

MAGRATHEA: an open-source spherical symmetric planet interior structure code

Chenliang Huang (黄辰亮),^{1*} David R. Rice^{2*} and Jason H. Steffen²

¹*Lunar and Planetary Laboratory, University of Arizona, Tucson, AZ 85721*

²*Department of Physics & Astronomy, University of Nevada Las Vegas, PO Box 454002, Las Vegas, NV 89154*

22 April 2022

ABSTRACT

MAGRATHEA is an open-source planet structure code that considers the case of fully differentiated spherically symmetric interiors. Given the mass of each layer and the surface temperature, the code iterates the boundary conditions of the hydrostatic equations using the method of shooting to a fitting point in order to find the planet radius. The first version of MAGRATHEA supports a maximum of four layers of iron, silicates, water, and ideal gas. With a few exceptions, the temperature profile can be chosen between isothermal, isentropic, and user-defined functions. The user has many options for the phase diagram and equation of state in each layer and we document how to add additional equations of state. We present MAGRATHEA’s capabilities and discuss its applications. We encourage the community to participate in the development of MAGRATHEA at <https://github.com/Huang-CL/Magrathea>.

Key words: planets and satellites: composition – planets and satellites: general – planets and satellites: interiors – equation of state

1 INTRODUCTION

Discoveries such as Kepler-10b, LHS 1140b, the seven planets of the TRAPPIST-1 system, and Proxima Centauri b (Batalha et al. 2011; Dittmann et al. 2017; Ment et al. 2019; Gillon et al. 2017; Grimm et al. 2018; Anglada-Escudé et al. 2016) show that the discovery of small rocky planets, or planets with minimal atmospheres, are now firmly within observational capabilities. Data from the Kepler space telescope shows that planets between 1-4 R_{\oplus} are among the most common types in our galaxy (Batalha et al. 2013; Fressin et al. 2013; Petigura et al. 2013). The densities of these observed terrestrial planets allow for a range of interior compositions from Mercury-like (Santerne et al. 2018) to volatile rich (Léger et al. 2004). Fitting mass-radius relationships to observed planets provides a probe into planet composition (e.g. Zeng & Sasselov (2013); Unterborn et al. (2018)). However, the amount of mass in each differentiated layer cannot be determined by density alone (Rogers & Seager 2010; Dorn et al. 2015).

The community uses a variety of models to characterize the interior structure of planets. Underlying these models are differing computational techniques (e.g. shooting in Nixon & Madhusudhan (2021), relaxation in Unterborn et al. (2018), or calculus of variation in Zeng et al. (2021)) and numerous experimental measurements and theoretical estimates of the equation of state (EOS) for planet-building materials. For example, a number of models use bridgmanite primarily in

the mantle and do not include high-pressure post-perovskite (Grasset et al. 2009; Zeng et al. 2016; Brugger et al. 2017) while others capture detailed upper-mantle chemistry (Valencia et al. 2007a; Dorn et al. 2017; Unterborn & Panero 2019) and recent work adds liquid silicates (Noack & Lasbleis 2020). Another point of difference is the model’s temperature profile with some using an isothermal mantle (Seager et al. 2007) rather than isentropic (Hakim et al. 2018). While others limit isothermal modeling to the water layer (Sotin et al. 2007; Brugger et al. 2016) and the atmosphere (Madhusudhan et al. 2021; Baumeister et al. 2020). Newer models try to capture the complex phase diagram of water (Mousis et al. 2020; Journaux et al. 2020; Haldemann et al. 2020), the hydration of mantle materials (Shah et al. 2021; Dorn & Lichtenberg 2021), and couple the atmosphere to the interior (Madhusudhan et al. 2020; Acuña et al. 2021).

We develop MAGRATHEA¹ an open-source planet interior solver that can be customized to different user-defined planet models. Compared to other codes, the package is designed to enhance ease-of-use and flexibility. MAGRATHEA features phase diagram options and transparent EOS formatting, which enables the user to choose between a large library of EOSs and add/change materials and equations.

We are motivated by our collaboration with high-pressure physicists to understand how uncertainties in experimental

¹ Magrathea is the legendary planet where hyperspatial engineers manufacture custom-made planets in Douglas Adams’s The Hitchhiker’s Guide to the Galaxy (Adams 2001).

* E-mail: huangcl@arizona.edu, david.rice@unlv.edu

EOSs affect predictions of planet interiors. In [Huang et al. \(2021\)](#), we use our adaptable planet interior model to implement new measurements for high-pressure water-ice from [Grande et al. \(2022\)](#). Their experiments confirmed the pressure of the transition from ice-VII to ice-X and identified a transitional tetragonal ice-VII_t phase. This improved H₂O equation of state changed the predicted radius of a pure water, 10 M_⊕ planet by over four per cent from [Zeng et al. \(2016\)](#).

In this paper, we document the features and demonstrate the functionality of MAGRATHEA. The most up-to-date version is hosted on the GitHub platform. Our paper is laid out as follows. We describe the fundamentals of a planet interior solver in Section 2. In Section 3, we describe the specifics of our code and how to build a planet model within the code. A model is designated by defining a phase diagram for each layer and choosing an equations of state from our library for each phase. We discuss limitations to our default model in 4. We describe the code’s functionality and discuss various tests in Section 5, and end with some summarizing remarks in Section 6.

2 INTERIOR STRUCTURE SOLVER

We consider a simplified planet structure with four layers: an iron core, a silicate-dominated mantle, a hydrosphere of water/ice, and an ideal gas atmosphere. We assume these layers are spherically symmetric and that a single solution to composition at a given pressure and temperature exists within each layer. Our approach is similar to other interior structure solvers used for exoplanets (e.g. [Léger et al. 2004](#); [Valencia et al. 2006](#); [Sotin et al. 2007](#); [Seager et al. 2007](#)). A schematic depiction of MAGRATHEA from input to output is shown in Fig. 1. Given the mass of each of these layers, $M_{\text{comp}} = \{M_{\text{core}}, M_{\text{mantle}}, M_{\text{hydro}}, M_{\text{atm}}\}$, the code calculates the radius returning the pressure $P(m)$, density $\rho(m)$, and temperature $T(m)$ with enclosed mass m by solving the following four equations:

(i) Mass continuity equation

$$\frac{dr(m)}{dm} = \frac{1}{4\pi r^2 \rho(m)}, \quad (1)$$

(ii) Hydrostatic equilibrium

$$\frac{dP(m)}{dm} = -\frac{Gm}{4\pi r^4}, \quad (2)$$

(iii) Temperature gradient

If the isothermal option is chosen, the temperature gradient is 0. When the isentropic option is chosen, depending on the available thermal properties of the phases, the temperature gradient can be calculated using either of the following two formulae. If the Grüneisen parameter γ is available, we can have

$$\begin{aligned} \frac{dT(m)}{dm} &= \left. \frac{dT}{dV} \right|_S \frac{dV}{d\rho} \frac{d\rho}{dm} \\ &= -\frac{m_{\text{mol}}}{\rho^2} \left. \frac{dT}{dV} \right|_S \left(\frac{\partial \rho}{\partial P} \frac{dP}{dm} + \frac{\partial \rho}{\partial T} \frac{dT}{dm} \right). \end{aligned}$$

Thus,

$$\frac{dT(m)}{dm} = \frac{\left. \frac{dT}{dV} \right|_S \frac{Gm}{4\pi r^4}}{\frac{\rho^2}{m_{\text{mol}}} \left. \frac{\partial P}{\partial \rho} \right|_T - \left. \frac{dT}{dV} \right|_S \left. \frac{\partial P}{\partial T} \right|_\rho}, \quad (3)$$

where

$$\left. \frac{dT}{dV} \right|_S = -\frac{\gamma T}{V}, \quad (4)$$

m_{mol} and V are the molar mass and volume respectively.

Alternatively, if the thermal expansion α is available, we can have

$$\frac{dT(m)}{dm} = -\frac{\alpha T G m}{4\pi r^4 \rho c_p}, \quad (5)$$

where c_p is the specific heat capacity at constant pressure.

(iv) Equation of state (EOS)

$$P(m) = P(\rho(m), T(m)), \quad (6)$$

which is unique for each material/phase.

The boundary conditions of the model are $r = 0$ at $m = 0$ and the user-defined surface temperature and pressure, $T(M_p)$ and $P(M_p)$. The default surface pressure is 100 mbar approximately the pressure level of the broad-band optical transit radius probes ([Grimm et al. 2018](#)). Since a rapid temperature jump may occur at the boundary between chemically distinct layers, where heat can only transfer conductively ([Valencia et al. 2006](#)), the user can also set a temperature discontinuity, T_{gap} , at the boundary of adjacent layers. The Earth is often modeled with a discontinuity ranging from a 300 to 1100 K increase across the core mantle boundary ([Nomura et al. 2014](#); [Lay et al. 2008](#)) and ([Valencia et al. 2007a](#)) and ([Sotin et al. 2007](#)) use discontinuities in their exoplanet models. In our demonstrations in Section 5, we use planets that are in thermal equilibrium between their layers with $T_{\text{gap}} = \{0, 0, 0, T(M_p)\}$.

The user can choose between isothermal and isentropic temperature mode by adjusting an *isothermal* flag of the integrator. When the flag is set to true, the whole planet is assumed to be isothermal (except the atmosphere layer, see section 3.3.4). When the input EOS parameters of a phase are not sufficient to calculate the isentropic temperature gradient (see section 2.2), this specific phase would be treated as isothermal, regardless of the value of the *isothermal* flag.

2.1 Phase diagrams

We model our planets with distinct differentiated layers with defined mass fractions. However, within each layer the phase may change due to the large pressure and temperature ranges. When integrating within a layer, the code first checks the pressure and temperature to determine the appropriate region of the phase diagram. A built-in phase function outputs a link to the EOS of the material or phase in the region. The density ρ or the volume of unit cell V can then be solved from the EOS (Eq. 6).

A key feature of MAGRATHEA is the user’s ability to change the phase diagram in each layer and choose between EOSs for each phase. Fig. 2 shows our default phase diagram and Sec. 3.2 further details their implementation. The atmosphere layer is not shown in the plots and has two “phases”, isothermal for $P < 100$ bar and adiabatic for $P > 100$ bar.

