Johann Wolfgang von Goethe-Universität

Frankfurt am Main

Frankfurt Institute for Advanced Studies

Constructing the Equation of State for Compact Stars

Bachelor thesis

Jan Röder

Supervisor and first examiner:
Prof. Dr. Stefan Schramm

Second examiner:
Prof. Dr. Armen Sedrakian

Contents

1 Introduction

To this day, the nature of the equation of state (EOS) for neutron stars is highly debated. Previously discussed EOS have emerged from various approaches to strong interaction physics.. Each one has different parameters determining its impact on calculations.
For the past years, it has been common to derive an EOS from a model and then use it to construct a mass-radius relation (MRR). That way, it was also somewhat possible to investigate what impact the state of matter in different regions of the star has on a whole sequence. Since science lacks observational information about neutron star radii it has only been possible to put constraints on models.
This method of constructing the MRR has one major limitation: an EOS corresponds to a single model. This motivates the model-independent construction of the MRR, with a minimum number of constraints and parameters while maintaining the highest possible physical relevance.
The approach chosen in this thesis inverts the above process. Instead of using an EOS to calculate masses and radii, the latter now take over the role as input parameters. For compact stars up to a certain mass, the EOS will be assumed to be known well enough. Above that mass a numerical reconstruction will take place, determining the EOS from an MRR. Using the MRR as input leaves us with only the low-density region to impose an EOS in; we do in fact have the possibility to model this region sufficiently well. If future measurements grant a decent number of data points on the MRR, an inverse algorithm has the potential to probe the highest densities in the universe without relying on a specific model. Due to the unique relation of MRR and EOS this would mean a significant contribution to the investigation of high-density phase diagrams of, for example, quark matter. The enormous amount of energy needed to create matter at such densities exceeds everything possible by humanity; therefore a neutron star would be a welcome laboratory. The Neutron Star Interior Composition Explorer (NICER) experiment, located at the International Space Station, is dedicated entirely to neutron star measurements. Measuring X-ray emissions directly from the star's surface and the rotational period, the radius can be determined with a precision of a few hundred meters. Hopefully, a coarse MRR can be drawn with actual data. At that point, EOS reconstruction will likely become a crucial part of neutron star physics.

2 Theoretical background

2.1 Compact stars

At some point in a star's life, it is no longer capable of maintaining nuclear fusion in its core. Hydrogen burning first moves to a shell around the central region, further growing the previously produced helium core up to the point where helium burning ignites. Depending on the star's mass, this process can either repeat until an iron core is formed or stop at some fusion product before that. In the latter case, i.e. for star masses up to $8\,M_\odot$, the outer envelopes will escape the core and form a planetary nebula. What remains is a white dwarf, with an upper mass limit given by Chandrasekhar of about $1.4\,M_\odot$ [1]. White dwarfs are the first known family of compact objects or -stars. Their interior structure depends on its progenitor star, hence their EOS is not universally defined ([2], p. 91). Their size is of order 10^3 km, temperatures usually reside in the 1 eV ($\sim 10^4$ K) domain but can be one or two orders of magnitude higher ([2], p. 90).
If the fusion processes have produced an iron core, it will be compressed to densities exceeding multiple times the nuclear saturation density when the star turns into a supernova. The remnant left behind is called neutron star, formed out of the iron core. Neutron stars therefore always have the same matter composition and can be described by one EOS universally. For white dwarfs on the other hand, composition varies depending on which burning phase the star's life ended. The EOS for each white dwarf is therefore unique.
Both kinds of compact stars are subject to fast neutrino and photon cooling. About one minute after the supernova explosion a neutron star reaches a temperature of 1 MeV ($\sim 10^{10}$ K); after a few million years temperatures will have dropped several orders of magnitude below that. On the nuclear scale compact stars can therefore be treated as cold.

2.2 The Tolman-Oppenheimer-Volkoff equations

The structure of compact stars will be determined by the Tolman-Oppenheimer-Volkoff equations (TOV). To make analytical and numerical calculations easier, the unit system is chosen to be $c = G = 1$, so that every unit is a power of length. The derivation is based on the assumption that the star matter can be described as a perfect/ideal fluid. Further, the space-time shall not evolve in time, therefore staying spherically symmetric (i.e. it is stationary). In terms of the energy-

momentum tensor we are left with:

$$T_{\mu\nu} = (\varepsilon + P)\, u_\mu u_\nu - P g_{\mu\nu} \tag{1}$$

Where ε is the energy density, P is the pressure and u_μ is the 4-velocity of the rest frame.
Spherical symmetry leads to a certain form of the metric,

$$ds^2 = e^{2\Phi} dt^2 - e^{2\lambda} dr^2 - r^2(d\theta^2 + \sin^2 d\phi^2) \tag{2}$$

containing spherical coordinates r, θ, ϕ and a time-like coordinate t. Equation (2) becomes the line element for a flat space-time if the exponents in the first two coefficients become zero, from which follows the role of the exponent terms as curvature description in the radial and the time "directions". To be precise, Φ is directly related to gravitational redshift, and λ determines the change of proper length due to curvature.
This gives us the energy-momentum tensor components:

$$T_{\mu\nu} = diag(\varepsilon e^{2\Phi},\ P e^{2\lambda},\ P r^2,\ P r^2 sin^2(\theta)) \tag{3}$$

One now imposes conservation of energy and momentum,

$$\nabla_\nu T^{\mu\nu} = 0 \tag{4}$$

and puts $T_{\mu\nu}$ into the Einstein equations (5) together with the metric, the components of which can be read from (2).

$$R_{\mu\nu} - \frac{1}{2} g_{\mu\nu} R = 8\pi\, T_{\mu\nu} \tag{5}$$

Here, $R_{\mu\nu}$ is the Ricci tensor and R is the Ricci scalar. Both can be calculated from the metric $g_{\mu\nu}$. Before one can get to work to get the actual TOV equations, the metric exponential functions have to be calculated.
One obtains, finally, the full TOV equation:

$$\frac{dP}{dr} = \frac{(\varepsilon + P)(m + 4\pi r^3 P)}{2mr - r^2} \tag{6}$$

Together with a second equation for the mass:

$$\frac{dm}{dr} = 4\pi r^2 \varepsilon \tag{7}$$

2.3 The equation of state and neutron star structure

Most generally, an EOS determines the relation between a set of quantities that future dynamics of a system can be extrapolated with. Such a set is often referred to as "state variables". Independent of the model applied to a certain type of matter, an EOS takes the form

$$f(p, V, T) = 0$$

with, in this case, pressure p, Volume V and Temperature T. The nature of f (the actual EOS) is then derived from a model (e.g. nuclear or quark) to properly describe the state of matter at hand, be it a fluid, solid or of an exotic type. Here, starting on the outside of a neutron star, states are still in or not far from the range of what can be experimentally observed. Towards the inner regions, exotic states and their impact on the EOS are still subject to investigation. A number of nuclear or fluid EOS have been proposed, each including different aspects like interactions and states of matter, however it remains unclear how exactly a neutron star is built up. While a single EOS for the whole neutron star would be of great interest, one usually determines the EOS piecewise for each density region separately, imposing a phase transition between the layers. For a brief collection of models, see [3].

The lack of information from direct measurement does not particularly improve this situation. However, for the first time, it might be possible to deduce information about the interior of a neutron star through data on neutron star radii obtained by the NICER experiment.

Generally, a neutron star can be divided into zones: The inner and outer core, with inner and outer crust above it. These zones are usually defined through their range of densities and have fundamentally different properties. For example, (one finds the original version of this proposal in [3] and [4]), with $[\rho] = \mathrm{g/cm^3}$:

$\rho \leq 10^6$ The *surface* is the outermost region of the star, therefore temperature and magnetic fields can have a big impact on the equation of state here. The surface may partly consist of accreted matter of unknown composition, followed by a layer of iron that marks the beginning of the actual neutron star. In this region, the EOS can be approximated by the EOS of Baym et al. [5] (BPS). BPS bases calculations on a lattice configuration of nuclei as expected from a lower-density domain. This leads to significant error with respect to the ground state

neutron numbers of neutron-rich nuclei further below the surface, as their masses could not be measured with sufficient precision at the time [5] [6].

$10^6 \leq \rho \leq 4.3 \cdot 10^{11}$ Descending into the *outer crust* a Coulomb lattice forms within a relativistic electron gas. Due to charge neutrality, $n_{electrons} = Z \cdot n_{nuclei}$; for the pressure holds $P = P_e + P_L$ [6], where P_L resembles a correction made to account for Coulomb interactions. Upon reaching higher densities, neutronization becomes energetically favorable. In order to determine the nucleus contribution to the energy of one Wigner-Seitz cell in the lattice, BPS subtract electron rest energies from the atomic masses. The difference between BPS and Haensel et al. [7] (HP) is the model for electron screening effects. BPS use the electron binding energies as such a model, while HP subtract those as well and impose electron density uniformities [6] instead.

