MAM4001W: Applied Mathematics Honours

Relativistic Neutron Stars Beyond General Relativity

Thato Thapo

Abstract

In order to describe static spherically symmetric neutron stars in general relativity, we call upon the TOV equations, which we solve here for $f(R)$ gravity. We use the standard metric formalism for the particular $f(R)$ theory of gravity used in this paper whereby the affine connection Γ^a_{bc} depends on the metric tensor g_{ab}. We solve the TOV equations by first deriving them in the General Relativity formulation and implementing various numerical schemes to integrate the resultant differential equations extracted from the Einstein field equations. Then using the TOV equations found from the field equations in the $f(R)$ gravity formulation, we find relations with the mass and radius of a neutron star and use those to verify that $f(R)$ gravity with $f(R) = R + \alpha R^2$ as a theory of modified gravity consistent with general relativity for certain choices of the parameter α.

Supervised by Dr Bishop Mongwane
University of Cape Town
October, 2022

Contents

1 Introduction

As far as theories of gravity go, there is only one which is widely accepted by most physicists, and that is General Relativity. Much of the universe can be explained using this theoretical framework however, it too displays a few shortcomings. There are a number of problems that still remain unsolved by this theory, such as the Dark Matter Problem in the context of Astronomy, and the Dark Energy Problem as well as Early Inflation in the context of Cosmology [10]. These problems, along with many others are what prompt the search for extended theories of gravity. Additionally, questioning General Relativity allows us to gain a much deeper understanding of gravitational interactions and the theory itself. Because General Relativity works really well already, these extended theories need only do that, extend GR, and so they need to be able to recover what we call the "classical tests" on top of explaining the open problems. The most common of these classical tests are, the precession of planetary orbits, gravitational lensing & the existence of gravitational waves. One of the most successful and simplest ways of extending GR, which will be explored here, is using what we call $f(R)$ theories of gravity.

1.1 Metric f(R) Gravity extending General Relativity

One can take the radical perspective that gravity is the governing force in the evolution of the universe and thus motivate a need to generalize GR such that our theory of gravity addresses the previously mentioned open problems. There are a multitude of ways to extend GR, and doing so is no simple task however, as mentioned before, we take a more simpler path in studying metric $f(R)$ gravity.

Before going into $f(R)$ gravity, a quick overview of general relativity is warranted. The main results derived from general relativity are the Einstein Field Equations, and these are retrieved from varying and extremizing the Einstein-Hilbert action,

$$S_{EH} = \frac{1}{16\pi G}\int d^4x\sqrt{-g}R \tag{1}$$

where G is the gravitational constant which we take here to equal 1 as we are using Geometrized units[1], g is the determinant of the metric tensor g_{ab} and R is the Ricci scalar calculated from the Ricci tensor R_{ab} which itself is derived from the Riemann curvature tensor R^a_{bcd} which essentially encodes the curvature properties of the spacetime we are working on. A full derivation can be found in any General Relativity textbook however, the Einstein Field Equations are ultimately found to be,

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

where g_{ab} is our spacetime metric and T_{ab} is the energy momentum tensor whose job is describing the matter content in the universe. With matter being a player in this game of gravity, one has to question what role do conservation laws play. In the case of GR, the Bianchi Identities govern the conservation of energy-momentum and are given by,

$$\nabla_a T^{ab} = 0, \tag{3}$$

where ∇_a is called the covariant derivative defined by,

$$\nabla_b v^a \equiv \partial_b v^a + \Gamma^a{}_{cb}v^c, \tag{4}$$

for an arbitrary covariant component of a vector v^a [see Hobson[7], for a detailed derivation], where $\Gamma^a{}_{cb}$ is the connection coefficient [see Sec. 2.2].

[1] $G = c = \hbar = 1$. We use Geometrized units to avoid dealing with numbers in extreme orders of magnitude