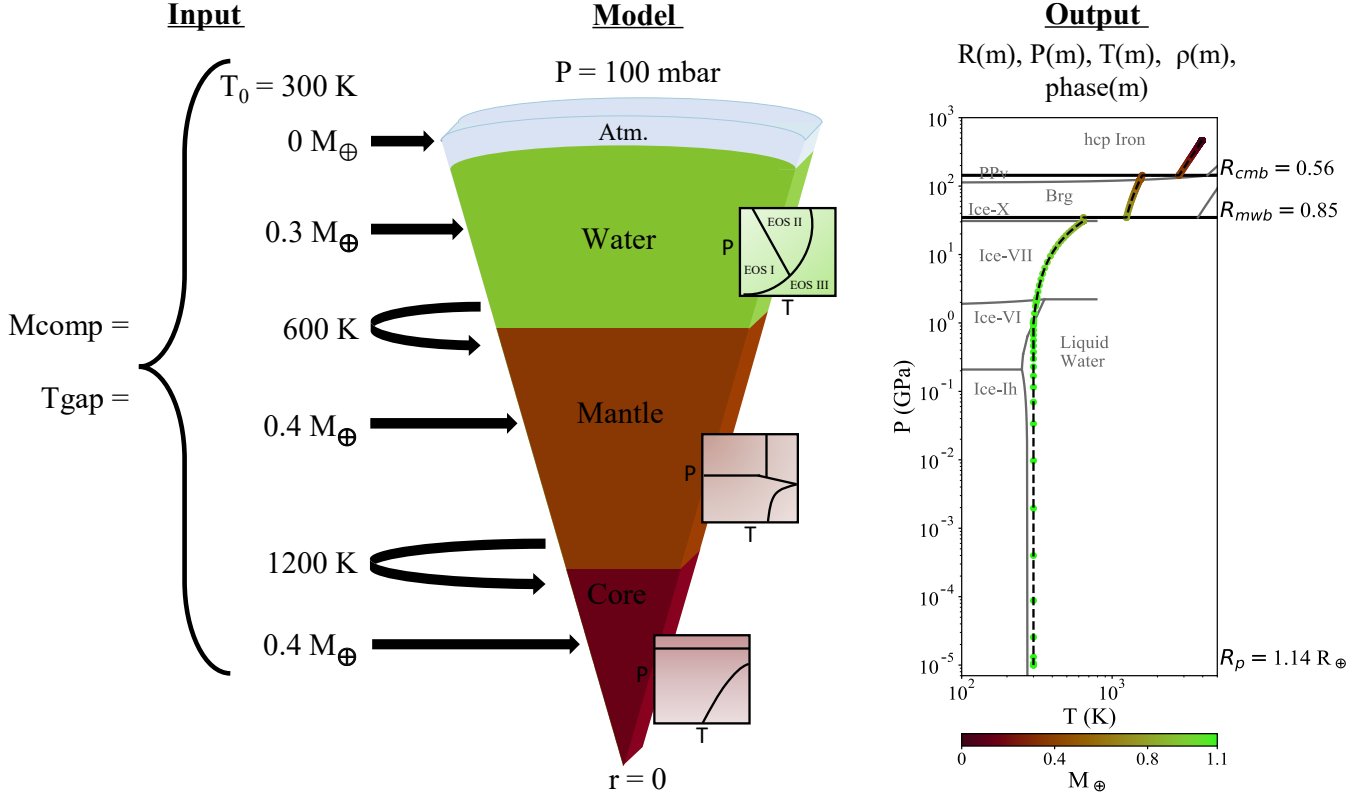


Figure 1. A schematic overview of MAGRATHEA. Showing an example input, *left*, of a $1.1 M_{\oplus}$ planet with $0.4 M_{\oplus}$ core, $0.4 M_{\oplus}$ mantle, and $0.3 M_{\oplus}$ hydrosphere. The planet is not in thermal equilibrium with a surface temperature of 300 K and jumps in temperature across boundary layers of 600 K and 1200 K. *Center*, shows MAGRATHEA's four input layers with cartoons of phase diagrams defined for each layer with an EOS chosen for each phase. Default phase diagrams shown in Fig. 2. *Right*, shows the pressure and temperature with enclosed mass. The radius at boundaries and the planet radii is also shown.

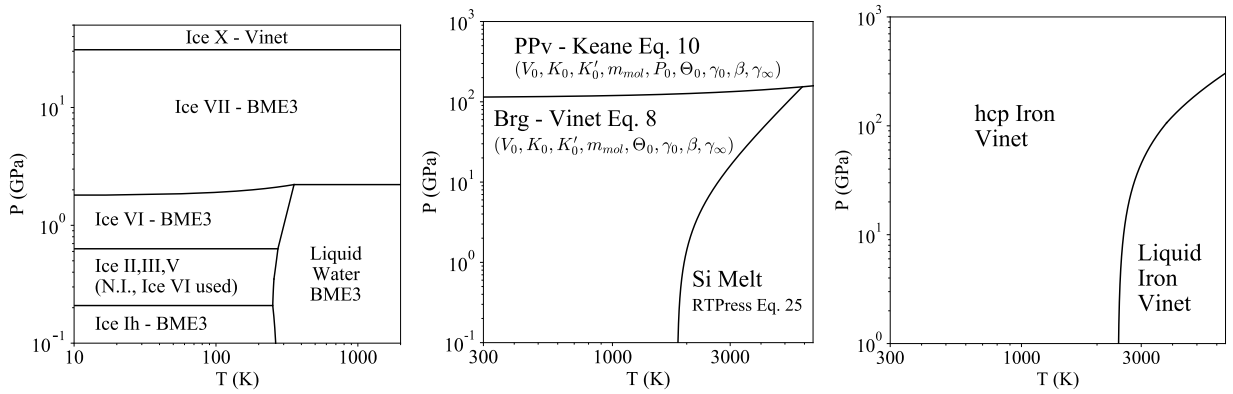


Figure 2. Default phase diagrams for hydrosphere, mantle, and core layers. EOS and phase transitions from a variety of sources detailed in Sec. 3.3.3. Type of EOS fitting equation shown from Sec. 2.2. Additionally, the parameters for mantle EOSs are shown, *middle*. The phase diagrams and choice of EOS can be customized by the user. *N.I.* is *not implemented*. The atmosphere layer also has a phase function, but our default is ideal gas at all pressure and temperatures, see Sec. 3.3.4

2.2 Equation of state formulae

The density of a phase at certain temperature and pressure is determined by the EOS, which can be fed into MAGRATHEA either through an analytical fitting formula or a tabulated pressure-density table. For the first option, each

material's EOS must provide the parameters for one of the following formulations.

According to the Mie-Grüneisen-Debye (MGD) formulation, the total pressure P can be divided into an isothermal term P_c , and a thermal term, P_{th} , expressed as (Dewaele

et al. 2006):

$$P(V, T) = P_c(V) + P_{th}(V, T) - P_{th}(V, T_0) \quad (7)$$

MAGRATHEA includes four common types of the EOS that give the reference isotherm P_c , including:

(i) The Eulerian finite strain Birch-Murnaghan EOS (BME) is the most commonly used EOS. The fourth-order BME (Seager et al. 2007) is

$$P_c = \frac{3}{2} K_0 \left(\eta^{7/3} - \eta^{5/3} \right) \quad (8)$$

$$\left(1 + \zeta_1 (1 - \eta^{2/3}) + \zeta_2 (1 - \eta^{2/3})^2 \right),$$

where

$$\eta = V_0/V, \zeta_1 = \frac{3}{4} (4 - K'_0)$$

$$\zeta_2 = \frac{3}{8} \left(K_0 K''_0 + K'_0 (K'_0 - 7) + \frac{143}{9} \right).$$

$K = -V(\partial P/\partial V)_T$ is the isothermal bulk modulus, K' is the first derivative of the bulk modulus with respect to pressure, and K'' is the second of the bulk modulus with respect to pressure. The subscript 0 refers to quantities at ambient-pressure conditions. If ζ_2 is set to zero the EOS reduces to third-order BME.

(ii) Vinet EOS (Seager et al. 2007; Smith et al. 2018; Wicks et al. 2018), which is considered to give more accurate result than BME at high pressure or large compression (Poirier 2000)

$$P_c = 3K_0 \eta^{2/3} \left(1 - \eta^{-1/3} \right) \quad (9)$$

$$\exp \left(\frac{3}{2} (K'_0 - 1) \left(1 - \eta^{-1/3} \right) \right);$$

(iii) Holzapfel EOS (Bouchet et al. 2013)

$$x = (V/V_0)^{1/3} \quad (10)$$

$$c_0 = -\ln \left(\frac{3K_0}{1003.6 \text{ GPa cm}^5 \text{ mol}^{-5/3} (Z/V_0)^{5/3}} \right)$$

$$c_2 = \frac{3}{2} (K'_0 - 3) - c_0$$

$$P_c = 3K_0 x^{-5} (1 - x) \exp(c_0(1 - x))(1 + c_2 x(1 - x));$$

(iv) Keane EOS (Sakai et al. 2016)

$$y = V_0/V \quad (11)$$

$$K'_\infty = 2 \left(\gamma_\infty + \frac{1}{6} \right)$$

$$P_c = \frac{K'_0 K_0}{K'^2_\infty} \left(y^{K'_\infty} - 1 \right) - (K'_0 - K'_\infty) \frac{K_0}{K'_\infty} \ln(y)$$

(v) An empty placeholder is available for an additional formulation of the EOS such as from Choukroun & Grasset (2007).

For phases that only exist at high pressure, the bulk modulus, K_P , can be measured more precisely at the phase transition pressure, P_0 , than at ambient pressure (Salamat et al. 2013). If K_P is applied instead of K_0 , a constant pressure P_0 under which the bulk modulus is measured is added to the

isotherm pressure. If the parameters to calculate the thermal term, P_{th} , are not provided, only P_c is calculated and a constant temperature is returned.

P_{th} is most commonly calculated from a quasi-harmonic Debye thermal pressure and the anharmonic and electronic thermal pressure, which are obtained from (Belonoshko et al. 2008; Duffy et al. 2015)

$$P_{th}(T) = \frac{\gamma}{V} E_{th}(T) + \frac{3nR}{2V} e_0 x^g g T^2, \quad (12)$$

where

$$x = V/V_0 = \eta^3, \gamma = \gamma_\infty + (\gamma_0 - \gamma_\infty) x^\beta \quad (13)$$

$$\Theta = \Theta_0 x^{-\gamma_\infty} \exp \left(\frac{\gamma_0 - \gamma_\infty}{\beta} (1 - x^\beta) \right) \quad (14)$$

$$z = \Theta/T, E_{th} = 3nRT D_3(z). \quad (15)$$

D_3 indicates a third-order Debye function. Θ_0 , γ_∞ , γ_0 , β , e_0 , and g are fitting parameters, R is the gas constant, and n is the number of atoms in the chemical formula of the compound. A numerical derivative of Eq. 7 gives the $\frac{\partial P}{\partial \rho} \Big|_T$ and $\frac{\partial P}{\partial T} \Big|_\rho$ that are required in the Eq. 3.

A different framework for the thermal term, referred to as RTpress in the following, is used in Wolf & Bower (2018) and built upon the Rosenfeld-Trazona model. In addition to the Grüneisen parameters, it has additional fitting terms with derivation in Wolf & Bower (2018).

In RTpress, the Grüneisen parameter can be written as

$$\gamma = \gamma_{0S} \frac{C_V(V, T_{0S})}{C_V(V, T)} + V \frac{b'(V)}{b(V)} \frac{\Delta S^{\text{pot}}(T_{0S} \rightarrow T)}{C_V(V, T)}. \quad (16)$$

The subscript 0S stands for the quantity evaluated along the reference adiabat,

$$T = T_{0S} = T_0 \sqrt{1 + a_1 f + \frac{1}{2} a_2 f^2}, \quad (17)$$

with,

$$f \equiv f(V) = \frac{1}{2} \left[\left(\frac{V_0}{V} \right)^{\frac{2}{3}} - 1 \right], \quad (18)$$

$$a_1 = 6\gamma_0, \quad \text{and} \quad a_2 = -12\gamma_0 + 36\gamma_0^2 - 18\gamma_0', \quad (19)$$

where γ_0 is the Grüneisen parameter at zero GPa and $\gamma'_0 = V_0(d\gamma/dV)_0$.