$4.3 \cdot 10^{11} \leq \rho \leq (2 - 2.4) \cdot 10^{14}$ In the *inner crust* neutron-rich nuclei form a lattice; as the density increases, neutron drip takes place. Now, there is not only an electron gas, but also a superfluid neutron gas formed by the dripped-out neutrons that could not be bound by the nuclei anymore. A density estimate for the drip to start is performed in [6] starting from

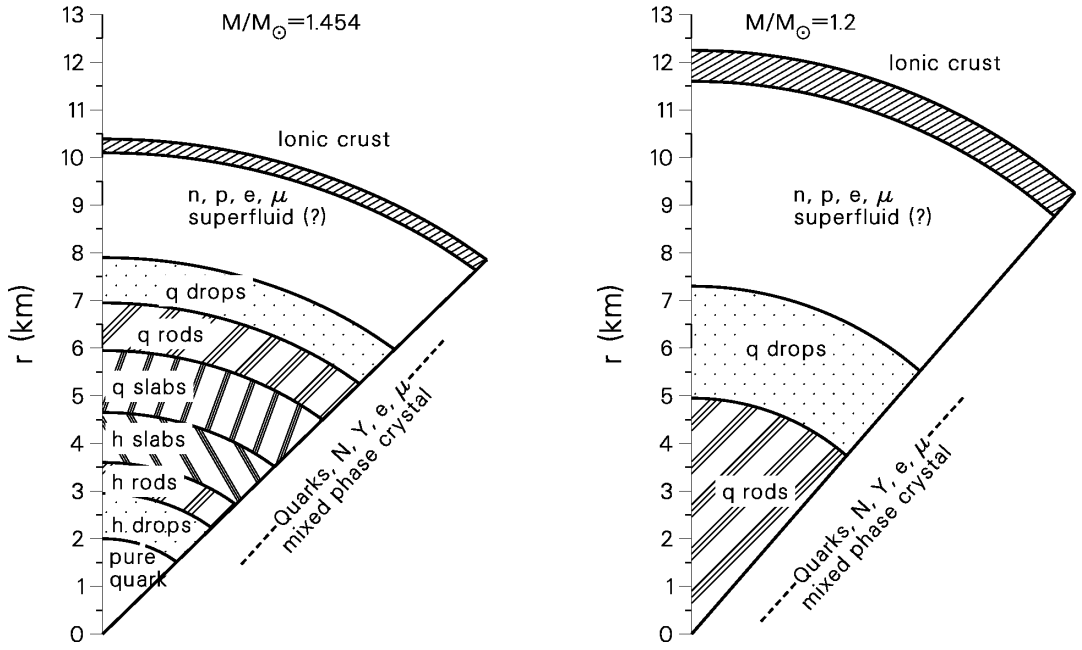
$$E_N(A, Z)/A \simeq E_0 + S_0\delta^2$$

which is a heavily simplified nuclear mass formula. Terms were neglected until only the symmetry energy S_0 ("symmetry" here means an equal number of protons and neutrons) and the energy per nucleon in a symmetric system E_0 are left. $\delta = (N - Z)/A$ is also a remnant from the mass formula. At neutron drip, the chemical potential $\mu_n = E_0 + (2\delta + \delta^2)S_0$ becomes zero and therefore $\delta_{\mathrm{ND}} = \sqrt{1 - (E_0/S_0)} - 1$. From the beta equilibrium condition and an estimate for the electron chemical potential the drip density can be estimated to about $\rho_{\mathrm{ND}} \simeq 2.2 \cdot 10^{11}\,\mathrm{g/cm^3}$.
As for the EOS, at roughly neutron drip density it is dominated by strong nucleon-nucleon interactions. There are various types of calculations and models for such a many-body system; for example, in [8] Douchin and Haensel use the Skyrme-Lyon (SLy) forces for nucleon-nucleon interaction within a liquid drop model. Likely, in this bottom region of the crust, nuclei are shaped differently to minimize their surface energy. This was first proposed by Ravenhall et al. 1983 [9], where three basic shapes were used for demonstration. In the domain of the low-density phase,

the higher-density phase should form spherical bubbles that turn into rods when descending further into the star. Eventually, both phases should be mixed to equal parts, nuclear matter arranged in planes. From here the process is "reverted"; first, rods of the first phase emerge, turning into bubbles again until only the second, high-density phase is present. This phase transition is often referred to as pasta phase due to the similarities between shapes.

$(2-2.4)\cdot 10^{14} \leq \rho \leq \rho_{core}$ The *outer core* or *neutron fluid* does no longer consist of a lattice. Remaining charge neutral, this region is dominated by a neutron superfluid along with a proton superfluid. The charge of the latter is evened out by electrons. In the above discussed liquid drop/SLy model, the crust-core transition appears in the EOS as a (weak) first-order phase transition. The jump in density is in the range 1%.
The ground state of nuclear matter can be calculated from the nuclear Hamiltonian [6]. The EOS is determined a little easier; starting from total energy density together with conditions for fixed baryon density and electrical neutrality, Lagrange multipliers give the equilibrium conditions for the chemical potentials. For neutrinos these can be set to zero as they escape the star. Evaluating the first law of thermodynamics at equilibrium leads to the pressure. This calculation can be extended to account for particle fractions varying with density.

$\rho \geq \rho_{core}$ Within the *inner core* lies the biggest mystery. One has no observational information about what processes happen at densities exceeding $10^{15}\,\mathrm{g/cm^3}$. There are various theories about the state of matter in this region; possible are quark core formation or other types of phase transition. As a simple model, $npe\mu$ matter can be extended with a hyperon fraction. This would soften the EOS due to a pressure drop. High-energy neutrons can convert to hyperons, resulting in a lower energy state (which contributes to the pressure) but a higher mass [6].



(a) Scheme of a star with enough mass to form a pure quark core after undergoing a full pasta phase transition as described above

(b) Lighter star, quark matter rods are the endpoint of the transition here

Figure 1: Schematic neutron star structure for different masses; h/q: hadronic/quark matter, taken from [10].

2.4 General constraints

There are two major conditions for any equation of state to be physically acceptable. For one, it cannot be *ultrabaric*, i.e. $P > \epsilon$. Sometimes this is confused with the superluminal limit even though an EOS might be ultrabaric and still subluminal. A physically relevant EOS cannot be superluminal either, the condition reads $dP/d\epsilon = (v/c_s)^2 < 1$ with effective speed of sound c_s. The two terms are sometimes confused as an EOS fulfilling both conditions initially will never cross the ultrabaric line. If its slope increases so that $dP/d\epsilon > 1$, the EOS becomes *superluminal*. While a superluminal EOS is widely thought to be acausal, that is not necessarily the case. In general neither Lorentz invariance nor causality explicitly forbid $v > c_s$ [6], but such EOS all have other significant disadvantages leaving them irrelevant.

2.5 Stability, mass-radius relation and third family

The term *stability* is generally used in the context of stability towards perturbations, i.e. radial oscillation modes. Stability can be seen in the mass-radius relation: If the curve reaches a critical point, the slope changes sign and the lowest stable mode switches its stability. Reading from the right, a counter-clockwise change of slope at a critical point leads to instability of a mode. This is visualized in figure 2 for three families. At points A, C and E, the maximum masses and

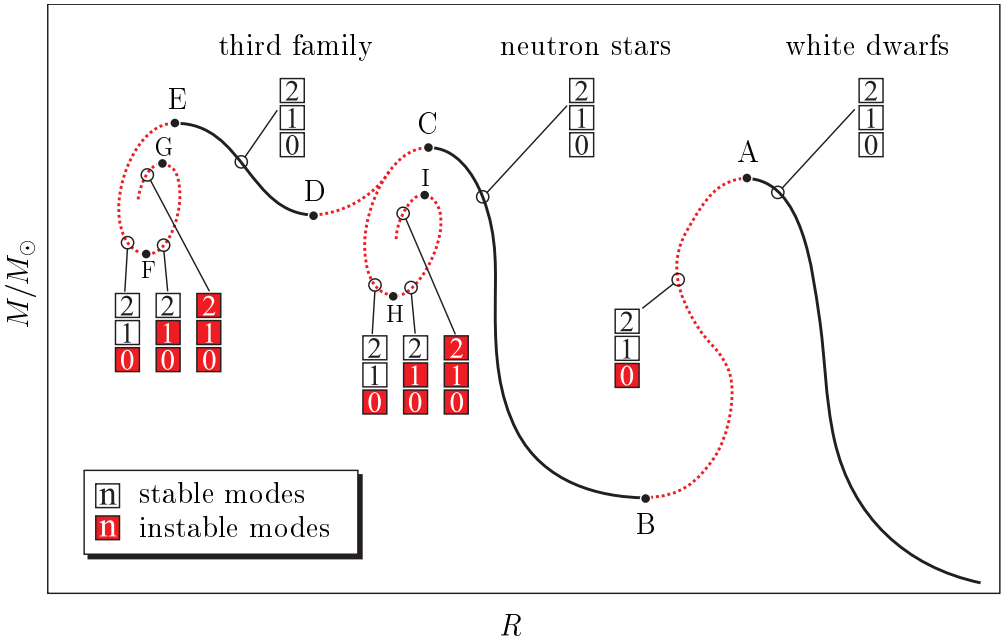


Figure 2: Mass-radius relation with a third stable branch and stability of lowest modes, taken from [11].

minimum radii for each family, the fundamental mode becomes unstable ($n = 0$). n is the number of radial nodes. When a new family starts with its lowest mass, $n = 0$ regains stability. A branch is required to have all modes stable against oscillation to be physically relevant as a family.

A branch usually ends when the EOS becomes sufficiently soft. A phase transition can then lead to a sudden increase in slope, causing the MRR to rise again, to a new branch. For example, a white dwarf is stabilized by the Pauli principle; if it crosses a certain mass limit, it collapses to a neutron star until strong interaction and degeneracy block further compactification. In figure 2 the neutron star branch hosts mixed phase cores, i.e. nuclear along with quark matter. Their high-mass twins (more below) do have a mixed phase region, but their core consists of pure

quark matter. The interior structure is heavily influenced by the EOS for all three regions, while the hadronic phase EOS is assumed to be known relatively well (as discussed previously).

In 1966, Wheeler showed that, if the EOS was smooth, there would only remain a white dwarf and a neutron star branch in the MRR that would contain stable configurations. However, in 2000 Glendenning and Kettner discovered a class of EOS that in contrast to Wheeler's work produced a third stable branch in the MRR [12]. These stars would have the same masses as neutron stars (depending on the exact EOS, this might hold for a certain mass window) but significantly smaller radii. Therefore, these stars are generally referred to as "high mass twins" or "twin stars".

