Star at https://github.com/Huang-CL/Magrathea if you find our code useful!

Further instructions for reference to complete projects.

Bulk Input Mode – `mode3.cfg`:

If run with **`./planet run/mode3.cfg`**, Magrathea will require a space separated file, where each row of the file lists the **total mass in Earth-masses and fraction of mass in each layer** for a planet (water=hydrosphere). Any remaining mass is put in the atmosphere layer.

Example:

```
Mass  fCore  fMantle  fWater
2     0.2    0.4      0.4
1.5   0.5    0.39     0.1
```

The config file requires the input file name, output file name, and Tgap similar to mode0. Example input files in the */input* directory. MAGRATHEA will generate an output file with mass of core, mantle, hydrosphere, and atmosphere and the radius of the core, mantle, hydrosphere, and planet for each line in the input file.

Changing Phase Diagrams for Each Layer:

The user-defined mass of each layer determines when the integrator changes layers. But within a layer, the pressure and temperature conditions determine the material and phase of material used.

We provide several preset phase diagrams for each layer. Brief descriptions of each can be found in *src/phase.h.*

The user can modify an existing phase diagram in *src/phase.cpp* which contains the definitions for the functions which determine which equations of state are used in each layer at P,T conditions. Changing a material can be accoplished by changing the corresponding return values of `find_phase_water_default`, `find_phase_Fe_default`, or

`find_phase_Si_default` in *phase.cpp* using conditionals to set the desired pressure and temperature where the material/phase will exist. Equations of State names from our library of materials are defined in *src/EOSlist.cpp*.

Example:

```
// Fe Default: hcp and Liquid iron
EOS* find_phase_Fe_default(double P, double T)
{
 if (P <= 0 || T <= 0)
 {
  return NULL;
 }
 P /= 1E10;              // convert microbar to GPa
 // Default Core
 if( T > 12.8*P + 2424 && T > 13.7*P + 2328)
        return Fe_liquid; // Liquid if above a certain Temp
 else
        return Fe_hcp;     // use hcp Iron for all regions.
}
```

Changing/Adding Equations of State:

The file *src/EOSlist.cpp* contains many equations of state for planet materials and different measurements for the same material. They can be returned in *phase.cpp* to build your planet model (see previous). Individual EOS can also be changed and added to.

Example EOS:

```
double Fe_hcp_array[][2] = {{0,2}, {1,mFe/8.43}, {2,177.7},
{3,5.64}, {5,mFe}, {7,322}, {8,2.09}, {9,1.01}, {10,0.05},
{14,1}, {15,26}};

EOS *Fe_hcp = new EOS("Fe hcp (Smith)", Fe_hcp_array,
sizeof(Fe_hcp_array)/2/sizeof(Fe_hcp_array[0][0]));
```

Adding a new EOS:

1.) Define a new EOS in *src/EOSlist.cpp*

1a.) Please document new EOS in a way to easily find the source.

2.) List the new EOS in *src/EOSlist.h*

3.) Delete the EOS at the end of *src/main.cpp*.

The dictionary contains keys to indexes of variables for the EOS and their corresponding value. Example: {0,2} means Vinet formulation, {2,177.7} is the bulk modulus of 177.7 GPa.

Index	Variable	Unit	Comment
0	EOS formula type		(0) 3rd order Birch-Murnaghan, (1) 4th order Birch-Murnaghan, (2) Vinet, (3) Holzapfel, (4) Keane, (6) Ideal Gas, (7) Density-Pressure input table from file for interpolation, (8-12) same as 0-4 in combination with RTPress.
1	V0	cm^3 mol^-1	Molar volume at reference point
2	K0	GPa	Bulk modulus
3	K0'		Pressure derivative of the bulk modulus. Default 4
4	K0"	GPa^-1	Second pressure derivative
5	m_mol	g mol^-1	Molar mass
6	P0	GPa	The minimum pressure, corresponding to V0. Default 0
7	Theta0	K	Fitting parameter of Einstein or Debye temperature. Default 1
8	gamma_0		Fitting parameter of Gruneisen parameter
9	beta		Fitting parameter of Gruneisen parameter
10	gamma_inf		Fitting parameter of Gruneisen parameter. Default 2/3
11	gamma_0'		Volume derivative of the Gruneisen parameter
12	e0	10^-6$ K^-1	Electronic contribution to Helmholtz free energy. Default 0
13	g		Electronic analogue of the Gruneisen parameter
14	n		Number of atoms in the chemical formula. Default 1
15	Z		Atomic number (number of electron)
16	T0	K	Reference temperature for the thermal pressure. Default 300
17	alpha0	10^-6 K^-1	The zeroth order coefficient of thermal expansion at a reference pressure P0
18	alpha1	10^-6 K^-2	The first order coefficient of thermal expansion at a reference pressure P0
19	xi		Power law index in the coefficient of thermal expansion. Default 0
20	c_p0	10^7 erg g^-1 K^-1	Specific heat capacity at constant pressure
21	c_p1	10^7 erg g^-1 K^-2	Coefficient for specific heat capacity
22	c_p2	10^7 erg g^-1 K	Coefficient for specific heat capacity
23	Debye approx		Positive number for Debye, otherwise Einstein
24	thermal type		See Paper