The Grüneisen parameter variation along the reference adiabat is

$$\gamma_{0S} = \frac{(2f + 1)(a_1 + a_2 f)}{6(1 + a_1 f + \frac{1}{2} a_2 f^2)}. \quad (20)$$

$$b(V) = \sum_n b_n \left(\frac{V}{V_0} - 1 \right)^n, \quad (21)$$

and

$$b'(V) = \sum_n b_n \left(\frac{n}{V_0} \right) \left(\frac{V}{V_0} - 1 \right)^{n-1} \quad (22)$$

are a polynomial representation of the thermal coefficients and its volume derivative in cgs units.

The total heat capacity

$$C_V(V, T) = b(V)f_T^{(1)} + \frac{3}{2}nR \quad (23)$$

is the sum of potential and kinetic contributions², where

$$f_T = \left(\frac{T}{T_0}\right)^\beta - 1, \quad f_T^{(1)} = \frac{\beta}{T_0} \left(\frac{T}{T_0}\right)^{\beta-1}. \quad (24)$$

The difference in entropy from the reference adiabat from the potential contribution is

$$\Delta S^{\text{pot}}(T_{0S} \rightarrow T) = \frac{b(V)}{\beta-1} \left(f_T^{(1)}(T) - f_T^{(1)}(T_{0S}) \right). \quad (25)$$

Finally, the thermal term of the pressure in the framework of Wolf & Bower (2018) can be written as

$$\begin{aligned} P_{th} = & -b'(V)f_T + \gamma_{0S}(V) \frac{C_{V,0S}(V) \cdot (T - T_0)}{V} \\ & + \frac{b'(V)}{\beta-1} \left[T \left(f_T^{(1)} - f_T^{(1)}(T_{0S}) \right) \right. \\ & \left. - T_0 \left(f_T^{(1)}(T_0) - f_T^{(1)}(T_{0S}) \right) \right] \end{aligned} \quad (26)$$

Besides the MGD representation, the P-V-T equation of state can also be expressed as (Fei et al. 1993)

$$V(P, T) = V(P, T_0) \exp \left[\int_{T_0}^T \alpha(P, T') dT' \right], \quad (27)$$

$\alpha(P, T)$ is the thermal expansion, which has the form of

$$\alpha(P, T) = \alpha(T) \left(1 + \frac{K'_0}{K_0} P \right)^{-\xi}. \quad (28)$$

$\alpha(T)$ is the zero-pressure thermal expansion coefficient. For $T > T_0$, it is expressed as a linear function of temperature

$$\alpha(T) = \alpha_0 + \alpha_1 T. \quad (29)$$

For phases that adopt the thermal expansion representation, the Eq. 5 can be used to calculate adiabatic temperature gradient. The heat capacity in the equation can be calculated using the fitting formula

$$c_p = c_{p0} + c_{p1}T - c_{p2}T^{-2}. \quad (30)$$

2.2.1 Tabulated Equation of State

In place of a fitting equation, the EOS can take the form of a tabulated density-pressure table which then code then interpolates. For MAGRATHEA the input file should have two columns, the first column is the density in g cm^{-3} and the second column is the pressure in GPa. The pressure must be strictly ordered. The first row of the table file, which contains header information, is skipped when the file is parsed. The program interpolates the table monotonically using the Steffen spline (Steffen 1990, no relation to our co-author) from the *gsl* package (Galassi et al. 2009).

² We believe that Eq. B.3 in Wolf & Bower (2018) should read $C_V^{\text{kin}} = \frac{3}{2}k_B$. To be consistent with the unit eV/atom chosen for b_n in their work, the number of atoms per formula unit should not be a factor.

2.2.2 User-defined function

For further flexibility, users can create or modify the EOS of a phase using their own C++ functions. MAGRATHEA supports three types of user-defined functions. First, a user can provide an EOS function that returns the material density as a function of pressure and temperature. In addition, either a temperature gradient, dT/dP , as a function of pressure and temperature or an entropy function dependent on density and temperature can be used to set up the temperature solver in a phase. If a user-defined temperature gradient function is set up, the temperature profile of this phase is not restricted to isentropic. The density function and the temperature function (dT/dP or entropy) can also be used in combination for a phase. The method to import these user-defined functions is shown in Appendix B.

3 OVERVIEW OF THE CODE STRUCTURE

MAGRATHEA is written in C++ and relies on the *GNU Scientific Library (GSL)*³ (Galassi et al. 2009). A step by step guide to run the code can be found in the *README* file on our GitHub repository.

The code is compiled with the included *Makefile*. The central interaction with the user occurs through *main.cpp*. The user may choose between seven modes by setting *input_mode*. They include the regular solver, a temperature-free solver, a two-layer solver (Sec. 3.1.1), three methods to change the EOS during run-time (Sec. 5.3), and a bulk input mode for solving many planets with the regular solver in a single run (Sec. 5.4). Each mode requires the user to define the mass of each layer in the planet. In modes where the regular solver is used, the user defines a temperature array which gives the temperature at the surface and any discontinuities between layers. This section covers the specific design of the code and options that the user may choose between—the solver in Sect. 3.1, the phase diagram in 3.2, and the EOSs in Sec. 3.3.

3.1 Solver

Solving the ordinary differential equations (ODE), Eq. 1-4, is a two-point boundary value problem (Press et al. 2007). We have a total of three differential equations for r , P , and T verses the independent variable m . P and T need to satisfy boundary condition at $m = M_p$, and the boundary condition for r is located at $m = 0$. The density may be discontinuous at a phase boundary whose location is unknown. Therefore, in our case, MAGRATHEA solves this problem with the method of shooting to a fitting point which is preferred to a relaxation method because it does not require predefined grids.

MAGRATHEA integrates ODEs using *GSL's gsl_odeiv2* (Galassi et al. 2009) with a Runge-Kutta-Fehlberg method, which is a fourth order integrator with a fifth order error estimator and an automated adaptive step-size. To have a better estimation on the boundary conditions, $P(m=0)$, $T(m=0)$, and R_p , we conduct an extra round of iteration using a “pure”

³ *GNU Scientific Library* can be found at <http://www.gnu.org/software/gsl/>

Table 1. List of status information of the planet profile object

Value	Meaning
0	Normal.
1	The result includes phase(s) that is(are) not formally implemented in MAGRATHEA.
2	The two shooting branches do not match within the required accuracy at the fitting point.
3	Under two-layer mode, the error of planet surface pressure is larger than the required accuracy.

shooting method. In this first iteration, the ODEs are integrated from an estimated planet radius R_p at M_p outside-in toward the center until $m = 0$ or $P(m) > 10^5$ GPa. The R_p , which is the only unset boundary condition at M_p , is iterated using Brent-Dekker root bracketing method. The initial estimated radius is calculated based on a crude density assumption for each layer (15, 5, 2, and 10^{-3} for iron, mantle, ice, and atmosphere respectively in the unit of g cm^{-3}).

Predetermined by the user, the values of the enclosed mass at each layer interface are set as the bounds of the ODE integrator. In contrast, the enclosed mass where the phase changes within a layer is determined by the phase diagram in P - T space. Thus, the step of the ODE integral typically does not land exactly on the location of the phase change. To avoid introducing extra inaccuracy, when the ODE integration reaches a different phase the integrator is restored to the previous step and the integration step is cut into half. The integrator can only move forward to the new phase when the step size that crosses the phase boundary is less than the ODE integrator accuracy tolerance multiplied by M_p .

Using the estimated boundary condition, we conduct the shooting to a fitting point method, which start ODE integration from both $m = 0$ and $m = M_p$ toward a fitting point with mass $m_{fit} = 0.2M_p$. The code will automatically adjusted m_{fit} if it occurs at a layer's boundary. The code uses GSL's *gsl_multiroot_fsolver_hybrids* to adjust $P(0)$, $T(0)$, and R_p , until P , T , and r at m_{fit} obtained integrating from the inner branch and from the outer branch agrees within a relative error $< 10^{-4}$. The ODE integrator tolerance will be reduced (made more strict) by a factor of three if the multidimensional root-finder can not find a solution within 15 tries. After the ODE integrator tolerance is reduced four times (three additional times, each by the stated factor of three), if the solver still cannot find a solution that satisfies the required joint accuracy at the fitting point, the code will output an error message as well as the best result it can find.

The solver returns a planet interior profile object after completion, which includes the r , P , T , m , ρ , and the component phase at each grid step. A *print* function is available to save the profile into an ASCII table. The function *getRs* returns the list of radii of the outer boundary of each component. Users can use *getstatus* to access the status information of the returned object which indicates whether precautions should be taken regarding the iteration result. A list of potential problems and their assigned return value is summarized in Table 1.

3.1.1 Simplified two layer mode

In addition to the regular solver, we include a mode for a simplified two layer solver. We use this for quick calculations of water/mantle and mantle/core planets in Huang et al. (2021). We recommend using our complete solver in most instances, but this method can be used for quicker and cheaper calculations; one such use is demonstrated in Sec. 5.3.

Two-layer mode can only calculate the structure of 300 K isothermal planets and does not support an atmosphere. Without the temperature differential equation, the program only solves the radius and pressure differential equations, with the radius boundary condition at the center, and the pressure boundary condition at the surface. This simpler problem is solved using the "pure" shooting method inside-out, starting from the equations' singular point at the planet center. The solver iterates the center pressure using the Brent-Dekker root bracket method until either the center pressure reaches the set accuracy target, or the surface pressure $P(M_p)$ is within 2% of the user specified value. If the iteration ends because the first criteria is satisfied first, an abnormal status value 3 will be returned by *getstatus*. To avoid this, users may adjust the ODE integration tolerance and center pressure accuracy target accordingly based on the specific problem.

With this solver, the two-layer input modes provide an interface to calculate mass-radius curves for hypothetical planets that are composed of only two components with fixed mass ratio (e.g. figure 3 in Huang et al. (2021)).

3.2 Phase diagram implementation

Phase diagrams are defined in *phase.cpp*. The file contains four functions corresponding to the four layers: *find_Fe_phase*, *find_Si_phase*, *find_water_phase*, and *find_gas_phase*. The *if* statement is used within each function to create the region of P - T space to which an EOS applies. The return value of each *if* statement should be the name of the pointer corresponding to the EOS for the given phase (further described in Appendix A).

Transitions can be defined as pressure or temperature conditionals. Our default phase diagrams are shown in Fig. 2. We use phase transitions which are linear with pressure/temperature within the conditionals for the transfer between solid and liquid iron (Dorogokupets et al. 2017) and between bridgmanite and post-perovskite (Ono & Oganov 2005). Parameterized phase transition curves can also be defined as separate functions in *phase.cpp* and called within the "find phase" functions. We define the function *dunaeva_phase_curve* to implement the fitting curve with five coefficients, $T(P) = a + bP + c \ln P + d/P + e\sqrt{P}$, from Dunaeva et al. (2010) for the transitions between phases of water.