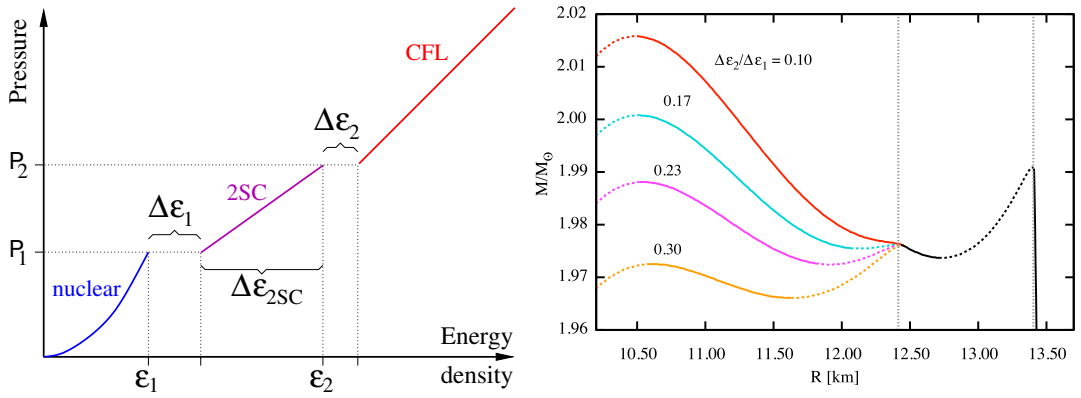
The main arguments against a third family would be instability to oscillations (resulting in explosion oder black hole formation) and distribution of Fermi pressure among more flavors [12]. Most theories resolve around quark core formation and a pasta-like phase transition from hadronic/mixed phase to quark matter analogue to the zone between crust and core.

The first to discover the general possibility of a third family was Gerlach in 1968 [13]; in the process of investigating his theory that some kind of sub-nuclear density interaction may have a significant impact on the EOS and MRR he was also the first to explore an inverse approach to stellar structure calculation. For more on that, see 2.8.

2.6 A fourth compact family?

In 2017, M. Alford and A. Sedrakian were the first to argue what effects a multiple phase QCD diagram would yield for new families of compact stars ([14] and references therein). Up to that point a quark core was always assumed to consist of only one phase.

In order to construct an MRR Alford and Sedrakian put together a piecewise EOS consisting of one for nuclear (hadronic) matter up to saturation density and one for above densities. The latter part is allowed to have two first-order phase transitions, with a constant speed of sound per quark phase. The EOS then looks as follows: From figure 3 a) four of the six parameters determining the EOS can be read off: the transition pressures $P_{1,2}$ as well as $\Delta\varepsilon_{1,2}$, the energy density jumps. The remaining parameters are the phases' sound speeds. Figure 3 b) shows an MRR for four different energy density jump ratios and stiff quark phase EOS. The sound



(a) Constructed EOS (b) Corresponding MRR

Figure 3: Constructed EOS and MRR for a second quark matter phase transition. Chosen phases are stiff two-flavor color superconducting (2CS) and color-flavor-locked (CFL). 2CS means only two flavors exist and color can be conducted while after the transition to CFL color is locked like the flavor. Taken from [14].

speeds s_1 and s_2 have been set to 0.7 and 1, respectively. For jump ratios above of about 0.15, a fourth branch of compact stars separates from the twin stars located around 12.5 km. White dwarfs are not shown, the very right branch is made of neutron stars. Within a certain ratio range triplet configurations arise: one mass, but three possible radii, depending on the star's composition. This is well visualized in fig. 4. If the phase transitions are sufficiently weakly first order, no instability arises, leaving the MRR with an extended twin star branch. From there, the fourth family emerges, completely vanishing when both transitions are strong enough, making both quark phases unstable.

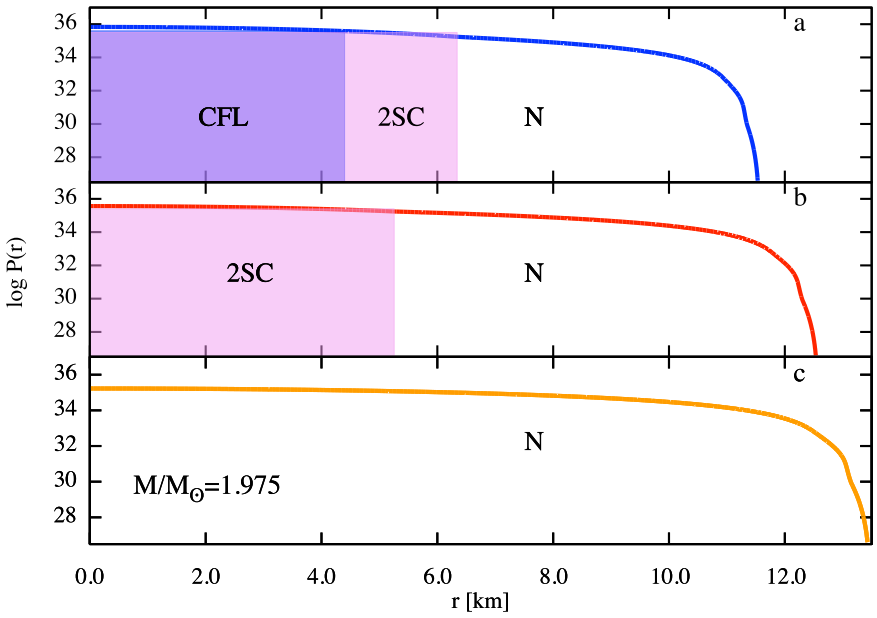


Figure 4: States of matter for a triplet configuration. $M = 1.975\,M_{\odot}$, $\Delta\varepsilon_1/\Delta\varepsilon_2 = 0.23$. Pressure in dyn/cm^2, N stands for Nuclear Phase. Taken from [14].

2.7 Fermi gas equation of state

The Fermi gas EOS will serve as a test case for the reconstruction algorithm (the constant in the polytrope will be set arbitrarily when using it as a first test). For a detailed version of this derivation, see [2].
Here, $p = k$ ($c = 1$) and therefore $E = \sqrt{k^2 + m^2}$. With that, the energy density, pressure and density become:

$$\varepsilon = \frac{\gamma}{2\pi^2} \int_0^{k_F} k^2 \sqrt{k^2 + m^2}\, dk \tag{8}$$

$$p = \frac{\gamma}{6\pi^2} \int_0^{k_F} \frac{k^4}{\sqrt{k^2 + m^2}}\, dk \tag{9}$$

$$\rho = \frac{\gamma}{2\pi^2} \int_0^{k_F} k^2\, dk \tag{10}$$

Setting the chemical potential to $\mu = \sqrt{k_F^2 + m^2}$ (the Fermi energy), the solutions to these integrals read

$$\varepsilon = \frac{1}{4\pi^2} \left[\mu k_F \left(\mu^2 - \frac{m^2}{2} \right) - \frac{m^4}{2} \ln \left(\frac{\mu + k_F}{m} \right) \right] \tag{11}$$

$$p = \frac{1}{12\pi^2} \left[\mu k_F \left(\mu^2 - \frac{5m^2}{2} \right) - \frac{3m^4}{2} \ln \left(\frac{\mu + k_F}{m} \right) \right] \tag{12}$$

$$\rho = \frac{k_F^3}{3\pi^2} \tag{13}$$

where $\gamma = 2$, the degeneracy factor for fermions due to spin orientation. The Pauli principle is therefore directly included. Charge neutrality can be imposed, and chemical equilibrium can be obtained as

$$\mu_n = \mu_p + \mu_e \tag{14}$$

with chemical potentials for neutrons, protons and electrons.
Numerically, it is convenient to calculate limits of equations (11) - (13) for the case of high and low densities, i. e. $k_F \gg m$ or $k_F \ll m$. Our test case EOS will resemble very low densities. Expanding (11) in k_F/m one obtains

$$p \sim K \cdot \rho^{5/3} \tag{15}$$

which is a polytropic type of equation. By absorbing densities into the energy density, one can get

$$p = K \cdot \varepsilon^{\gamma} \tag{16}$$

which is common to use for simple star models, especially white dwarfs.

2.8 The inverse problem

The "standard" problem of stellar structure is an integration of the TOV equations with a given EOS and central pressure, determining the total mass and radius (values of the variables at the surface). The inversion of this "standard" problem was originally brought up by Gerlach in 1968, in a paper investigating the possibility of a third stable family [13]. Another well-established method was developed by Lindblom. Both methods shall be recalled here.
Just like any other published approach to the problem, Gerlach assumes the EOS to be known below a certain density. In order to determine the changes in density and mass for the core region of unknown type for the next configuration, he sets up variations for the TOV equation (6):

$$\frac{d\Delta\rho}{dr} = A(r)\Delta\rho + B\Delta m \tag{17}$$

$$\frac{d\Delta m}{dr} = 4\pi r^2 \Delta\rho \tag{18}$$

Note that Gerlach uses ρ instead of ϵ in his equations. From here he solves the inverse problem in two steps. First, he requires the changes made to a star's surface in terms of radius and mass to be related to changes at the beginning of the core region, here in terms of density and mass. Next, he specifically calculates

$\Delta\rho$ and Δm for the core, which are required to smoothly fit to the core surface. Gerlach puts the above equations in matrix form and includes an iterative method, together with proper matching conditions.

Lindblom on the other hand presents an MRR-EOS-pair to be uniquely linked by an abstract map [15]. This gives rise to the possibility of using that map in the other direction, i.e. from MRR to EOS. His "traditional" solution of the inverse problem works as follows:

Instead of integrating the TOV equations from the core towards the surface with initial values $p = p_c$ and $r = 0$, Lindblom uses $M = m(R_{surface}) = m(R)$ and R as starting point of the calculation. Also, the EOS is assumed to be known up to some pair $\{\epsilon_i, p_i\}$ corresponding to a star $\{M_i, R_i\}$ (as Gerlach did). Looking at the next star, $\{M_{i+1}, R_{i+1}\}$ are set as initial values and integration takes place until $\{\epsilon, p\}$ reach $\{\epsilon_i, p_i\}$. At this point, a core $\{m_{i+1}, r_{i+1}\}$ is left to be integrated. Expanding the TOV equations in this small region as a power series and inverting that, $\{\epsilon_{i+1}, p_{i+1}\}$ can be obtained as a function of previously calculated quantities. This process is repeated for the rest of the MRR, linking $\{\epsilon_{i+1}, p_{i+1}\}$ to the known EOS by interpolation.

Here, a different and more simplistic approach to this problem is developed. This new method does not require the star to be integrated from crust to core, but will merely be a modification of the standard problem. This leads to a kind of "trial and error" type of solution, as section 3 will explain in more detail.