Practice Problems

Choose one project below for your team to work on.

Project 1: Find Possible Compositions of LHS 1140b and HD 137496b

Goal: Find the possible interior structures of two Super Earth's.

LHS 1140 b parameters from Lillo-Box et al. 2020 https://arxiv.org/abs/2010.06928

HD 137496 b parameters from Silva et al. 2022 https://arxiv.org/abs/2111.08764

LHS 1140 b		
Orbital period, P_b [days]	$\mathcal{G}$(24.736959,0.0004)	$24.73694^{+0.00041}_{-0.00040}$
Time of mid-transit, $T_{0,b} - 2400000$ [days]	$\mathcal{U}$(58399.0,58401.0)	$58399.9303^{+0.0012}_{-0.0013}$
Planet mass, M_b [$M_\oplus$]	$\mathcal{U}$(0.0,50.0)	$6.38^{+0.46}_{-0.44}$
Planet radius, R_b [$R_\oplus$]	$\mathcal{T}$(1.727,0.1,0,10)	$1.635^{+0.046}_{-0.046}$
Orbital inclination, i_b [deg.]	$\mathcal{T}$(89.89,0.05,70,90)	$89.877^{+0.049}_{-0.045}$
Planet density, ρ_b [$g \cdot cm^{-3}$]	(derived)	$8.04^{+0.84}_{-0.80}$
Transit depth, Δ_b [ppt]	(derived)	$4.93^{+0.27}_{-0.27}$
Orbit semi-major axis, a_b [AU]	(derived)	$0.0957^{+0.0019}_{-0.0019}$
Relative orbital separation, $a_b/R_\star$	(derived)	$96.4^{+2.2}_{-2.1}$
Transit duration, $T_{14,b}$ [hours]	(derived)	$2.055^{+0.048}_{-0.049}$
Planet surface gravity, g_b [$m \cdot s^{-2}$]	(derived)	$23.4^{+1.9}_{-2.0}$
Incident Flux, $F_{\mathrm{inc},b}$ [$F_{\mathrm{inc},\oplus}$]	(derived)	$4.98^{+0.23}_{-0.22}$
Stellar effective incident flux, S_b [$S_\oplus$]	(derived)	$0.477^{+0.022}_{-0.021}$
Stellar luminosity, $L_\star$ [$L_\odot$]	(derived)	$0.477^{+0.022}_{-0.021}$
Equilibrium temperature, $T_{\mathrm{eq},b}$ [K]	(derived)	$378.9^{+4.3}_{-4.2}$

	HD 137496 b
Period P_b [days]	$1.62116^{+7.91\,10^{-5}}_{-8.06\,10^{-5}}$
RV semi-amplitude K_b [m s^{-1}]	2.14 ± 0.29
Radius ratio $R_{p,b}/R_*$	$0.0076^{+0.00030}_{-0.00029}$
Impact parameter b_b	$0.27^{+0.14}_{-0.17}$
Eccentricity e_b	0
Time of inferior conjunction $t_{ic,b}$ [BJD-2450000]	$8039.1317^{+0.0028}_{-0.0029}$
Transit depth [ppm]	$57.5^{+4.61}_{-4.28}$
Transit duration T_b [hr]	$3.28^{+0.10}_{-0.14}$
Ingress to egress duration τ_b [hr]	$3.23^{+0.11}_{-0.15}$
Scaled semi-major axis a_b/R_*	3.70 ± 0.07
Semi-major axis a_b [AU]	0.02732 ± 0.00019
Inclination i_b [°]	$85.8^{+2.6}_{-2.2}$
Radius R_b [$R_\oplus$]	$1.31^{+0.06}_{-0.05}$
Mass M_b [$M_\oplus$]	4.04 ± 0.55
Density ρ_b [g cm^{-3}]	$10.49^{+2.08}_{-1.82}$
Equilibrium temperature $T_{eq,b}$ [K]	2130^{+30}_{-29}