3.3 Built-in Equations of State

We have over 30 EOSs available in MAGRATHEA which can be called within the phase diagrams. These include EOS functions for various planet building materials, and different parameter estimates for the same material from various works. We discuss the equations currently available for each layer in the following four subsections. New EOSs can be added to *EOSlist.cpp* by following the storage structure described in

Appendix A. In Fig. 3 we show mass-radius relationships up to ten Earth-masses for planets of one layer with a large selection of our implemented equations. The index of equations in Fig. 3 is also noted in the text in brackets.

In addition to the equations listed below, the program includes tabulated EOS of iron, silicate, and water from Seager et al. (2007) [CS, MS, WS], who combined empirical fits to experimental data at low pressure and Thomas-Fermi-Dirac theory at high pressure.

3.3.1 Core/Iron

At the extreme pressures of a planetary core, iron is stable in a hexagonal close-packed (HCP) phase. The program includes HCP iron EOSs from Bouchet et al. (2013) [C1], Dorogokupets et al. (2017) [C2], and Smith et al. (2018) [CD] (see Fig. 3). We choose as our default equation the Vinet fit (Eq. 9) from Smith et al. (2018) measured by ram compressing iron to 1.4 TPa. We determine the fitting parameters of the Grüneisen by fitting Eq. 13 to Fig. 3b in Smith et al. (2018) through maximum likelihood estimation. The default core layer includes a liquid iron EOS and melting curve from Dorogokupets et al. (2017) who compressed iron to 250 GPa and 6000 K.

Iron-silicate alloy EOSs are implemented as well. Wicks et al. (2018) experimentally determined the EOS for Fe-Si alloys with 7 wt per cent Si [CW1] and 15 wt per cent Si [CW2]. These alloy equations are useful in reproducing the density of the Earth's core which contains an unknown mixture of light elements (Hirose et al. 2013). An additional liquid iron equation is included from Anderson & Ahrens (1994).

At lower temperatures and pressures, fcc (face-centered cubic) iron is not currently implemented. The EOS of fcc iron has similar parameters to that of hcp iron ($K_{0,fcc} = 146.2$, $K_{0,hcp} = 148.0$). The fcc-hcp-liquid triple point is at 106.5 GPa and 3787 K (Dorogokupets et al. 2017).

3.3.2 Mantle/Silicate

The main mineral constituent of the Earth's mantle is bridgmanite (Brg), referred to in the literature cited here as silicate perovskite, which at high pressure transitions to a post-perovskite (PPv) phase (Tschauner et al. 2014). MAGRATHEA includes third-order BME (Eq. 8) Brg from Shim & Duffy (2000) [M2] and Vinet EOSs (Eq. 9) for Brg and PPv measured with *ab initio* simulations and confirmed with high pressure experiments from Oganov & Ono (2004) and Ono & Oganov (2005) [M1].

Our default mantle [MD] includes Brg from Oganov & Ono (2004) and an updated PPv thermal EOS from Sakai et al. (2016). Sakai et al. (2016) compressed PPv to 265 GPa with a laser-heated diamond anvil cell and extended the EOS using a Keane fit (Eq. 11) to 1200 GPa and 5000 K with *ab initio* calculations. At high temperatures (>1950 K at 1.0 GPa), we use a liquid MgSiO_3 with RTPress EOS from Wolf & Bower (2018) with melting curve from Belonoshko et al. (2005). The silicate melt transfers directly to PPv when $P > 154$ GPa and $T > 5880$ K since the behavior in this regime is unknown. An alternate silicate melt is implemented from Mosenfelder et al. (2009).

Our default mantle is thus pure MgSiO_3 in high-pressure

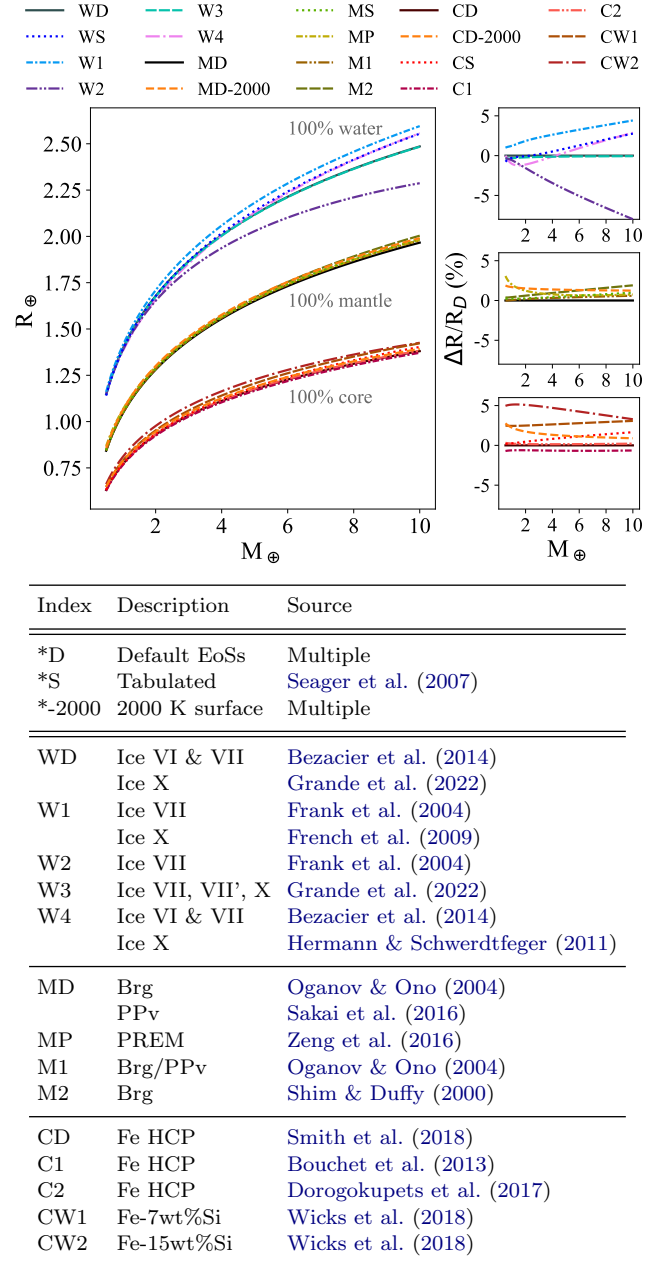


Figure 3. Left, mass-radius relationship for planets with 100 per cent of mass in either the core, mantle, or hydrosphere demonstrating many of the EoSs implemented in MAGRATHEA. Right, percent difference in final planet radii compared to our selected “default” EoSs for water (top), mantle (middle), and core (bottom). Table lists the major components of each model with 300 K surface temperature unless designated with a “-2000”. Near the surface, water planets have water and Ice VI, and hot mantle planets have silicate melt.

phases which is more SiO_2 rich than the Earth. At low pressure ($\lesssim 25$ GPa), materials such as olivine, wadsleyite, and ringwoodite are the main constituents (Sotin et al. 2007). Although we don’t include these minerals and more complex mantle chemistry at this time, we include a tabulated EOS for the mantle using the mantle properties in the Preliminary Reference Earth Model (PREM) (Stacey & Davis 2008, Ap-

pendix F) [MP]. We also have not explored additional measurements of transition curves between compositions/phases.

3.3.3 Hydrosphere/Water

Similar to the icy moons in the Solar System; exoplanets with low density are theorized to have a large fraction of mass in a hydrosphere composed of primarily high-pressure water-ice. Laboratory measurements of the thermal properties of ice at high pressures are difficult to obtain and sometimes inconsistent (Thomas & Madhusudhan 2016; Myint et al. 2017). Due to the complexity of the phase diagram of water and large uncertainties in current measurements, in the current version of MAGRATHEA, we use a simplified hydrosphere phase diagram.

At pressures below 2.216 GPa, we applied liquid water (Valencia et al. 2007a) if the temperature is above the melting curve (Dunaeva et al. 2010). Below the melting curve, we have Ice Ih (Feistel & Wagner 2006; Acuña et al. 2021) at pressures below 0.208 GPa and Ice VI (Bezacier et al. 2014) (described below) at pressures above 0.632 GPa. Ice II, III, and V, which exist at pressures in between, are currently not built-in since this layer is thin and has negligible impact on the planet. If a planet passes through these phases, the Ice VI EOS is used and the user is notified that they passed through this region. For water and Ice Ih the isothermal temperature is applied regardless the value of the *isothermal* flag.

At high pressure, our default hydrosphere layer uses Ice VII with thermal expansion (Bezacier et al. 2014). Instead of using the MGD representation, most high pressure ice studies express their result with thermal expansivity, with the exception of Fei et al. (1993) which present their result with both methods. Although the isobaric specific heat capacity c_p is necessary to determine the temperature gradient using the thermal expansivity representation (see Eq. 5), it is not reported in most studies. Similar to Zeng et al. (2021), we include a formulation using the thermal expansion representation for Ice VI and VII from Bezacier et al. (2014) with estimated c_p of 2.6 and 2.3 J g⁻¹ K⁻¹ based on Tchijov (2004).

At this time, we do not include a phase boundary or EOS for supercritical water. To avoid extrapolating the Ice VII EOS to high temperature that may lead to non-physical density, an artificial melting curve at constant temperature 1200 K is drawn in between 2.216 GPa and 30.9 GPa. The water EOS (Valencia et al. 2007a) is used as an approximation in supercritical regions and the user is notified. Lastly, the default hydrosphere transitions at 30.9 GPa to a Vinet EOS for Ice X from Grande et al. (2022) which is calculated at 300 K and returns a constant temperature throughout regardless the value of the *isothermal* flag.

The following high-pressure water-ice EOSs are also available as alternative. Third-order BME Ice VII EOS with thermal expansion and heat capacity from Frank et al. (2004) and 3rd order BME Ice VII EOS using MGD expression from Sotin et al. (2007). Other ice EOSs that implemented do not support the isentropic temperature gradient. They include Ice VII EOS along the melting curve (Frank et al. 2004) and tabulated Ice X (French et al. 2009) which combined were used in the planetary models of Zeng et al. (2016) [W1]. Our library has three fits to the 300 K Ice VII measurements from Frank et al. (2004) – their original 3rd order BME, their BME

parameters in a Vinet equation [W2], and a Vinet fit to their results. We have the three phases of ice including the transitional Ice VII from Grande et al. (2022) which was used in Huang et al. (2021) [W3]. Lastly, we have a 3rd order BME Ice X from Hermann & Schwerdtfeger (2011) [W4]. A future version of MAGRATHEA will expand upon these options with results from French & Redmer (2015), Journaux et al. (2020), and other water phases from the AQUA collection (Haldemann et al. 2020).