3 Reconstruction method

3.1 General goal

The goal is to reconstruct the EOS with as few biasing and imposing of mathematical form as possible. Therefore, the only pre-determined input is an EOS corresponding to a given mass in the MRR (Fig. 5). From there, the EOS can

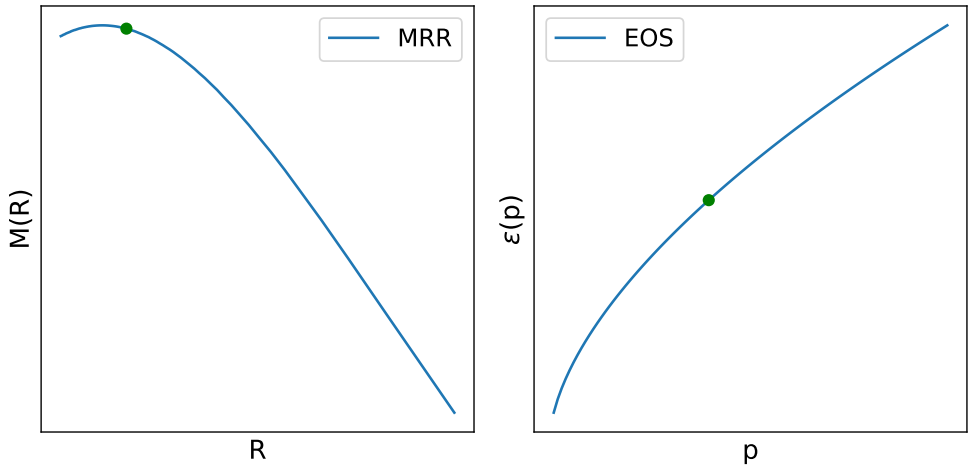


Figure 5: Corresponding points in MRR and EOS. Point in EOS resembles values in the center of the star.

be constructed without further constraints. Here, a straight line is added to the end of the EOS, that is, in the high density region (Fig. 6 b). The ODE solver starts at a point near the end of the given EOS and will calculate a mass with the (now small) piece of straight line and the EOS. If the mass does not fit the corresponding one in the MRR that was read in previously, the starting point on the line will be shifted to a higher pressure by a small step. At some very high pressure, a cut-off can be applied due to un-physical energy densities. If the starting point reaches the cut-off and the correct mass was not found, the slope of the line is varied. The latter also happens when the radius that corresponds to the calculated mass does not fit the one corresponding to the radius in the input data. In fig. 6 c) the slope and end point of the orange line have been successfully varied and the next mass and radius are reconstructed using the variation of the green line. The orange line is now a part of the given EOS. This process will be repeated until the end of the MRR file.

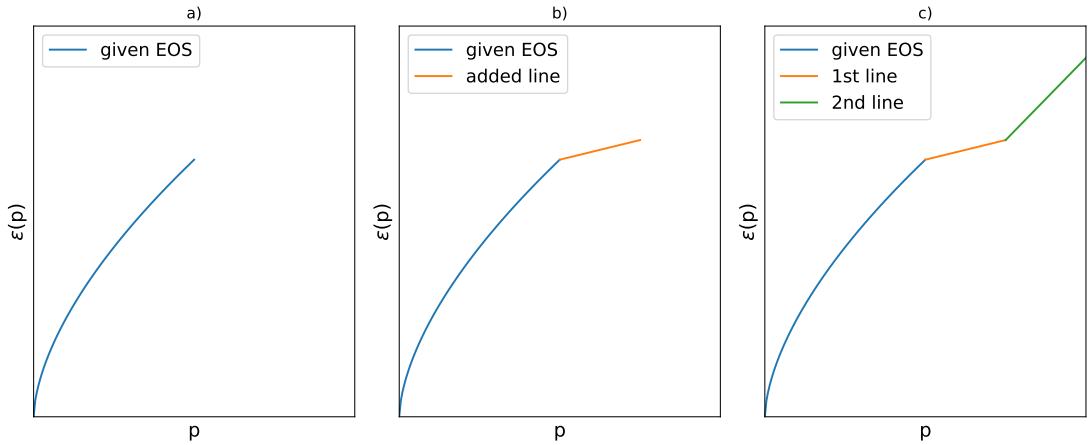


Figure 6: a) Given polytropic equation of state, b) added a constructed straight line, c) Successfully varied orange line, green line for next mass

3.2 Pressure intervals

For the first mass in the file that is supposed to be reconstructed, the code will use the given EOS over the complete pressure interval p_c to 0. That way, the first mass can be reproduced exactly in a test case. p_c will be the last pressure value of the given EOS interval. From there, lines will be added to that end point; in the program, a flag labelled "one" will be set to false after the first mass was calculated.

From this point on it is important to determine which EOS will be used for what pressure interval inside the star. The second mass to be calculated will use the given EOS and one single line after it; masses number three and above then also need a line, but before finishing with the given EOS the previously determined lines are supposed to fill in the rest of the pressure range.

The line currently subject to variation will start at the central pressure (p_c) and will end at the last pressure a star was reconstructed for. This is resembled by the end point of the last previous line (or in the special case, the last pressure of the given EOS). The last given EOS pressure is saved in the very beginning as p_{init}, marking the point from which the given EOS will finish up the mass calculation.

3.3 Linear interpolation

Four parameters are needed to calculate one line equation (two points in a two-dimensional plane). In the code, every EOS point consists of a pressure and an energy density value. The parameters needed for one line are stored in an array α:

$$(\alpha_0,\, \alpha_1,\, \alpha_2,\, \alpha_3) = (p_1,\, \varepsilon_1,\, p_2,\, \varepsilon_2)$$

Inserting this array into $y = mx + b$ or rather $\varepsilon = mp + b$, the line equation looks as follows:

$$\varepsilon(p) = (p - \alpha_1)\,\frac{\alpha_3 - \alpha_1}{\alpha_2 - \alpha_0} + \alpha_0 \tag{19}$$

where $m = \Delta\varepsilon/\Delta p$ and b is the y axis point. While one line is being varied, α_0 and α_1 stay fixed as they are the last line's end point. Calculating the mass, α_2 is varied at fixed α_3. If the corresponding radius does not fit, α_3 is shifted through a slope change. If mass and radius have been reconstructed with sufficient precision, the α array is extended by two in order to fit in the next end point to be varied. The EOS function always checks which interval the current pressure lies in to determine the α for further calculation. The array will be extended to index values high above three; in figure 6, indices zero to three were chosen to showcase the algorithm. In figure 6 c) $\alpha_{4,5}$ are being varied. In the actual program, (α_0, α_1) resembles the known EOS region's end. In order to generalize equation (19), an index i is introduced and initialized with alpha.size()-2. In the above special case, alpha.size() (function in the C library) would equal to four. The subscripts in (19) are now rewritten with i:

$$\varepsilon(p) = (p - \alpha_{i-1})\,\frac{\alpha_{i+1} - \alpha_{i-1}}{\alpha_i - \alpha_{i-2}} + \alpha_{i-2} \tag{20}$$

That way, the array can be more easily accessed an used by the code even with continuous extension.

Centering the whole reconstruction around the α array also immediately makes external EOS input possible. An EOS table read in by the code along with an MRR would be stored in α and used for the calculation with either the same linear function as the rest, or with another interpolation method. Note that the external table would need the energy density in one column (many EOS use the matter or baryon density instead).

3.4 Numerical solution

The TOV equations come in the form of a first-order ordinary differential equation (ODE) system. In order to obtain sufficient precision, a fourth-order Runge-Kutta algorithm (RK4) can be used to solve said ODE. By looking at both equations, our ODE system is given in the form:

$$\dot{y}(t) = f(y(t), t) \tag{21}$$

where $\dot{y}(t)$ is a two component "vector":

$$\dot{y}(t) = \begin{pmatrix} dP/dr \\ dm/dr \end{pmatrix} \tag{22}$$

$f(y(t), t)$ then contains the right hand side of equations (6) and (7).
Numerically, solving the ODE is achieved by approximation of a step τ in the function variable t (in TOV, this is the radius r) with derivatives of the function that is the ODE solution (y).

$$y(t + \tau) = y(t) + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{23}$$

with increments k_i:

$$\begin{aligned} k_1 &= f(y(t), t) \cdot \tau \\ k_2 &= f(y(t) + k_1/2, t + \tau/2) \cdot \tau \\ k_3 &= f(y(t) + k_2/2, t + \tau/2) \cdot \tau \\ k_4 &= f(y(t) + k_3, t + \tau) \cdot \tau \end{aligned}$$

All k_i are of course also two component "vectors".
For simpler test versions of the code, it is convenient to reduce the RK4 algorithm to RK1, or Euler's method. Also, the step size does not have to be adaptive. That way, runtime efficiency can be adjusted.

3.4.1 Adaptive step size

If mass and radius calculation algorithms do not yield a sufficient precision, the step size within the TOV solver can be implemented as an adaptive variable instead of a constant. Step size adaption works as follows: In order to maintain a minimum level of precision, the program checks for the change in pressure resulting from one Runge-Kutta step. If this change is too large, the step size τ is varied by a factor and the Runge-Kutta step count is reduced by one. Upon repeating the step, the same check takes place, and if necessary another τ is adjusted again. If the check is successful, the program will accept the step result and proceed with the next one.

3.4.2 Implementation

In the actual program (written in C), the left hand side of our ODE system is implemented as a two-dimensional array of form y[N][x]. N will be two, for the whole program, as we will not add other equations to the system. k_i are all one-dimensional arrays; that way x is only reflecting the step count. x will therefore be in range zero up to the number of iteration steps. We set an arbitrary maximum step amount to prevent segmentation faults. The RK4 algorithm will be implemented easily modifiable at any time to another order of Runge-Kutta or even Euler.

3.5 From TOV solver to reconstruction

After implementing the TOV equations together with the integration method, one can now generate an MRR by looping over several central pressures using one EOS only. This MRR will be used for testing the final reconstruction code, which will consist of the TOV solver extended with the reconstruction algorithm explained below.

The main additions will be the splitting of the pressure range into segments, each requiring their own part of the EOS, as well as a segment determining the "correctness" of calculated mass and radius. The code will work not as "inversely" as proposed before; the stars will still be integrated from the core towards the surface. However, the reconstruction no longer uses one single EOS but rather constructs its own.