Following similar procedures, one would be able to derive the field equations for $f(R)$ gravity, where in order to proceed with that, the action would need to be generalized for $f(R)$ gravity. In the instance of modified gravity, we deal with higher order curvature invariants which in this case refers only to the Ricci scalar. The function $f(R)$ being dependent on the Ricci Scalar with higher orders of magnitude, would need to play a role in the action and so the generalized action for $f(R)$ theories of gravity will be,

$$S = \frac{1}{16\pi G} \int d^4x \sqrt{-g} f(R). \tag{5}$$

From the action above the field equations can then be derived and will be given by,

$$f' R_{ab} - \frac{1}{2} f g_{ab} - \nabla_a \nabla_b f' + g_{ab} \Box f' = 8\pi T_{ab}, \tag{6}$$

as shown in [8], where the prime represents derivatives of $f(R)$ with respect to the Ricci scalar and $\Box$ is the d'Alembertian operator defined as $\Box \equiv g^{ab} \nabla_a \nabla_b$. These field equations for $f(R)$ gravity can be seen to be very distinct from the Einstein field equations in general relativity (Eq. 2) however in both cases we see that the matter content is still accounted for on the right hand side and in both instances, the conservation of energy-momentum (i.e. Eq. 3) still holds.

1.2 Static Spherically Symmetric Neutron Stars

In order to get a physical grasp of our situation, we consider the simplest physical system we can describe using this model of gravity. Given a general line element,

$$ds^2 = g_{ab} dx^a dx^b, \tag{7}$$

a static system is one of the simplest systems we can construct. A static system is one where all the components of our metric will be independent of time, that is independent of the "timelike" coordinate x^0, and the geometric nature of the system will be unchanged by time reversal which means the line element of concern ds^2 will be invariant under under the transformation $x^0 \to -x^0$[7] . Another constraint we are privy to impose on our system is that it be spherically symmetric, which will be incredibly useful because spherically symmetric systems are also reasonably simple to work with but also very physical; in fact much of the objects described by astrophysics appear to be nearly spherical. An example of such objects, which will be of great importance in this paper, are neutron stars.

A neutron star forms after the event of a supernova with a collapse of the core of an exploded massive star into a smaller object whose radius is approximately $8km$. This would make neutron stars denser than any other known types of stars [2]. Because of this, we are interested in exploring how this high density can allow us to describe these stars both in GR and in $f(R)$ gravity. This will warrant the use of an appropriate equation of state (EoS), so that we can establish the relation between pressure and density $p(\rho)$, and that will be useful in quantifying the matter content in our field equations governed by the energy momentum tensor T_{ab}. In order to describe neutron stars using these theories of gravity, it proves useful to learn about their origins and the observations made to in fact, confirm that they are existing astrophysical structures.

1.2.1 The History and Discovery of Neutron Stars

Neutron stars were first proposed in 1934 by Baade and Zwicky as high density astrophysical objects with a very small radius formed in supernova explosions and more gravitationally bound than normal stars [1].

Oppenheimer and Volkoff in 1939 were the first to make significant calculations in this regard, assuming matter to be made up of an "ideal gas of free neutrons at high density" [9]. The main research on neutron stars around this period focused on the neutron cores in ordinary stars however due to the advancements in thermonuclear fusion, neutron stars would be significantly abandoned for the next 30 years bar a few papers throughout the years, reasons attributed to the potential inability to observe their residual thermal radiation due to their small surface area. The discovery of cosmic, non-solar X-ray sources [6] generated a great flurry of interest in neutron stars. Further interest in neutron stars was generated by the discovery of "quasi-stellar objects"/"quasars" by Schmidt at Mt. Palomar in 1963.

Despite the re-ignition of interest and increasing efforts from theorists, neutron stars and black holes were still not taken very seriously by most physicists and astronomers. Which all changed in 1967 with the discovery of pulsars which were proposed by Gold (1968) to be rotating neutron stars.

The near simultaneous discovery of the Crab and Vela pulsars in 1968, both situated in supernova remnants, provided evidence for the formation of neutron stars in supernova explosions. e.g. The Crab nebula is the remnant of the supernova explosion observed by Chinese astronomers in 1054 AD. All this led to great scientific feats which resulted in over 350 pulsars known and more stellar observations made [12].

1.2.2 Neutron Star Masses

Most observations confirm that neutron stars with masses near the Chandrasekhar limit of $1.4M_{\odot}$ are the most favoured in nature[12], with the minimum mass of a stable neutron star said to be $M_{\min} = 0.0925M_{\odot}$. An important feature though is the ability to identify compact objects as black holes by merely comparing the masses with the maximum allowed mass for a stable neutron star. Shapiro [12], goes through the methodology of determining this upper limit of the mass of a stable neutron star without dependence on the equation of state. The largest reported mass for a neutron star is $M = 2.01 \pm 0.4M_{\odot}$, and in the context of this paper, we shall use that value to verify the consistency of the theories presented here.