3.3.4 Atmosphere/Gas

The ideal gas equation of state:

$$P = \frac{\rho RT}{m_{mol}}, \quad (31)$$

and temperature relation:

$$T \propto P^{(\gamma_{gas}-1)/\gamma_{gas}} \quad (32)$$

is applied to the gas layer at all pressures and temperatures, where γ_{gas} is the adiabatic index of the gas. Combining Eq. 1, 2, 31, and 32, we have the ideal gas temperature gradient

$$\frac{dT(m)}{dm} = -\frac{(\gamma_{gas}-1)m_{mol}Gm}{4\pi r^4\gamma_{gas}R\rho}. \quad (33)$$

The temperature profile can be chosen between isothermal and isentropic by setting the number of atoms per molecule n (see Table A3), which further determines γ_{gas} . The adiabatic index γ_{gas} can be chosen from $\gamma_{gas} = \frac{5}{3}$ for monatomic gas, $\gamma_{gas} = \frac{7}{5}$ for diatomic gas, and $\gamma_{gas} = \frac{4}{3}$ for polyatomic gas. An isothermal gas EOS can be achieved by setting its $n = 0$. The temperature profile of the gas layer is determined by the properties of the gas EOS and will not be overwritten by the *isothermal* flag.

Nixon & Madhusudhan (2021) shows that self-consistently modelled atmosphere temperature profiles can be closely matched by a radiative profile above a radiative-convective boundary, and an adiabatic profile below the boundary. The pressure of the radiative-convective boundary can span from 1 bar to over 1 kbar, mainly depending on the intrinsic effective temperature and equilibrium temperature of the planet (Fortney et al. 2007). To simulate this temperature structure, the default settings for the atmosphere in MAGRATHEA include an isothermal atmosphere at $P < 100$ bar to approximate the radiative temperature gradient, and an adiabatic temperature gradient at $P \geq 100$ bar. The mean molecular weight of each gas EOS is fixed. The dependence of planetary radius on atmosphere mean molecular weight is explored in Fig. 7.

Radiative temperature profile and coupled atmosphere-interior models such as Mousis et al. (2020) are potential future avenues of work. Additionally, we anticipate that future versions of MAGRATHEA will include non-ideal atmospheres from Saumon et al. (1995) and Chabrier et al. (2019).

4 KNOWN LIMITATIONS

We design MAGRATHEA with a focus on extensibility. MAGRATHEA permits users to extrapolate EOS functions beyond the temperature and pressure bounds from experimental and theoretical works. The phase diagram can be set up

to allow a material where it would not be physical (e.g. use water in place of supercritical water). If the integrator passes through these extrapolated regions, it may find an incorrect solution or may fail to continue to integrate. Here we detail some limitations one should be aware of for each layer with our current defaults.

For the core, the liquid iron EOS from Dorogokupets et al. (2017) has no solution when the temperature of a given step gives a thermal pressure much larger than the total pressure at that step. This occurs most often when the iron core outer boundary is near the surface of the planet and at unusually high temperatures. In this case, a *no solution found* tag will be returned.

Extrapolated EOS functions may have two density solutions that satisfy the equation set at given P and T . The nonphysical solution typically has a lower density and the $\frac{\partial \rho}{\partial P} < 0$. The Newton’s method density solver may converge to the incorrect branch at a transition from a low density phase to a higher density phase because the initial density guess of the high density phase is too small. We experienced this problem in the mantle when transitioning from Si melt to Brg from Oganov & Ono (2004) at low pressures. To avoid this, if the initial density guess gives $\frac{\partial \rho}{\partial P} < 0$, a density larger than $\rho(P = 0, T = T_0)$ will be applied as the initial density guess of the Newton’s solver, instead of the density solution of the previous step.

For the hydrosphere, users should be aware that MAGRATHEA will use the water EOS and return a constant temperature even if the water would be in the vapor or supercritical phase. Because water EOS is softer and the thermal expansion is not included, the density is overestimated in supercritical regions. This may lead to a density drop at the phase transition from supercritical fluid to Ice X. Other phases without thermal parameters, such as Ice X, will return a constant temperature. Lastly, the adiabatic portion of the default atmosphere quickly rises in temperature and pressure and could force internal layers into extreme regions of the phase diagram. Ideal gas atmospheres with constant mean molecular weight are less realistic for large gas envelopes with pressures over 0.1 GPa (Zeng et al. 2021; Nixon & Madhusudhan 2021).

5 TEST PROBLEMS AND UTILITY

5.1 One-Earth mass and comparison with ExoPlex

We turn our attention to showing the outputs and utility of MAGRATHEA. We first simulate a one-Earth mass, two-layer planet with our full solver. The planet has a structure similar to Earth with 33 per cent of its mass in the core (Stacey & Davis 2008). With the default EOSs described in Sec. 3.3 and a surface temperature of 300 K, MAGRATHEA produces a planet with radius of 6166 km or $0.967 R_{\oplus}$. Fig. 4 shows the pressure, density, and temperature found throughout the planet.

Our default model differs from a detailed model of Earth in that our mantle is only Brg/PPv, the core has no lighter elements, and there is thermal equilibrium between the layers. Rather than choosing our default settings, users can choose to use an iron-silicate alloy EOS (Wicks et al. 2018) in the core and PREM in the mantle. The user can also start the

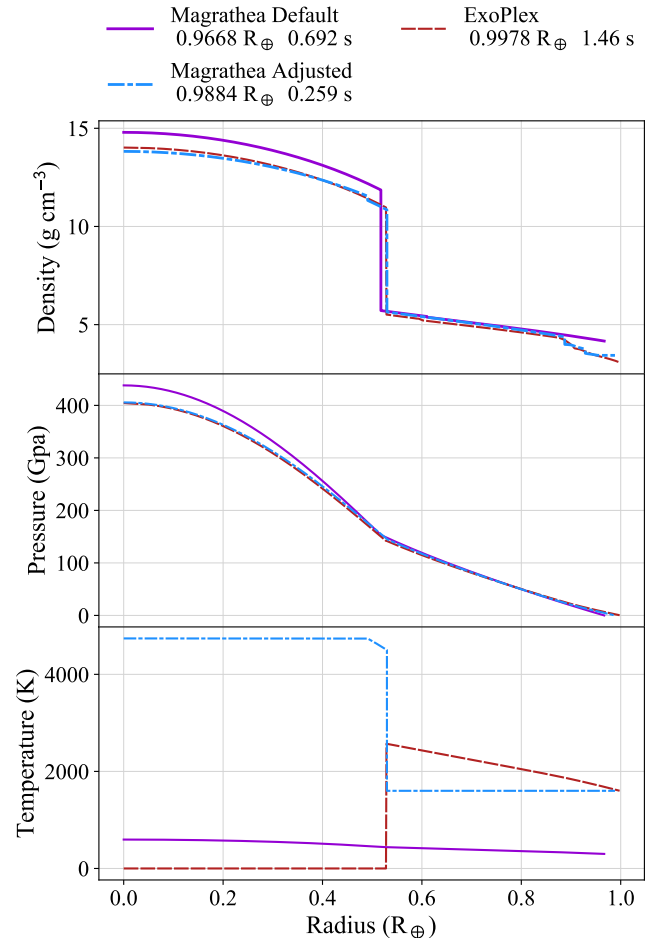


Figure 4. Density, pressure, and temperature versus radius solution for a two-layer, one Earth-mass planet with 33 per cent by mass core. The three models shown are *Magrathea Default* with hcp-Fe core (Smith et al. 2018) and Brg/PPv mantle (Oganov & Ono 2004; Sakai et al. 2016), *Magrathea Adjusted* with Fe-Si alloy core (Wicks et al. 2018) and PREM mantle (Zeng et al. 2016) and a temperature discontinuity, and the default settings in *ExoPlex*. *Magrathea Default* is set to 300 K at the surface while *ExoPlex* suggests a 1600 K mantle. Temperature is solved throughout *Magrathea Default* and in *ExoPlex*’s mantle. *Magrathea* returns no change in temperature when a phase does not have temperature parameters available while *ExoPlex* returns zero Kelvin. The planet’s radius and the average run time over 100 integrations is listed in the legend.

mantle hot at 1600 K and implement a 2900 K jump across the core-mantle boundary which creates a layer of liquid iron. This “adjusted” model, shown in Fig. 4, has a radius within 1.16 per cent of the radius of Earth ($0.9884 R_{\oplus}$).

We compare our Earth-like planets to one created with a version of *ExoPlex*⁴, an open-source interior structure solver written in Python. *ExoPlex* is used in Unterborn et al. (2018); Unterborn & Panero (2019); Schulze et al. (2021). The code uses a liquid iron core (Anderson & Ahrens 1994) and self-consistently calculates mantle phases with *Perple_X* (Con-

⁴ <https://github.com/CaymanUnterborn/ExoPlex>

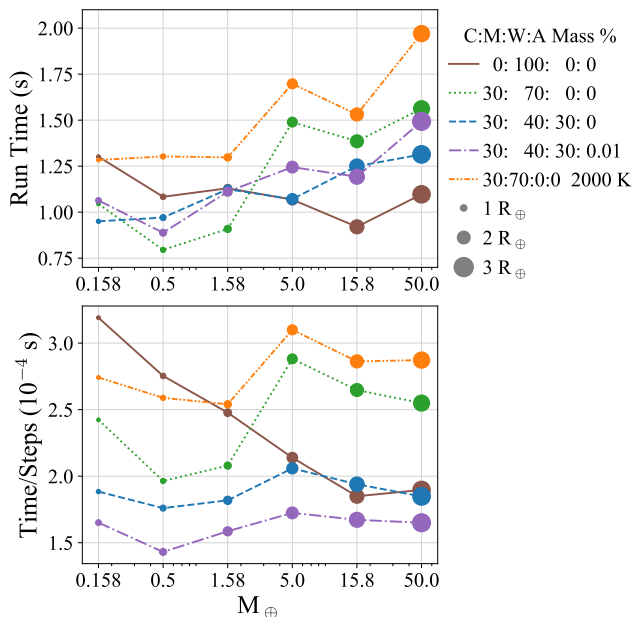


Figure 5. Plot of run time, *top*, and of run time divided total number of steps, *bottom*, for our default model across six planet masses and four compositions: 100% core, 30% core 70% mantle, 30% core 40% mantle 30% water, and 30% core 40% mantle 29.99% water 0.01% H/He atmosphere with 300 K surface temperature. The last set of planets, *orange*, have 30% core, 70% mantle, and surface temperature of 2000 K. The run time is the average time measured across 100 runs. Sizes of the markers are proportional to the planet radius.

nolly 2009) using EOS for mantle materials from [Stixrude & Lithgow-Bertelloni \(2005\)](#). *ExoPlex* comes with predetermined grids of mineralogy to capture the mantle chemistry of planets between 0.1 and $\sim 2 M_{\oplus}$.

As seen in the top panel of Fig. 4, *ExoPlex* captures changes in density in the upper mantle that our Brg/PPv model does not. *ExoPlex* creates an Earth-like planet with a radius of $0.9978 R_{\oplus}$ which is 3.2 per cent larger than our default planet and 1.0 per cent larger than our adjusted model. At least half of this difference is from a different choice in core EOS; *ExoPlex*’s innermost density is 14.0 while our default is 14.8 g cm^{-3} . The rest of the difference is *ExoPlex*’s ability to capture upper-mantle chemistry which decreases the density from our default for approximately 15% of the planet’s mass.