1. For each planet, use *mode0.cfg* to keep the total mass the same but vary core and mantle mass percentages.
    a. Start with 6.38 Earth-mass for LHS 1140b and 32% of mass in core:
    `mass_of_core=2.0416 mass_of_mantle=4.3384`
    b. Save output file with a name to remember percentages. (Output file will be appended to every time if output file name is not changed.)
2. Check output for planet radius in the last line of file.
3. Change core and mantle percentages until radius matches observed values.
4. Report planet's Core Mass Fraction, the core mass divided by total mass.

Suggested Roles: Divide into teams for each planet, 1+ group member runs the code, 1 group member records radii found for each core mass.

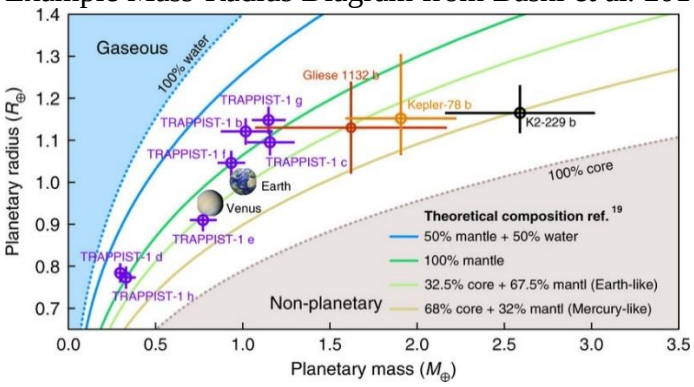
Extension: Add hydrosphere/atmosphere and find more possible interior structures

Side note: This project gives you a chance to feel the effect of internal structure on planet observables. Project 4 will find possible internal structures systematically.

Project 2: Mass-Radius Diagram

Goal: Create mass-radius lines of constant interior structure for a mass-radius diagram.

Example Mass-Radius Diagram from Bashi et al. 2017



Use *mode3.cfg* to run planets from 0.1 to 4 Earth-masses with the 6 compositions in the Figure above: 100% water, 100% core, 100% mantle, 50%/50% core/mantle, 32.5%/67.5% core/mantle, 68%/32% core/mantle.

1.) 3 input files for 100% planets are already in the */input* directory. Create remaining 3 files with text editor or run with python the script, *input/massradiusinput.py*.

2.) Set up mode3.cfg.

b) Save output with differing filenames

b) Use full solver: solver=`1`

c) Set temperature profile. Surface_temp=300 all temp jumps equal to 0

3.) Run: `./planet run/mode3.cfg`

4.) Check output file for planet radius in final column.

5.) Plot the radius with mass of each resulting file. Suggestion to use Excel or Python.

6.) Add Trappist-1 system's mass and radius from Agol et al. 2020 https://arxiv.org/abs/2010.01074:

Planet:	b	c	d	e	f	g	h
R [$R_\oplus$]	$1.116^{+0.014}_{-0.012}$	$1.097^{+0.014}_{-0.012}$	$0.788^{+0.011}_{-0.010}$	$0.920^{+0.013}_{-0.012}$	$1.045^{+0.013}_{-0.012}$	$1.129^{+0.015}_{-0.013}$	$0.755^{+0.014}_{-0.014}$
M [$M_\oplus$]	1.374 ± 0.069	1.308 ± 0.056	0.388 ± 0.012	0.692 ± 0.022	1.039 ± 0.031	1.321 ± 0.038	0.326 ± 0.020
ρ [$\rho_\oplus$]	$0.987^{+0.048}_{-0.050}$	$0.991^{+0.040}_{-0.043}$	$0.792^{+0.028}_{-0.030}$	$0.889^{+0.030}_{-0.033}$	$0.911^{+0.025}_{-0.029}$	$0.917^{+0.025}_{-0.029}$	$0.755^{+0.059}_{-0.055}$

Python: plt.errorbar(x,y, yerr=num, xerr=num)

Suggested roles: 1 makes 100% planets, 1 each makes the other 3 compositions, 1 starts a python program to plot data.