1.3 The Polytropic Equation of State

The equation of state is an expression whose purpose is relating quantities present in the energy-momentum tensor T_{ab}, usually the pressure p, to other hydrodynamic variables like the rest-mass density ρ, i.e., $p = p(\rho)$. This expression is particularly important because it represents the realism in how a given physical system is described. The natural conclusion here is that for many different physical systems, there would be many different equations of state, they can however be classed into categories depending on the properties certain systems have in common, Rezzola [11] goes through different classes of equations of state [in Sec. 2.4].

The equation of state we will be focusing on in this paper will be the polytropic equation of state expressed as

$$p = K\rho^{\Gamma} = K\rho^{1+1/N_p}, \tag{8}$$

where K is called the *polytropic constant* and $N_p = 1/(\Gamma - 1)$ is the *polytropic index* with Γ called the *polytropic exponent*, both K and Γ are constants.

The polytropic equation of state is the most commonly used equation of state for self-gravitating spherically symmetric gaseous spheres and is widely used in numerical calculations, and so it will be very relevant in our study of neutron stars.

Since this equation of state strictly correlates with the structure of the neutron star [4], we will be used in the efforts to solve the TOV equations which can be derived from the field equations and Bianchi identities as will be shown in Sec. 2 for the GR case. These equations were first calculated by Tolman to analyse spherically symmetric models and later extended by Oppenheimer and Volkoff to form what we now know as the Tolman-Oppenheimer-Volkoff equations. And so we use these equations along with the polytropic equation of state in order to obtain certain relations that will lead us to a maximal mass for the neutron star allowing for the comparison between GR and $f(R)$ gravity.

1.3.1 Sound Speed

Using the equation of state alone, we can extract other quantities related in any way to the pressure or the density. For example, by taking the direct derivative of the polytropic equation of state $dp/d\rho$, one can derive the equation of the sound speed Cs as done in [11], this will yield,

$$Cs^2 = \frac{\Gamma p}{\rho h} = \frac{\Gamma(\Gamma-1)p}{\rho(\Gamma-1)+\Gamma p}. \tag{9}$$

We are dealing with static spherically symmetric neutron stars as an example of a selfgravitating fluid configuration, so we can use the sound speed as a measure of the stiffness and compressibility of this fluid. The stiffness is relayed via Γ where, the larger Γ is, the more stiff the equation of state, and a smaller Γ refers to a soft equation of state. A stiff EoS means if there small changes to the density, the pressure will experience large changes, and a soft equation of state means smaller changes to the density result in smaller changes to the pressure[11].

The sound speed will come in handy once we have numerically solved the TOV equations. The polytropic equation of state will be of greater use as well in Sec. 4 with describing the state of our chosen system, especially when dealing with the pressure.

2 Theory

There is a wide range of $f(R)$ theories of gravity that we can choose from, implying that before we derive the TOV equations, we need to specify the class of f(R) theories that we will be working with.

2.1 The Starobinsky R-squared model

In this project, we will focus on a particular class of $f(R)$ models, introduced in 1980, known as the Starobinsky models, given by,

$$f(R) = R + \alpha R^n, \tag{10}$$

specifically the R-squared model given when ($n = 2$), where α is free parameter discussed below. From the field equations (Eq. 6), it is clear that one will be working with derivatives of the $f(R)$ which for this case will be given by,

$$f'(R) = 1 + 2\alpha R \tag{11}$$
$$f''(R) = 2\alpha \tag{12}$$
$$f'''(R) = 0. \tag{13}$$

Taking $\alpha = 0$ recovers the Einstein field equations. It can shown that α can also be constrained from observations by either using binary pulsar data giving $|\alpha| \lesssim 5 \times 10^{15} cm^2$ constraining α in the weak field limit; and $|\alpha| \lesssim 10^{10} cm^2$ constraining α in the strong gravity regime [5]. Here we will not be imposing said constraints, and we will instead choose a value of α consistent with general relativity, and vary it to see how the inconsistency arises.