5.2 Run Time

We perform timing tests for both MAGRATHEA and *ExoPlex* for 100 repeat simulations of the Earth-like planet from the above section. MAGRATHEA takes on average 0.69 s to converge. This is 2.1 times faster than *ExoPlex* (1.46 s). The “adjusted” model uses a tabulated EOS for the mantle and a core EOS with no thermal information, thus most of the planet is isothermal with temperature only solved in the small region of liquid iron. This model runs 5.6 times faster than *ExoPlex*. Average times are also listed in Fig. 4.

The run time of MAGRATHEA is dependent on the mass of the planet, the user’s choice of planet model, and the error tolerances. In Fig. 5, we show run times of some typical

planet masses and compositions with our default settings. Four compositions are run with a surface temperature of 300 K. A final set has a surface temperature of 2000 K which has a region of silicate melt at the surface. The figure shows that time remains near one second, but is not easily determined by input mass and composition. Here the average run time is 1.2 seconds. The longest and shortest run times are a factor of 2.5 times slower/faster. These solutions take between 15 and 30 iterations, and the final solutions have between 520 and 840 steps in mass.

In general the run time increases with mass, though not strictly since the step size is adaptable. Crossing phase or compositional boundaries costs an insignificant amount of run time. However, certain phases may take shorter or longer to solve. The planets with atmospheres have the most compositional layers, but the ideal gas solves quickly resulting in the shortest time per step. The runs which take the longest to converge are large, hot planets with a surface of silicate melt and large mantles.

5.3 Uncertainty from equation of state

MAGRATHEA’s structure allows the user to change the EOS and test their effects on planet structure. In Fig. 3, we show how choice of EOS can change the radius of single layer planets of masses from one to ten Earth-masses. For the hydro-sphere, further discussed in [Huang et al. \(2021\)](#), new measurements of Ice VII, Ice VIII, and Ice X change the resulting planet radius from those in [Zeng et al. \(2016\)](#) by 1-5 per cent across this range of masses.

The mantle’s EOS has the smallest effect on radius in agreement with [Unterborn & Panero \(2019\)](#). Comparing the tabulated EOS from PREM to our default mantle, we find that the radii of pure mantle planets differ by less than one percent for 2-5 M_{\oplus} . Planets $< 2 M_{\oplus}$ have a larger difference in simulated radii. Capturing the density of the upper mantle is more important for these smaller planets.

HCP iron measurements generally agree when looking at bulk planet properties. However, it is uncertain how much lighter elements can be incorporated into a planet’s core. Included in MAGRATHEA are two silicate alloy EOSs from [Wicks et al. \(2018\)](#). Using a 15 per cent by weight silicate alloy EOS results in a 1 M_{\oplus} , core-only planet that differs by five per cent in radius to our default. For comparison to the above radius changes, the 1-sigma uncertainties in radii of the seven Trappist-1 planets are 1-2 per cent ([Agol et al. 2021](#)).

A single measured EOS also has uncertainties in its own parameters. MAGRATHEA includes the ability to modify an EOS at run-time to determine the effects that measurement uncertainty has on a planet radius estimates. This feature is available in *input_mode* 3, 4, and 5. It’s implementation is further detailed in Appendix B. The first two modes are limited to using the isothermal *twolayer* solver explained in Sect. 3.1.1. The *input_mode*=5 runs the full model with user specified planet composition and mass. MAGRATHEA iterates over EOS parameters from an input file and outputs the planet radius for each EOS modification.

In Table 2, we show the reported uncertainties in three experimental EOS from [Sakai et al. \(2016\)](#); [Smith et al. \(2018\)](#); [Grande et al. \(2022\)](#). We model 10 M_{\oplus} planets each with a single layer. We then modify the EOS of that layer from 1000

Table 2. EOS parameters for Smith et al. (2018), Sakai et al. (2016), and Grande et al. (2022) with uncertainty and the uncertainty in radius for a 10 M_{\oplus} single-layer planet from uncertainty in EOS parameters.

Phase	TP GPa	V_0 $\text{cm}^3 \text{mol}^{-1}$	K_0 GPa	K'_0 GPa	σ_R/μ_R for 10 M_{\oplus} %
Fe HCP	-	6.625	177.7(6)	5.64(1)	0.089
PPv	112.5(8.1)	24.73	203(2)	5.35(9)	0.19
ice-VII	-	12.80(26)	18.47(4.00)	2.51(1.51)	0.24
ice-VII _t	5.10(50)	12.38(16)	20.76(2.46)	4.49(35)	
ice-X	30.9(2.9)	10.18(13)	50.52(16)	4.5(2)	

values of the reference volume, bulk modulus, and the derivative of the bulk modulus drawn from Gaussians centered at the reported mean and using the given uncertainties. For the 100 per cent mantle and 100 per cent hydrosphere models we also draw 1000 values of the transition pressure (TP) between phases. The mantle planets have a Brg phase, but its parameters are held constant as they are derived from *ab initio* simulations. The resulting uncertainty in planet radius for each single-layer model is < 1 per cent for 10 M_{\oplus} planets. Although the choice of EOS is of more consequence to a single measurement’s uncertainty, this module allows users to investigate new measurements and their associated uncertainties.

5.4 Ternary diagram

In Section 3.3, we show mass-radius relationships for single-layer planets. However, these relationships are not unique when considering a planet with three or more layers. For a three-layer planet, ternary diagrams provide a way to visualize the radius parameter space for a planet with a certain mass (conversely we could show the various masses for a certain radius). Ternary diagrams have been used for exoplanet interiors in Valencia et al. (2007b), Rogers & Seager (2010), Madhusudhan et al. (2012), Brugger et al. (2016), Neil & Rogers (2020), and Acuña et al. (2021).

The location of a point on the equilateral triangle is given by the percentages of mass in the three layers—core, mantle, and water. The three coordinates add to a constant 100%. Since axes are interchangeable, we decide to use the same representation as Zeng & Seager (2008). With this orientation the core to mantle ratio increases along the bottom axes and the water percentage increases perpendicular to the bottom axis.

We include with MAGRATHEA, a ternary plotting python script that uses python-ternary by Harper et al. (2015). To run 5151 planets with integer percentages of mass in each layer we use MAGRATHEA’s bulk input mode (*input_mode=6*). This function requires an input file that contains each planet’s total mass and the mass fraction of core, mantle, and hydrosphere. Any mass not allocated to a layer is put into an atmosphere layer.

In Fig. 6 left each position on the triangle gives the radius of a unique one Earth-mass planet. We use planets in thermal equilibrium, a 300 K surface temperature, and with our default EOSs. This model underestimates the Earth’s radius as discussed in Sec. 5.1. A dry, one Earth-mass, and one Earth-radius planet in these simulations has approximately

a 19 per cent core mass fraction (CMF). Because of the difference between water’s and iron’s density, we can see that the radius varies most dramatically along the water-core axis. A one Earth-mass and Earth-radius planet is also consistent with having 75 per cent CMF and 25 per cent of mass in a hydrosphere.

The entire suite of simulations needed for the one Earth-mass ternary takes approximately 87 minutes to complete. The time for each run is shown in Fig. 6, right. The average run time is 1.014 seconds. In agreement with Fig. 5, time has a weak correlation with mantle percentage.

5.4.1 Ternary with atmosphere

As a final example, we add an atmosphere to our ternary planets using our default two “phase” ideal gas atmosphere. In Fig. 7, we show contour lines of constant radius. The solid lines come from Fig. 6. The 5151, one Earth-mass planets are simulated again with 0.01 per cent of their mass taken equally from the core and mantle and put it into an atmosphere layer. This is comparable to the atmospheric mass fraction of Venus. Three types of ideal gas are used with varying molecular weight.

A small atmosphere of a hydrogen/helium mixture inflates the planets by almost 1000 km for all interior structures. A 6000 km, one-earth mass planet with a small H/He atmosphere is consistent with a 2-layer planet of 20:80 mantle:core (intersection of mantle-core axis, bottom) or 5:95 water:core (intersection of core-water axis). Together, the one Earth-mass ternaries explore the radii of a large range of possible interior (plus atmosphere) structures.

6 SUMMARY

In this work, we presented MAGRATHEA, an open-source planet interior solver. MAGRATHEA is available at <https://github.com/Huang-CL/Magrathea>. Given the mass in the core, mantle, hydrosphere, and atmosphere, the code uses a shooting method to find the planet’s radius and internal conditions.

The code allows users to easily modify their planet models. Each layer is given a phase diagram where the EOS may change based on P-T conditions. Multiple formulations, both thermal and non-thermal, for the EOS are implemented. We document in this work our built-in library of EOSs and how to add new equations. Our default EOSs feature up-to-date experimental results from high-pressure physics. We show that

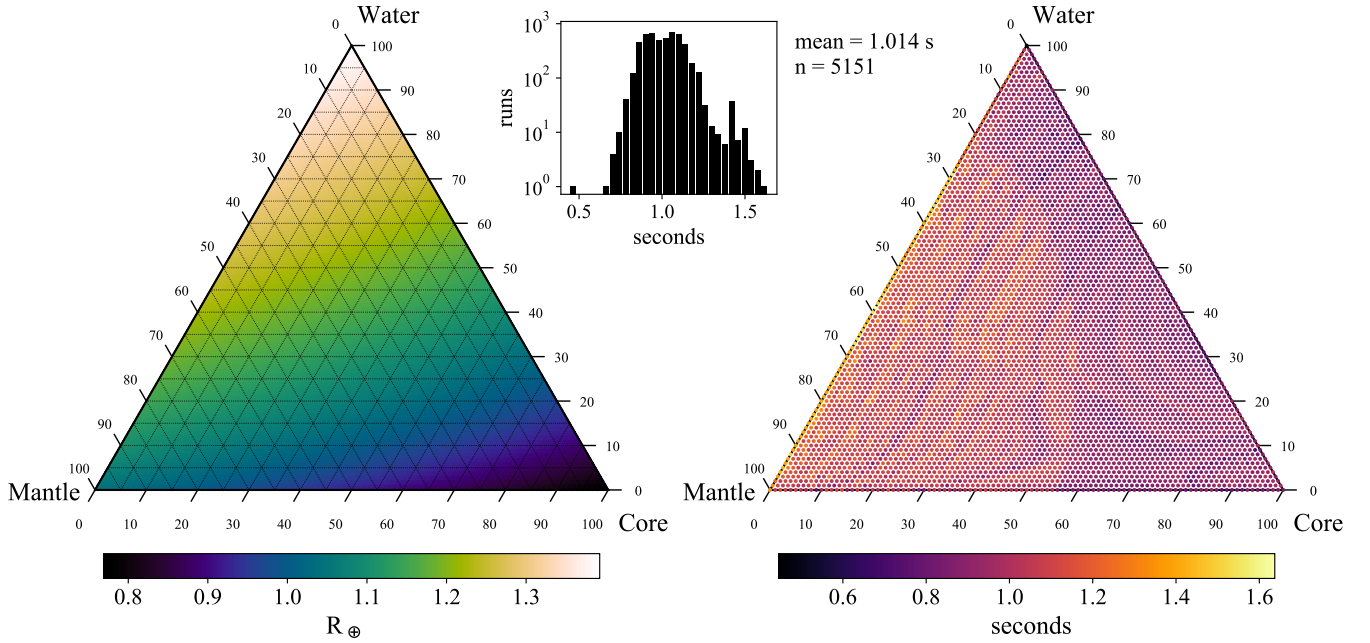


Figure 6. *Left*, ternary diagram where the axes are the percentage of mass in a core, mantle, and water/ice layer. The radius in Earth radii is shown by the color scale for 5151, one Earth-mass planets at integer percentages with our default model. Color map is interpolated between the simulations. *Right*, plot of MAGRATHEA’s run time for each planet. *Middle*, histogram of run time on a log-log scale showing mean time of 1.014 seconds. Ternary plots are generated with the python-ternary package by Harper et al. (2015) with colormaps from Van der Velden (2020).

the choice of planet model has an effect on inferences of planet composition.