3.6 Alternative method: Monte Carlo

In case the previously described algorithm is not capable of finding all correct radii and masses, an alternative shall be proposed here.
As before, up to a certain central pressure a known EOS will be applied. The next point will be defined as a "box" around a pressure some step higher than the previous one. Within that box, a Monte Carlo simulation generates energy densities and pressures, to be combined (every pressure with every energy density) to central values used for the calculation. The best point will then be the new central value and in further calculation a line will be drawn between that and the previous pressure. If this best point does not reach the wanted mass or radius, the Monte Carlo simulation will be repeated with more data points.
Alternatively, a Monte Carlo reconstruction would technically be capable of drawing a complete (somewhat arbitrary) EOS and varying it as a whole. Still, the curve would be discretized, for example into bins defined by certain pressure ranges. With that curve an MRR could be drawn with some number of stars. The drawn curve could then be compared to the input curve as a whole, constraining some global variation parameter by using a probability method (e.g. the Metropolis algorithm). The advantage of drawing the whole EOS is that the produced MRR can be of any precision (central pressure separation between stars). With the input MRR and an interpolation method of some kind for both MRR the two could be compared using for example a method of least squares, therefore discretizing both curves in the same manner. The sum of least squares would be a possible parameter for the Metropolis algorithm.
That way, the mass-radius points generated by the code would not have to be the same as those in the input data. Curve comparison should indeed take place using interpolation of points as an MRR does not follow a type of function that could be used fit it to the data.
This method is of course of highly numerical nature. However, in the past statistical approaches have often been of great success, which is why here a Monte Carlo method could potentially be used as a comparison.

3.7 Optimal radius and mass for a given slope

Before comparing the radius to the input, the optimum mass can easily be obtained by fixing a slope for the EOS and varying the central pressure in small steps. The latter has no significant impact on the radius. Here it proved to be convenient to use an indirect check instead of a direct comparison to the input mass. After the first calculation, an initial difference between input and outcome is stored. The difference for the next central pressure will be multiplied with the initial one; variation takes place until the multiplication outcome flips the sign. Then, the pressure step is decreased and variation also switches direction. This can be repeated until a desired precision is reached. The radius corresponding to the optimum mass will be given to the main function along with it.
The reason for not directly comparing radii is flexibility. A direct comparison ob the absolute values would make the problem numerically stiffer. The sign flip method works like a "ball" rolling down into a minimum, crossing it and rolling back from the other direction and at some point staying in the exact minimum.

3.8 Direct and indirect radius check

Upon finishing the calculation for a star, the code has to check mass and radius. This can either be done by directly comparing the calculated values with the input (in the following, direct method) or by choosing an indirect way of doing so. The direct method can be described as a search for the root of a function f_{dir}:

$$f_{\mathrm{dir}} = R_{\mathrm{calc}} - R_{\mathrm{want}}$$

An elegant way to unite radius and slope control is by choosing an indirect method function f_{ind}:

$$f_{\mathrm{ind}} = (R_{\mathrm{calc}} - R_{\mathrm{want}})^2 + \lambda \cdot (\xi_{\mathrm{curr}} - \xi_{\mathrm{prev}})^2 \tag{24}$$

with varying slope ξ_{curr} and fixed slope from before ξ_{prev}. Instead of trying to find the roots of f_{dir}, the code is now supposed to find extrema of f_{ind}. The functions are loosely related as

$$f_{\mathrm{ind}} \sim \int f_{\mathrm{dir}}$$

λ is an "importance parameter". In order to avoid big jumps in the slope ξ_{curr} with respect to ξ_{prev}, λ is set to some reasonable value. That way, ξ_{curr} will not deviate more from ξ_{prev} than λ allows. Before slope variation, a star is calculated once with the current slope and with a higher and lower one, respectively. The

slope variation direction is then determined by the behavior of f_{ind}. After the first reconstruction, λ is set to a value above zero while initially using $\lambda = 0$ as there is no slope from before to use.
λ can be estimated to have a starting value for a more extensive study os its impact on the calculation. In the minimum of f_{ind},

$$\lambda \sim \frac{(\Delta R)^2}{(\Delta \xi)^2} \tag{25}$$

roughly holds. ΔR here is the difference between two radii in the MRR and $\Delta \xi$ is the change of slopes between to linear interpolated pieces in an EOS table. Therefore λ is also dependent on how coarse the "grid" of the EOS table is. Often ΔR is of order 10^{-1} and not dependent of physical parameters (but just of the way the MRR was saved in the table), while $\Delta \xi$ is different for every EOS.
Some time in the future, the program should be capable of modeling first-order phase transitions in the EOS as it their existence is likely. Such a transition requires a jump in the slope and discontinuities in the EOS. It is obvious that the indirect method is biased towards smooth EOS without any unusual progression − ruling out the possibility to model a phase transition from the beginning. That can be dealt with by tracking extremal points in the MRR while reading it in as input. As discussed previously, a maximum in the MRR corresponds to a transition from stable to unstable when reading from larger to smaller radii. A new stable branch emerges when new effects take place to prevent further gravitational collapse.
When the reconstruction part of the code reaches the first mass behind a maximum, the algorithm is changed from f_{ind} to f_{dir} by setting $\lambda = 0$. ξ_{curr} may now freely deviate from ξ_{prev} to find the EOS points during and after the transition.

4 Results

4.1 Issues

During development of the code, various smaller and bigger issues had to be dealt with or worked around. A rather fundamental question is what numerical methods can or should be allowed in order to get nice and physically relevant results while still being able to motivate a change in numerical methods with physical reasons. For example, during some development stages cut-offs or statements were implemented that should only be taken into account by the code at a certain point through the MRR, which is obviously not particularly physical. The first attempt at global constraints in the calculation was the introduction of an indirect method; the λ importance parameter was the consequence of assuming the EOS to be (rather) smooth. Radius control issues will now be discussed in a more detailed fashion.

4.1.1 Radius and slope control

While the direct method (3.8) somewhat works in the low-density region, it quickly starts suffering from big slope jumps for the reconstructed lines. Some MRR points are found corresponding to values above the wanted EOS curve, continuing with only very slight changes in slope until points below the EOS are used for calculation. From there, a slope jump pushes the next point above the curve again. These jumps get bigger and pile up to a point where the code cannot push the current energy density above the EOS again; from here, none of the produced values are of use as now all further points strongly deviate from the EOS.
Initially, the indirect method (24) was chosen to ensure that the slope would not contain big jumps. This however has a huge impact on runtime, but as efficiency is not the main goal, optimization is postponed. The biggest problem is, ironically, the precision needed to avoid slope jumps. At some point, the code will calculate in slope steps so small that radius changes start to become exceedingly rare. As discussed by various researchers, the radius does not depend on a specific parameter except for maybe the order of magnitude of the pressure (see for example [16]) and often requires drastic changes in various values used in the calculation. Here, the slope is often the only parameter that has any impact in the first place; the mass is relatively easy to find along with a rather accurate central pressure. The indirect method includes the λ parameter in order to connect a radius check with

a condition regarding the slope with respect to the previous one. λ is to determine the maximum deviation of a reconstructed point from the curve, here is however no way of determining a "perfect" λ other than trial and error. The parameter remains rather arbitrary even though it can be motivated with a smoothness assumption. What that means for phase transitions has been discussed in 3.8: a transition has an immediate effect on the MRR, the code can therefore be modified based on input analysis. When the reconstruction reaches the point of transition, λ is used to modify the algorithm accordingly.
Setting $\lambda = 0$ of course bears the risk of "finding" wrong points easily as there is less control over how the calculation deals with the extra degree of freedom. If in the rest of the reconstruction small errors appear, the calculation will probably not find a decent radius at all. If now the constraints are loosened, precision in evaluating the dimensions and properties of the phase transitions are sacrificed.

4.2 Test case

The first test will take place with a simple polytropic EOS such as

$$\varepsilon(p) = \left(\frac{p}{10}\right)^{3/5}$$

Using the TOV solver, an MRR is generated. The MRR seen in fig. 7 will be the first one to be reconstructed. The curves both consist of 100 data points

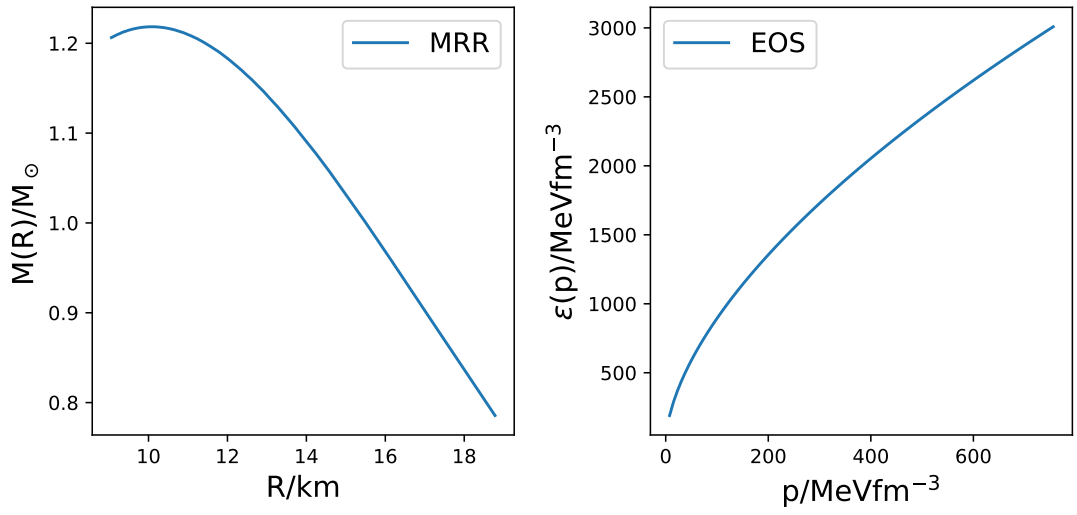


Figure 7: Test case MRR and EOS

which will be read in by the reconstruction code. The result can be seen in fig. 8: The radius was checked indirectly, also the mass has been calculated to an optimal value. Clearly, up to a density of about $100\,\mathrm{MeV\,fm^{-3}}$ the code works rather fine. In terms of computation time this was one of the longest runs as precision was achieved within the comparisons of mass and radius to the input while calculating the TOV equations without an adaptive step size nor with even a Runge-Kutta method (but Euler's instead). It is questionable if an RK4 algorithm would bring significant improvement to the obvious failure at higher densities; most likely it will mainly increase the computation time rather than the precision of the result. Considering how well the reconstruction algorithm worked in the low-density region, the sudden deviation of $3\,\mathrm{GeV\,fm^{-3}}$ in energy density paired with a $25\,\mathrm{MeV\,fm^{-3}}$ pressure jump is likely not a precision issue. The exact reason for the problem remains unclear as there do not seem to be fundamental issues regarding the algorithm as a whole.