Extension: Add curves for planets with thin atmospheres (<0.1% atmosphere). The default atmosphere is 3 g mol^{-1} for H/He.

Project 3: Vary a planet's composition

Goal: Find how the radius and interior of a "water-world" changes with various materials. With each change below you can choose to reset the previous changes or keep them and add the new change.

1. Using *mode0.cfg* to run a 1 Earth-mass water world with 16.5% of mass in core, 33.5% in mantle, and 50% in the hydrosphere with a 300 K surface.
    a. Check radius of planet in last line of output file.
2. Change the temperature up to 350 K and the hydrosphere phase diagram to "water_tabulated" in *mode0.cfg* and save as a separate output file. Run the planet again.
    a. Our default water phase diagram only has condensed water phases, "water_tabulated" uses the AQUA table from Haldemann et al. 2020.
3. Change the EOS used for the core to an Iron-Silicate Alloy.
    a. Change the temperature back to 300 K.
    b. In *EOSlist.cpp* find the names of the two Iron-Silicate alloys: `EOS *[name]`
    c. In *phase.cpp* find the function that is the default core phase diagram: `EOS* find_phase_Fe_default (double P, double T)`
    d. Change the default `Fe_hcp` (hexagonal closed packed iron) to the iron silicate alloys name of your choice (7% silicate or 15% or repeat this and do both)
    e. Recompile the code with: `make`
    f. Run the planet with a different output file name.
4. Change the Bulk Modulus of the Post-Perovskite.
    a. The default mantle uses `Si_PPv_Sakai` which is the EOS for high-pressure MgSiO3 in the lower mantle from Sakai et al. 2016.
    b. They report a bulk modulus of 203+/-2 GPa for a Keane EOS fit. However, previous results found a bulk modulus as high as 250 GPa. A higher bulk modulus means the material is stiffer: the density doesn't increase as quickly with pressure.
    c. In *EOSlist.cpp* change the bulk modulus of `Si_PPv_Sakai` from 203 to 250 GPa. The storage structure for EOS is described on Page 4.
    d. Recompile with `make` and run the planet with a different output file name.
    e. Make sure to change it back to 203 GPa for future use.
5. Record the four planets' (default, AQUA, Fe-Si core, and PPv) radii. Also, find the radius, temperature, and pressure of the core-mantle boundary (CMB) in each output file.
    a. You will see in the output file the point where it switches for silicates to iron which is the CMB.

Extension: Use plot/quickdensityplot.py to plot the interior conditions of all four planets. See how the interiors change. What unphysical thing happens in the post-perovskite by changing its bulk modulus by so much? The python can be modified to plot all four on the same plot also.

Project 4: Composition Finder for LHS 1140b

Goal: Systematically find compositions that match LHS 1140 b parameters from Lillo-Box et al. 2020. Mass: 6.38 Earth-Masses. Radius: 1.635 Earth-Radii.

1. Make a text file named "LHS1140binput.txt" which has two rows with values separated by spaces:
   M (Earth-masses) R (Earth-radii)
   6.38 1.635
2. We will be working with *mode4.cfg* which uses a secant method to shoot for the 3rd layer while holding the other 2 layers in constant ratio.
   a. Set `input_file` to *LHS1140binput.txt* and its location.
3. First find the CMF if all rock and iron as done in Project 1.
   a. Set `find_layer=2` to find the mantle mass
   b. Set outer layer to water and inner layer to core.
   c. Set `PMR_min` to 0.0 so that there's no water and all core and set the max equal to min.
   d. Run the code with `./planet run/mode4.cfg`
   e. Look at output file to find the CMF (core/total mass) compared to Project 1.
4. LHS 1140b could have a hydrosphere. Find the maximum hydrosphere it could have if it is only iron and water with no rock.
   a. Set `find_layer` to the hydrosphere, and hold the mantle and core in constant PMR of 0%.
5. Finally, find the water mass fraction across a range of mantle:core PMRs.
   a. Set `find_layer` to the hydrosphere.
   b. `PMR_max` to approximately one minus the CMF you found in #3 above.
   c. `PMR_min` to around a Mercury-Like PMR of 20%
   d. Set the `PMR_step` to 1%
   e. This run will take ~5 seconds*number of steps (~2-5 minutes)
6. Look at the output file and report/graph how the possible WMF varies with CMF.

Extension: Draw multiple mass and radius measurements from the reported errors (see Project 1) and add them to the input file. Rerun step #5 above to find possible compositions across all of draws of mass and radius.