The timings and results presented here show that MAGRATHEA is an efficient and useful tool for characterizing the possible interiors of planets. MAGRATHEA is currently under active development and we plan future expansions to the package. In this version, the isentropic temperature gradient is only available for the Ice VI and Ice VII in the hydrosphere. Future work includes implementing additional core alloys, upper mantle materials, more phases of water, a thermodynamic Ice X, atmosphere EOSs, and a graphically determined phase diagram. We are currently working on a planet composition finder which returns a sample of possible interiors from a given radius and mass. We encourage the community to contribute and use MAGRATHEA for their interior modeling needs.

ACKNOWLEDGEMENTS

CH acknowledge support by the NASA Exoplanet Research Program grant 80NSSC18K0569. DRR and JHS thank the National Science Foundation for their support for our work under grant AST-1910955. We thank Ashkan Salamat and Oliver Tschannerl for their important help in creating the EOS list. DRR thanks Cayman Unterborn for the use of ExoPlex.

DATA AVAILABILITY

The data underlying this article will be shared on request to the corresponding authors, or can be accessed online at our GitHub repository.

REFERENCES

- Acuña L., Deleuil M., Mousis O., Marcq E., Levesque M., Agui-chine A., 2021, *A&A*, **647**, A53
- Adams D., 1995–2001, *The Hitchhiker’s Guide to the Galaxy*. Harmony Books, New York
- Agol E., et al., 2021, *PSJ*, **2**, 1
- Anderson W. W., Ahrens T. J., 1994, *Journal of Geophysical Research*, **99**, 4273
- Anglada-Escudé G., et al., 2016, *Nature*, **536**, 437
- Batalha N. M., et al., 2011, *ApJ*, **729**, 27
- Batalha N. M., et al., 2013, *ApJS*, **204**, 24
- Baumeister P., Padovan S., Tosi N., Montavon G., Nettelmann N., MacKenzie J., Godolt M., 2020, *ApJ*, **889**, 42
- Belonoshko A. B., Skorodumova N. V., Rosengren A., Ahuja R., Johansson B., Burakovsky L., Preston D. L., 2005, *Phys. Rev. Lett.*, **94**, 195701
- Belonoshko A. B., Dorogokupets P. I., Johansson B., Saxena S. K., Koči L., 2008, *Physical Review B*, **78**, 104107
- Bezacier L., Journaux B., Perrillat J.-P., Cardon H., Hanfland M., Daniel I., 2014, *The Journal of Chemical Physics*, **141**, 104505
- Bouchet J., Mazevet S., Morard G., Guyot F., Musella R., 2013, *Physical Review B*, **87**, 094102
- Brugger B., Mousis O., Deleuil M., Lunine J. I., 2016, *ApJ*, **831**, L16
- Brugger B., Mousis O., Deleuil M., Deschamps F., 2017, *ApJ*, **850**, 93

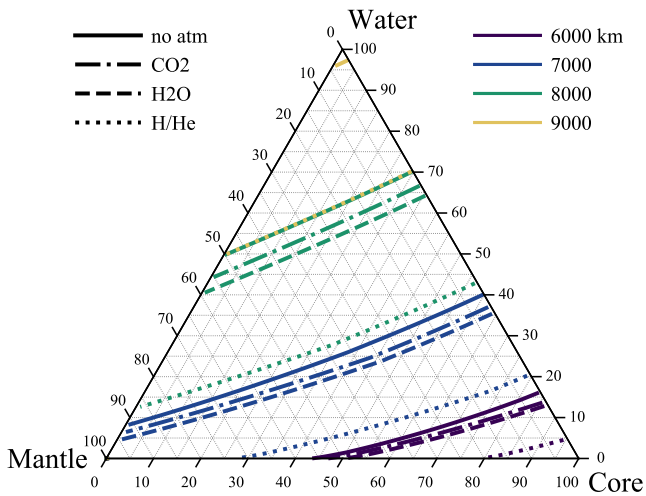


Figure 7. Core-Mantle-Water mass percentage ternary plot with colored contours of constant radius for one Earth-mass planets. Four types of planets, represented by the line-style, are calculated: one with no atmosphere and three with 0.01 per cent of their mass in an atmosphere layer. The atmosphere mass was subtracted equally from both the mantle and core to keep total mass equal to one Earth-mass. The three atmospheres have varying mean molecular weight: CO₂ with 44 g mol⁻¹, H₂O with 18 g mol⁻¹, and H/He mixture with 3 g mol⁻¹.

Chabrier G., Mazevet S., Soubiran F., 2019, *ApJ*, **872**, 51
 Choukroun M., Grasset O., 2007, *Journal of Chemical Physics*, **127**, 124506
 Connolly J. A. D., 2009, *Geochemistry, Geophysics, Geosystems*, **10**, Q10014
 Dewaele A., Loubeyre P., Occelli F., Mezouar M., Dorogokupets P. I., Torrent M., 2006, *Phys. Rev. Lett.*, **97**, 215504
 Dittmann J. A., et al., 2017, *Nature*, **544**, 333
 Dorn C., Lichtenberg T., 2021, *ApJ*, **922**, L4
 Dorn C., Khan A., Heng K., Connolly J. A. D., Alibert Y., Benz W., Tackley P., 2015, *A&A*, **577**, A83
 Dorn C., Venturini J., Khan A., Heng K., Alibert Y., Helled R., Rivoldini A., Benz W., 2017, *A&A*, **597**, A37
 Dorogokupets P. I., Dymshits A. M., Litasov K. D., Sokolova T. S., 2017, *Scientific Reports*, **7**, 41863
 Duffy T., Madhusudhan N., Lee K., 2015, in , *Treatise on Geophysics*. Elsevier, Oxford, pp 149–178
 Dunaeva A. N., Antsyshkin D. V., Kuskov O. L., 2010, *Solar System Research*, **44**, 202
 Fei Y., Mao H.-K., Hemley R. J., 1993, *J. Chem. Phys.*, **99**, 5369
 Feistel R., Wagner W., 2006, *Journal of Physical and Chemical Reference Data*, **35**, 1021
 Fortney J. J., Marley M. S., Barnes J. W., 2007, *ApJ*, **659**, 1661
 Frank M. R., Fei Y., Hu J., 2004, *Geochimica et Cosmochimica Acta*, **68**, 2781
 French M., Redmer R., 2015, *Phys. Rev. B*, **91**, 014308
 French M., Mattsson T. R., Nettelmann N., Redmer R., 2009, *Physical Review B*, **79**, 054107
 Fressin F., et al., 2013, *ApJ*, **766**, 81
 Galassi M., et al., 2009, *GNU Scientific Library Reference Manual* (3rd Ed.)
 Gillon M., et al., 2017, *Nature*, **542**, 456
 Grand Z. M., et al., 2022, *Phys. Rev. B*, **105**, 104109
 Grasset O., Schneider J., Sotin C., 2009, *ApJ*, **693**, 722
 Grimm S. L., et al., 2018, *A&A*, **613**, A68
 Hakim K., Rivoldini A., Van Hoolst T., Cottenier S., Jaeken J., Chust T., Steinle-Neumann G., 2018, *Icarus*, **313**, 61

Haldemann J., Alibert Y., Mordasini C., Benz W., 2020, *A&A*, **643**, A105
 Harper M., Weinstein B., Simon C., 2015, *Zenodo* 10.5281/zenodo.594435
 Hermann A., Schwerdtfeger P., 2011, *Phys. Rev. Lett.*, **106**, 187403
 Hirose K., Labrosse S., Hernlund J., 2013, *Annual Review of Earth and Planetary Sciences*, **41**, 657
 Huang C., et al., 2021, *MNRAS*, **503**, 2825
 Journaux B., et al., 2020, *Space Sci. Rev.*, **216**, 7
 Lay T., Hernlund J., Buffett B. A., 2008, *Nature Geoscience*, **1**, 25
 Léger A., et al., 2004, *Icarus*, **169**, 499
 Madhusudhan N., Lee K. K. M., Mousis O., 2012, *ApJ*, **759**, L40
 Madhusudhan N., Nixon M. C., Welbanks L., Piette A. A. A., Booth R. A., 2020, *ApJ*, **891**, L7
 Madhusudhan N., Piette A. A. A., Constantinou S., 2021, *ApJ*, **918**, 1
 Ment K., et al., 2019, *AJ*, **157**, 32
 Mosenfelder J. L., Asimow P. D., Frost D. J., Rubie D. C., Ahrens T. J., 2009, *Journal of Geophysical Research (Solid Earth)*, **114**, B01203
 Mousis O., Deleuil M., Agüichine A., Marcq E., Naar J., Aguirre L. A., Brugger B., Gonçalves T., 2020, *ApJ*, **896**, L22
 Myint P. C., Benedict L. X., Belof J. L., 2017, *J. Chem. Phys.*, **147**, 084505
 Neil A. R., Rogers L. A., 2020, *ApJ*, **891**, 12
 Nixon M. C., Madhusudhan N., 2021, *MNRAS*, **505**, 3414
 Noack L., Lasbleis M., 2020, *A&A*, **638**, A129
 Nomura R., Hirose K., Uesugi K., Ohishi Y., Tsuchiyama A., Miyake A., Ueno Y., 2014, *Science*, **343**, 522
 Oganov A. R., Ono S., 2004, *Nature*, **430**, 445
 Ono S., Oganov A. R., 2005, *Earth and Planetary Science Letters*, **236**, 914
 Petigura E. A., Marcy G. W., Howard A. W., 2013, *ApJ*, **770**, 69
 Poirier J.-P., 2000, *Introduction to the Physics of the Earth's Interior*
 Press W. H., Teukolsky S. A., Vetterling W. T., Flannery B. P., 2007, *Numerical Recipes 3rd Edition: The Art of Scientific Computing*, 3 edn. Cambridge University Press, http://www.amazon.com/Numerical-Recipes-3rd-Scientific-Computing/dp/0521880688/ref=sr_1_1?ie=UTF8&s=books&qid=1280322496&sr=8-1
 Rogers L. A., Seager S., 2010, *ApJ*, **712**, 974
 Sakai T., Dekura H., Hirao N., 2016, *Scientific Reports*, **6**, 22652
 Salamat A., et al., 2013, *Physical Review B*, **88**, 104104
 Santerne A., et al., 2018, *Nature Astronomy*, **2**, 393
 Saumon D., Chabrier G., van Horn H. M., 1995, *ApJS*, **99**, 713
 Schulze J. G., Wang J., Johnson J. A., Gaudi B. S., Unterborn C. T., Panero W. R., 2021, *The Planetary Science Journal*, **2**, 113
 Seager S., Kuchner M., Hier-Majumder C. A., Militzer B., 2007, *ApJ*, **669**, 1279
 Shah O., Alibert Y., Helled R., Mezger K., 2021, *A&A*, **646**, A162
 Shim S.-H., Duffy T. S., 2000, *American Mineralogist*, **85**, 354
 Smith R. F., et al., 2018, *Nature Astronomy*, **2**, 452
 Sotin C., Grasset O., Mocquet A., 2007, *Icarus*, **191**, 337
 Stacey F. D., Davis P. M., 2008, *Physics of the Earth*
 Steffen M., 1990, *A&A*, **239**, 443
 Stixrude L., Lithgow-Bertelloni C., 2005, *Geophysical Journal International*, **162**, 610
 Tchijov V., 2004, *Journal of Physics and Chemistry of Solids*, **65**, 851
 Thomas S. W., Madhusudhan N., 2016, *MNRAS*, **458**, 1330
 Tschauner O., Ma C., Beckett J. R., Prescher C., Prakapenka V. B., Rossman G. R., 2014, *Science*, **346**, 1100
 Unterborn C. T., Panero W. R., 2019, *Journal of Geophysical Research (Planets)*, **124**, 1704