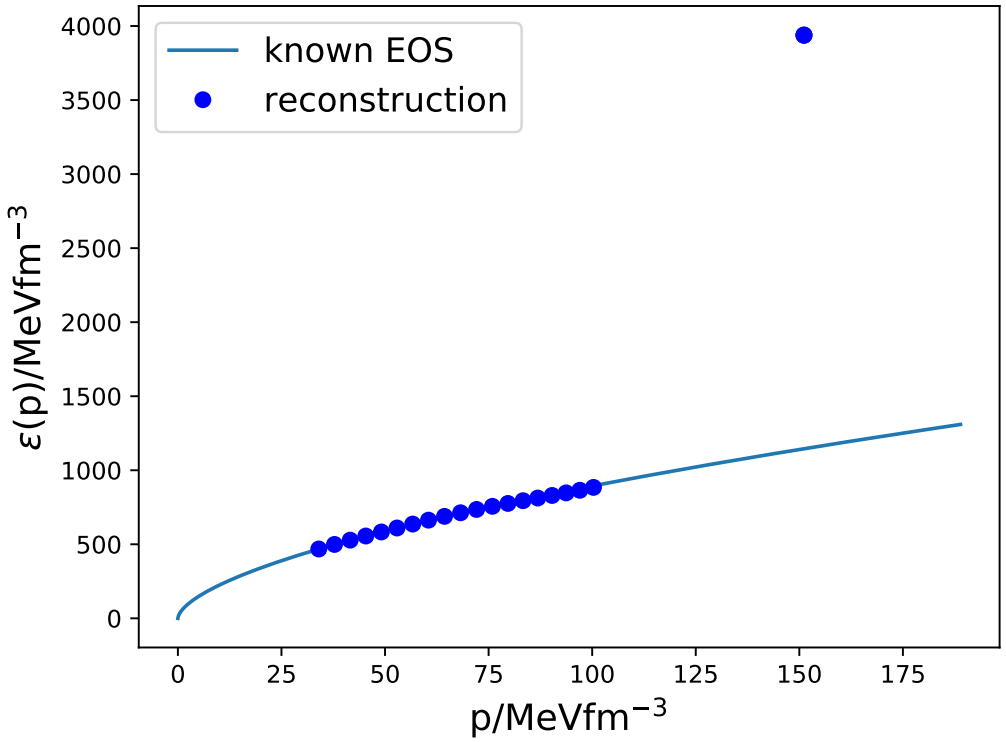


Figure 8: Reconstruction test with a code containing all discussed algorithm modifications and $\lambda = 10^{-2}$.

Figure 9 shows the result for a code without indirect radius check. At first glance, the program proceeded giving results above 100 MeV fm^{-3}. However, a closer look reveals sudden drops in slope to negative values, which is obviously unphysical. These drops occur after the code ran into a limit |slope| = 4 which was implemented to prevent large jumps (instead of an indirect radius comparison involving λ). Above about 110 MeV fm^{-3} the variable "slope" is still in an absolutely expected region, but calculating it from the pressures and energy densities stored in the α array yields negative values. This issue could not be resolved as it seems to appear for no apparent reason, especially since up to that point the program does what it should, giving masses and radii with surprising precision compared to the input.

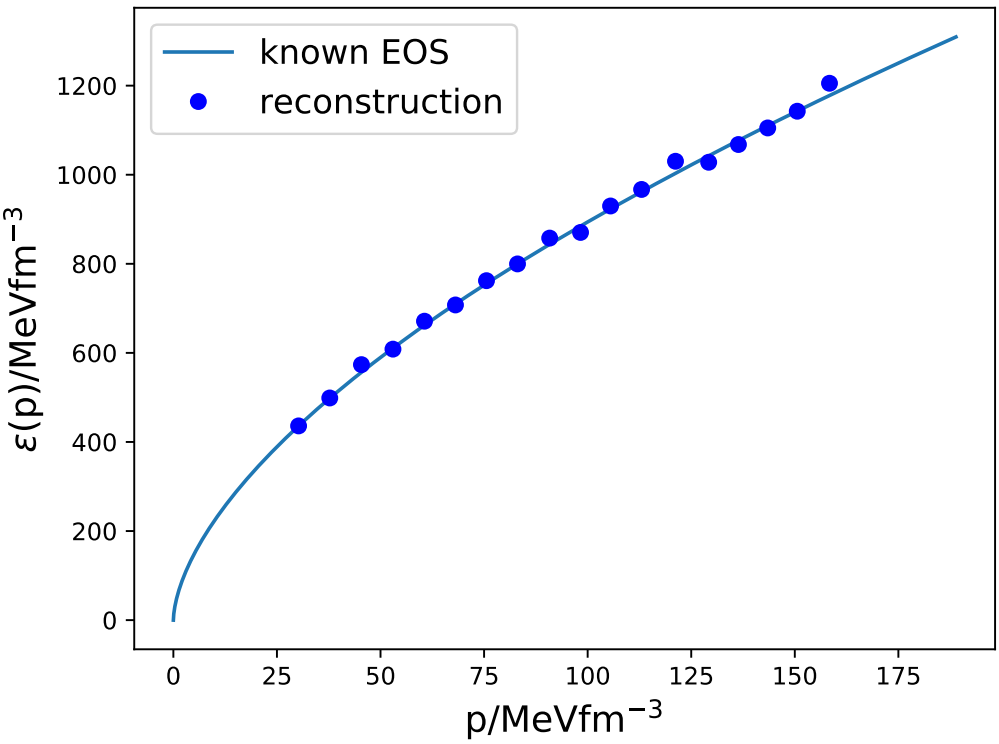


Figure 9: Reconstruction test, with another code version. At about 125 MeV fm^{-3} a negative slope occurs.

4.2.1 Varying the λ parameter

Figure 8 was made with $\lambda = 10^{-2}$. Varying λ has significant impact on precision. With no changes made otherwise, decreasing λ to, for example, 10^{-3}, leads to a strong deviation from the desired curve, whereas an increase in λ rapidly leads to an unreasonably long computation time. In order to find "good" values for λ, ten were chosen between $4.5 \cdot 10^{-2}$ and $8 \cdot 10^{-2}$. Below $\lambda = 5 \cdot 10^{-2}$ the EOS look like in fig. 8 and were not plotted again. Fig. 10 shows the other results.
By looking at fig. 10 a correlation between precision loss towards higher radii and sudden drops in the EOS can be seen immediately. This is technically plausible as lower energy densities causes larger radii (otherwise, the increased gravitational pull would lead to a smaller radii). It however remains unclear why there is no distinct range for λ to produce a "good" reconstruction; the "'failed" runs are distributed randomly over the range of used λ. This arbitrariness is concerning as a certain range of working λ would have made sense at least numerically if not physically. The results show a strong tendency to a trial-and-error based way of finding a well-working value which can in no way be motivated by any numerical or physical law, which might be the biggest issue the code currently has.
Looking at the sudden dropping EOS, correspondence to their MRR is questionable. For that reason, the EOS were re-entered into the TOV solver as a table paired with the polytropic low density region. That way correspondence could be confirmed.

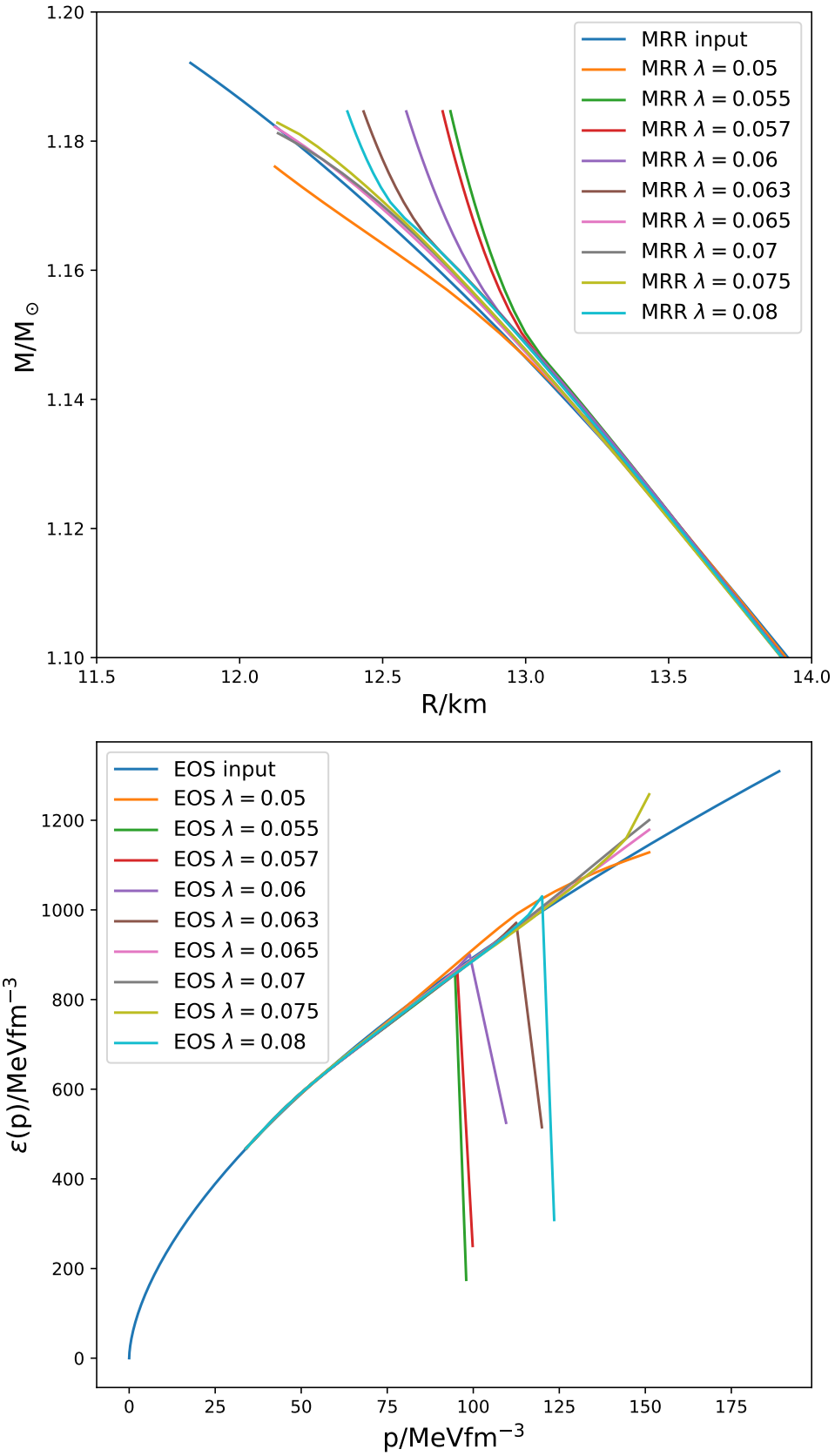


Figure 10: EOS and MRR for various values of λ. The precision was not changed.

4.2.2 MRR-EOS map uniqueness

As mentioned in 2.8, Lindblom makes use of the unique map linking EOS and MRR to construct his inverse algorithm. It seems peculiar that the radius, with given λ, only deviates from the input by a few meters, barely visible when shown in a plot. In contrast the EOS can deviate visibly from the desired input curve. However, as can be seen in fig. 11, for the EOS without obviously un-physical behavior (more precisely, the repeatedly occurring drops in energy density) the maximum deviation is about 10 %. To obtain fig. 11, energy densities were calculated with the original EOS and the reconstructed pressures. The reconstructed energy densities were then divided by these "default" ones. For comparison to unity the blue dashed line (division of original EOS by itself) was drawn together with the ratios.

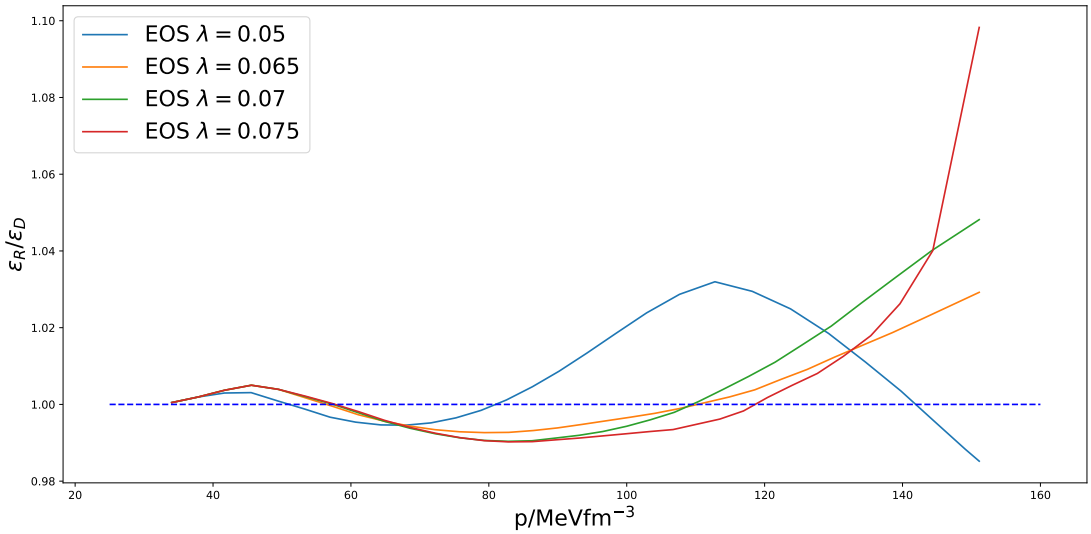


Figure 11: Ratio plots for the EOS that did not show peculiar behaviour. ε_R: reconstructed energy density, ε_D: default energy density.

4.3 Second equation of state

In order to have another EOS to test the program with, the following one was chosen due to the fact that energy density and pressure were given: Like before,

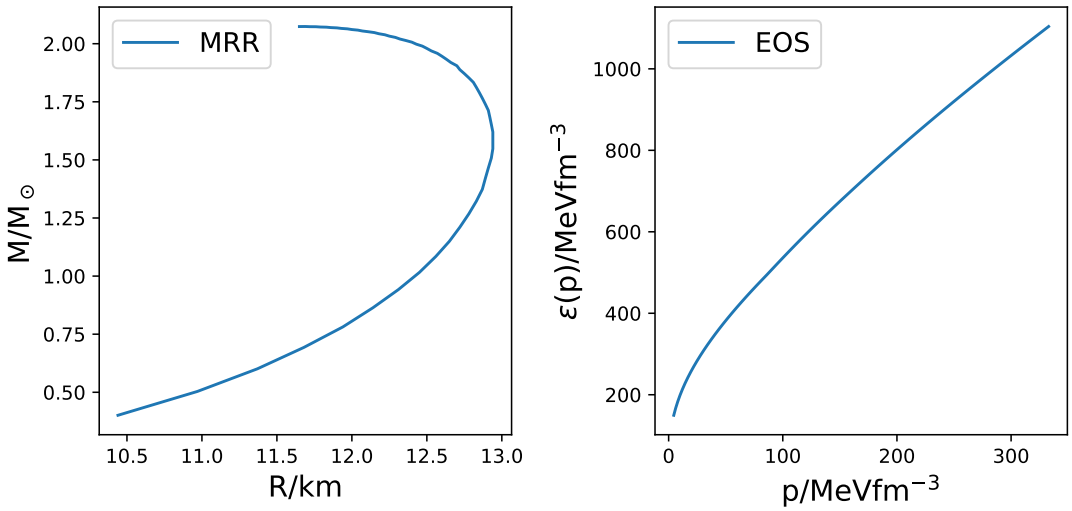


Figure 12: A second EOS together with the MRR it produces

various values for λ were tested out. Surprisingly, no sudden drops in energy density did occur; rather, all reconstructed curves follow the same "path". λ here does not seem to have a huge effect from $4 \cdot 10^{-2}$ and above. Again, both MRR and EOS have been plotted for all the λ (fig. 14). The similarity of the curves an also be seen clearly in fig. 13: in the beginning, a 5 % deviation occurs no matter what λ was set to, to then cross the unity line and deviate into the other direction to up to 15 %. The curves would undoubtedly deviate even further upon extension of the calculation to higher densities. Fig. 13 shows the problem of maintaining numerical precision: if the starting star for the reconstruction already deviates significantly from the input, the rest of the calculated masses and radii will likely never converge in terms of the corresponding central pressure and energy density values. Still, finding the correct masses was not an issue. The only massively deviating variable is the radius; sometimes, the deviation reaches almost 1 km. Varying the precision did not help either. The only part of the curve behavior that changes when changing required precision for the code to move on to the next star is the position of the intersect of reconstruction and input. Just as this point seems to move to higher densities with increasing λ, the precision can have

the same effect with slightly greater magnitude. Intersects may move to densities $200\,\mathrm{MeV\,fm^{-3}}$ higher.

Values of λ far out of the shown range were also tested, such as 10^3 or even extending the range to negative λ. For this specific EOS it did never make much of a difference in terms of precision: While the first mass and radius were found (almost) exactly, the following stars had the correct masses but radii about 0.5 km off from the input. With some λ, the masses started to deviate as well.

Computation time is a factor independent of the EOS. In order to achieve a complete curve in the first place, no matter its behavior, a statement regarding the slope variation step has to be implemented. For figures 10 and 14, the minimum step size for the slope was 10^{-6}. If the code goes below this minimum, the calculation stops and the next star is started. A test run without this statement for the external EOS managed to reconstruct the first star in both mass and radius to order 10^{-4}, but getting stuck on the second star with a slope step size of order 10^{-11}. This resulted in an unreasonably long computation time of at least multiple hours per star.

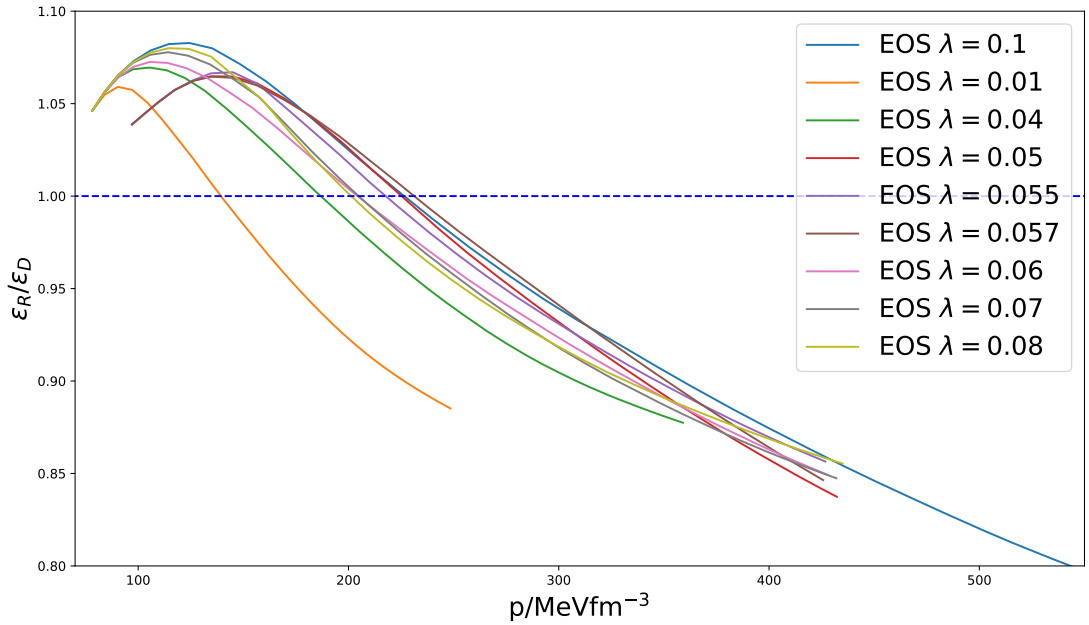


Figure 13: Ratio plots for the external EOS. ε_R: reconstructed energy density, ε_D: default energy density.