Table A1. Index for types of isothermal EOS formulae

Index	Function Name	Comment
0	BM3	3rd order Birch-Murnaghan ^a
1	BM4	4th order Birch-Murnaghan
2	Vinet	
3	Holzappel	
4	Keane	
5		Empty
6		Ideal gas law
7		Interpolation
8-12	Same as 0-4	In combination with RTPress

^a Default

Unterborn C. T., Desch S. J., Hinkel N. R., Lorenzo A., 2018, *Nature Astronomy*, **2**, 297
Valencia D., O’Connell R. J., Sasselov D., 2006, *Icarus*, **181**, 545
Valencia D., Sasselov D. D., O’Connell R. J., 2007a, *ApJ*, **656**, 545
Valencia D., Sasselov D. D., O’Connell R. J., 2007b, *ApJ*, **665**, 1413
Van der Velden E., 2020, *The Journal of Open Source Software*, **5**, 2004
Wicks J. K., et al., 2018, *Science Advances*, **4**, eaao5864
Wolf A. S., Bower D. J., 2018, *Physics of the Earth and Planetary Interiors*, **278**, 59
Zeng L., Sasselov D., 2013, *Publications of the Astronomical Society of the Pacific*, **125**, 227
Zeng L., Seager S., 2008, *PASP*, **120**, 983
Zeng L., Sasselov D. D., Jacobsen S. B., 2016, *ApJ*, **819**, 127
Zeng L., et al., 2021, *ApJ*, **923**, 247

APPENDIX A: EOS STORAGE STRUCTURE

Calculating the density for a variety EOS formulae or tabulated EOS is packaged in the structure *EOS* declared in *EOS.h* and accomplished in *EOS.cpp*. The EOS of each phases needed for the calculation is an object of the *EOS* structure.

The name of pointers to *EOS* objects should be declared as an external variable in the header file *EOSlist.h* and then accomplished in *EOSlist.cpp*. To use the formula method, user should create a dictionary style two-dimensional double array with a shape of (*length*,2), where *length* is the number of provided EOS parameters. Table A1 - A3 explains all acceptable parameters. In the array, the first number of each row is the index key of the parameter according to Table A3. The second number is the value in the required unit. Not all spaces in the table need to be filled up. Not available values can be skipped. V_0 and m_{mol} are required for all types of EOSs. K_0 is required for condensed phase material. γ_0 and β are required for isentropic temperature profile and thermal expansion. The minimum required parameters can be determined by the corresponding formulae listed in Section 2.2 and default values in Table A3.

Then the EOS object pointer can be constructed using the EOS constructor that have three arguments. The first one is a string of the phase description, then is the name of dictionary style double array, and the third argument is *length*, the number of parameters provided. The name of the EOS object pointer should be the same as the one declared in *EOSlist.h*.

Table A2. Temperature profile option ^a

Index	Temperature profile calculation method
0	No temperature profile, must be isothermal
1	External entropy function
2	External temperature gradient function ^b
3	Ideal gas
4	EOS is fitted along isentrope ^c
5-7	Isentropic curve ^d
8	RTPress
9	Thermal expansion ^e

^a Only need to specify this option in the input list if using external entropy function or external temperature gradient function (index 1 or 2), or the EOS is fitted along isentrope (index 4). Otherwise, the code will determine the option based on the parameters provided.

^b The only method to set the gradient is using the modify_extern_dTdP function.

^c Selecting this option requires parameters to calculate Debye thermal pressure in Eq. 15. The isentrope temperature profile is enforced to the phase with this option even the isothermal option is chosen for the planet solver.

^d Selecting this option requires parameters to calculate Debye thermal pressure in Eq. 15.

^e Selecting this option requires α and c_p .

The following shows the EOS of post-perovskite phase of MgSiO_3 in the code as an example.

Code in *EOSlist.h*

```
\\Declare pointers to EOS objects
extern EOS *Si_PPv_Sakai;
```

Code in *EOSlist.cpp*

```
\\Dictionary style double array
double Si_PPv_Sakai_array[][2]
= {{0,4}, {1,24.73}, {2,203}, {3,5.35},
   {5,mMg+mSi+3*mO}, {8,848}, {9,1.47},
   {10,2.7}, {11,0.93}, {15,5}};
```

```
\\ Create a EOS object pointer
EOS *Si_PPv_Sakai
= new EOS("Si_PPv_(Sakai)",
Si_PPv_Sakai_array,
sizeof(Si_PPv_Sakai_array)/2
/sizeof(Si_PPv_Sakai_array[0][0]));
```

For RTPress EOS framework, a list of additional polynomial fitting terms b_n , in the unit of erg mol^{-1} , are required (see Eq. 21). A separate array and its size *blength* are used to construct a RTPress EOS object. The following shows the EOS of liquid MgSiO_3 as an example.

Code in *EOSlist.h*

```
extern EOS *Si_Liquid_Wolf;
```

Code in *EOSlist.cpp*

```
double Si_Liquid_Wolf_array[][2]
= {{0,10}, {1,38.99}, {2,13.2}, {3,8.238},
   {5,mMg+mSi+3*mO}, {9,0.1899}, {10,0.6},
   {12,-1.94}, {15,5}, {17,3000}};
double Si_Liquid_Wolf_b[]
= {4.738E12, 2.97E12, 6.32E12,
```

Table A3. List of EOS parameters

Index	Variable	Unit	Comment
0	EOS formula type		See table A1
1	V_0	$\text{cm}^3 \text{mol}^{-1}$	Molar volume at reference point
2	K_0	GPa	Bulk modulus
3	K'_0		Pressure derivative of the bulk modulus. Default 4
4	K''_0	GPa^{-1}	Second pressure derivative
5	m_{mol}	g mol^{-1}	Molar mass
6	P_0	GPa	The minimum pressure, corresponding to V_0 . Default 0
7	Θ_0	K	Fitting parameter of Einstein or Debye temperature (see Eq. 14). Default 1
8	γ_0		Fitting parameter of Grüneisen parameter (see Eq. 13 and 19)
9	β		Fitting parameter of Grüneisen parameter (see Eq. 13 and 24)
10	γ_∞		Fitting parameter of Grüneisen parameter (see Eq. 14). Default 2/3
11	γ'_0		Volume derivative of the Grüneisen parameter (see Eq. 19)
12	e_0	10^{-6}K^{-1}	Electronic contribution to Helmholtz free energy (see Eq. 12). Default 0
13	g		Electronic analogue of the Grüneisen parameter (see Eq. 12)
14	n		Number of atoms in the chemical formula ^a Default 1
15	Z		Atomic number (number of electron)
16	T_0	K	Reference temperature for the thermal pressure (see Eq. 7). Default 300
17	α_0	10^{-6}K^{-1}	The zeroth order coefficient of thermal expansion at a reference pressure P_0
18	α_1	10^{-6}K^{-2}	The first order coefficient of thermal expansion at a reference pressure P_0
19	ξ		Power law index in the coefficient of thermal expansion (Eq. 28) • Default 0
20	c_{p0}	$10^7 \text{erg g}^{-1} \text{K}^{-1}$	Specific heat capacity at constant pressure (see Eq. 30)
21	c_{p1}	$10^7 \text{erg g}^{-1} \text{K}^{-2}$	Coefficient for specific heat capacity (see Eq. 30)
22	c_{p2}	10^7erg K g^{-1}	Coefficient for specific heat capacity (see Eq. 30)
23	Debye_approx		Positive number for Debye, otherwise Einstein
24	thermal_type		See table A2

^a Number of atoms in the volume of V/N_A , where N_A is the Avogadro constant. The n of ideal gas is the number of atoms per molecule for the purpose of adiabatic index. For example, $n = 2$ for collinear molecules e.g. carbon dioxide. Isothermal atmosphere can be achieved by setting $n = 0$.

$-1.4\text{E}13, -2.0\text{E}13\}$;

```
EOS *Si_Liquid_Wolf
= new EOS("Si_liquid_(Wolf)",
Si_Liquid_Wolf_array, Si_Liquid_Wolf_b,
sizeof(Si_Liquid_Wolf_array)/2
/sizeof(Si_Liquid_Wolf_array[0][0]),
sizeof(Si_Liquid_Wolf_b)
/sizeof(Si_Liquid_Wolf_b[0]));
```

APPENDIX B: MODIFY A BUILT-IN EOS IN RUNTIME

Once an EOS is built-into the code, it is possible to modify its parameter in the run-time without the request of recompile the full code. This feature is useful to rapidly repeat runs with similar EOS parameters, for example when study the impact of the uncertainty of EOS parameters on planet size.

One can modify one parameter of a EOS using

```
void EOS::modifyEOS
(int index, double value),
```

where index is listed in Table A3. Or using

```
void EOS::modifyEOS
(double params[][2], int length)
```

to modify multiple parameters at once, where length is the number of parameters that need to be modified and params is the dictionary style 2D double array (see Appendix A).

If the phase boundary between phases are independent of temperature and purely determined by the pressure, e.g. the phase boundary commonly assumed between ice VII and ice X, the phase transition pressure within a layer can also be modified using

```
void set_phase_highP
(int k, double *start_pressure,
EOS** phase_name),
```

where k is the number of phases involved in this modification, *phase_name* is an array of EOS pointers of these k phases, and *start_pressure* is the array of phase transition pressures between these k phases in the unit of GPa with a length of $k-1$. If the name of the first EOS in the *phase_name* EOS list matches the name of one of the original EOS of this layer, the first input EOS will replace the original one. Otherwise, the first EOS in the list will be ignored in the calculation.

External EOS function, entropy function, or temperature gradient function can be modified using

```
void modify_extern_density
(double (*f)(double P, double T)),
void modify_extern_entropy
(double (*g)(double rho, double T)),
```

or

```
void modify_dTdP
(double (*h)(double P, double T))
```

respectively.