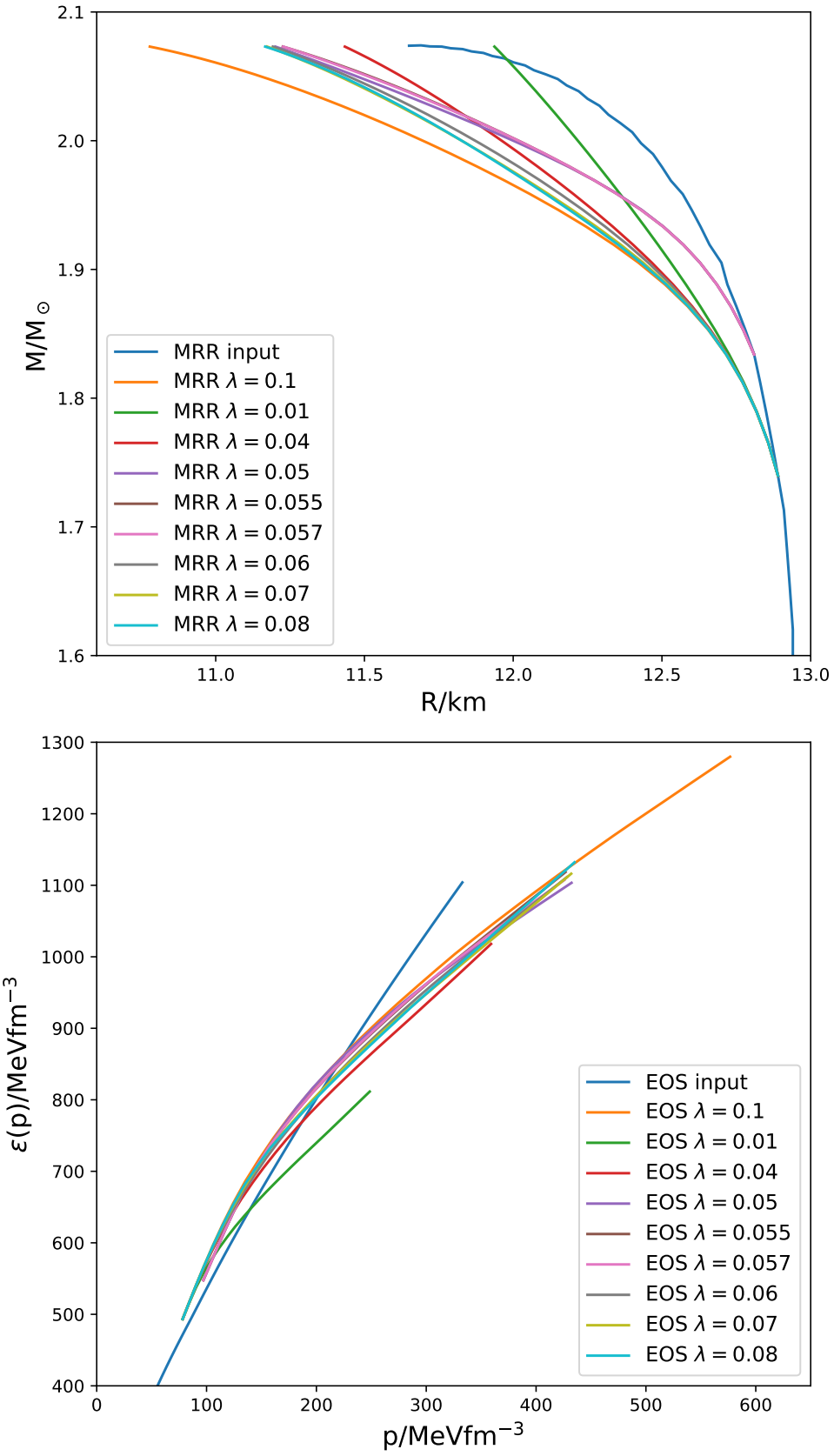


Figure 14: EOS and MRR for various values of λ. The precision was not changed.

5 Conclusion and outlook

Looking at the results in the previous section, the flaws and issues of the method used are obvious. The numerical algorithm produces un-physical behavior for no particular, addressable reason and is not capable of correcting deviations occurring in the beginning of the calculation. Therefore the code still needs rather extensive adjusting before use with another EOS.

Nonetheless, studying the effect of λ revealed the potential an inverse algorithm has for future analysis of actual data. Looking at the variety of EOS that were proposed and generated MRR with, it seems reasonable to choose an inverse algorithm to avoid being constrained to a single model. This does not mean that the approach in this thesis was the correct one, it is rather one of many that should be pursued further (mainly the Monte Carlo and Lindblom methods discussed earlier). With a well-working code, the high-density phase diagram of strongly interacting matter can likely be probed better than with any other approach. Earth bound experiments might never reach energies and densities needed to study matter in extreme states for a sufficient time, making astrophysics increasingly relevant as a laboratory.

Both EOS that were used to test the code had no connection to actual data from an experiment. When NICER obtains the first precise radius measurements, a main goal will be the maximum information extraction from these new constraints, whether through new theoretical models or numerical codes. The advantage lies clearly in the latter as it will immediately produce results that can be of further use in other fields of physics. The development or combination of theories will follow soon after in order to implement the numerically gathered information into already existing theoretical bodies.

What kind and how much information experiments like NICER will bring in the end remains of course unclear until the point of measurement. Until then, it is important to prepare for that moment as best as possible. The only reasonable preparation is obviously code development, as theoretical calculations will only make sense when they can be tested properly.

All approaches can, will, and should be continuously modified to maintain a variety of possible directions for future research, which is where this work contributes. If the above discussed method holds up against other types of inverse calculations can be examined in the future, once there is access to experimental data.

A References

[1] Paolo A. Mazzali, Friedrich K. Ropke, Stefano Benetti, and Wolfgang Hillebrandt. A Common Explosion Mechanism for Type Ia Supernovae. *Science*, 315:825, 2007.

[2] N. K. Glendenning, editor. *Compact stars : nuclear physics, particle physics, and general relativity*, 2000.

[3] V. R. Pandharipande, D. Pines, and R. A. Smith. Neutron star structure: theory, observation, and speculation. *ApJ*, 208:550–566, September 1976.

[4] S. L. Shapiro and S. A. Teukolsky. *Black holes, white dwarfs, and neutron stars: The physics of compact objects*. 1983.

[5] G. Baym, C. Pethick, and P. Sutherland. The ground state of matter at high densities: Equation of state and stellar models. *ApJ*, 170:299, December 1971.

[6] P. Haensel, A. Y. Potekhin, and D. G. Yakovlev, editors. *Neutron Stars 1 : Equation of State and Structure*, volume 326 of *Astrophysics and Space Science Library*, 2007.

[7] P. Haensel and B. Pichon. Experimental nuclear masses and the ground state of cold dense matter. *A&A*, 283:313–318, March 1994.

[8] F. Douchin and P. Haensel. A unified equation of state of dense matter and neutron star structure. *A&A*, 380:151–167, December 2001.

[9] D. G. Ravenhall, C. J. Pethick, and J. R. Wilson. Structure of matter below nuclear saturation density. *Physical Review Letters*, 50:2066–2069, June 1983.

[10] N. K. Glendenning, editor. *Compact stars. Nuclear physics, particle physics, and general relativity*, 1997.

[11] K. Schertler, C. Greiner, J. Schaffner-Bielich, and M. H. Thoma2. Quark phases in neutron stars and a third family of compact stars as signature for phase transitions[1]. *Nuclear Physics A*, 677:463–490, September 2000.

[12] N. K. Glendenning and C. Kettner. Possible third family of compact stars more dense than neutron stars. *A&A*, 353:L9–L12, January 2000.

[13] U. H. Gerlach. Equation of state at supranuclear densities and the existence of a third family of superdense stars. *Physical Review*, 172:1325–1330, August 1968.

[14] Mark G. Alford and Armen Sedrakian. Compact stars with sequential QCD phase transitions. *Phys. Rev. Lett.*, 119(16):161104, 2017.

[15] L. Lindblom. Determining the nuclear equation of state from neutron-star masses and radii. *ApJ*, 398:569–573, October 1992.

[16] J. M. Lattimer and M. Prakash. Neutron Star Structure and the Equation of State. *ApJ*, 550:426–442, March 2001.

B List of abbreviations

EOS equation of state

MRR mass-radius relation

NICER Neutron Star Interior Composition Explorer

ODE ordinary differential equation

RK4 fourth-order Runge-Kutta algorithm

TOV Tolman-Oppenheimer-Volkoff equations

C Statutory Declaration

I herewith declare that I have composed the present thesis myself and without use of any other than the cited sources and aids. Sentences or parts of sentences quoted literally are marked as such; other references with regard to the statement and scope are indicated by full details of the publications concerned. The thesis in the same or similar form has not been submitted to any examination body and has not been published. This thesis was not yet, even in part, used in another examination or as a course performance.

Place, Date Jan Röder

D Eidesstattliche Erklärung

Ich versichere hiermit, dass ich die vorliegende Arbeit selbständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel verfasst habe. Wörtlich übernommene Sätze oder Satzteile sind als Zitat belegt, andere Anlehnungen, hinsichtlich Aussage und Umfang, unter Quellenangabe kenntlich gemacht. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen und ist nicht veröffentlicht. Sie wurde nicht, auch nicht auszugsweise, für eine andere Prüfungs- oder Studienleistung verwendet.

Ort, Datum Jan Röder