2.2 The TOV equations

In order to extract the TOV equations from the Einstein Field Equations 2, we begin with general relativity, starting with the metric of a static, spherically symmetric spacetime [8]:

$$ds^2 = -e^{2\nu}dt^2 + e^{2\lambda}dr^2 + r^2 d\theta^2 + r^2 \sin^2\theta d\phi^2, \tag{14}$$

which we can use to get the particular geometric quantities. We will first need the Christoffel connections which are given by,

$$\Gamma^a{}_{bc} = \frac{1}{2} g^{ad}(g_{db,c} + g_{dc,b} - g_{bc,d}), \tag{15}$$

where our metric tensor g_{ab} is given by and its inverse g^{ab} are given by,

$$g_{ab} = \text{diag}(-e^{2\nu}, e^{2\lambda}, r^2, r^2 \sin^2\theta) \tag{16}$$

$$g^{ab} = \text{diag}\left(-e^{-2\nu}, e^{-2\lambda}, \frac{1}{r^2}, \frac{1}{r^2 \sin^2\theta} \right), \tag{17}$$

and we denote the partial derivative using a comma as in $\partial_c g_{db} = g_{db,c}$.

If this was a general case, we would have ν and λ be functions of (t, r, θ, ϕ), but since we assumed spherical symmetry in Sec. 1, there is no dependence on θ and ϕ. Also, since this is a static configuration, meaning there is no mass loss or gain, and there is no rotation, there is no time dependence. Therefore we will have $\nu \to \nu(r)$ and $\lambda \to \lambda(r)$ depend only on r.

We can now use this information to calculate the non-zero components of the Christoffel symbols staring with,

$$\begin{aligned}
\Gamma^t{}_{tr} &= \frac{1}{2} g^{tt}(g_{tt,r} + g_{tr,t} - g_{tr,t}) \\
&= \frac{1}{2} g^{tt}(g_{tt,r}) \\
&= \frac{1}{2}(-e^{-2\nu}) \frac{d}{dr}(-e^{2\nu}) \\
&= \frac{1}{2}(-e^{-2\nu})(2e^{2\nu}\nu') = \nu'(r) = \Gamma^t{}_{rt}.
\end{aligned}$$

The rest of the non-zero components are calculated in Appendix A, but in summary, we list them below,

$$\begin{aligned}
&\Gamma^t{}_{tr} = \Gamma^t{}_{rt} = \nu'(r); \quad \Gamma^r{}_{tt} = \nu'(r) e^{-2\lambda + 2\nu}; \quad \Gamma^r{}_{rr} = \lambda'(r) \\
&\Gamma^r{}_{\theta\theta} = r e^{-2\lambda}; \quad \Gamma^r{}_{\phi\phi} = -r \sin^2\theta e^{-2\lambda}; \quad \Gamma^\theta{}_{r\theta} = \Gamma^\theta{}_{\theta r} = \frac{1}{r} \\
&\Gamma^\theta{}_{\phi\phi} = -\sin\theta\cos\theta; \quad \Gamma^\phi{}_{r\phi} = \Gamma^\phi{}_{\phi r} = \frac{1}{r}; \quad \Gamma^\phi{}_{\theta\phi} = \Gamma^\phi{}_{\phi\theta} = \cot\theta.
\end{aligned}$$

Now that we have all the non-zero components for the Christoffel symbols, we can calculate the Ricci tensor components using the formula to calculate the Riemann curvature tensor because the Ricci tensor is given by,

$$R_{\beta\delta} = R^\alpha{}_{\beta\alpha\delta} = R^t{}_{\beta t\delta} + R^r{}_{\beta r\delta} + R^\theta{}_{\beta\theta\delta} + R^\phi{}_{\beta\phi\delta}. \tag{18}$$

Therefore, using the Christoffel symbols we get,

$$\begin{aligned}
R^t{}_{\beta t\delta} &= \Gamma^t{}_{\beta\delta,t} - \Gamma^t{}_{\beta t,\delta} + \Gamma^t{}_{\mu t}\Gamma^\mu{}_{\beta\delta} - \Gamma^t{}_{\mu\delta}\Gamma^\mu{}_{\beta t} \\
R^r{}_{\beta r\delta} &= \Gamma^r{}_{\beta\delta,r} - \Gamma^r{}_{\beta r,\delta} + \Gamma^r{}_{\mu r}\Gamma^\mu{}_{\beta\delta} - \Gamma^r{}_{\mu\delta}\Gamma^\mu{}_{\beta r} \\
R^\theta{}_{\beta\theta\delta} &= \Gamma^\theta{}_{\beta\delta,\theta} - \Gamma^\theta{}_{\beta\theta,\delta} + \Gamma^\theta{}_{\mu\theta}\Gamma^\mu{}_{\beta\delta} - \Gamma^\theta{}_{\mu\delta}\Gamma^\mu{}_{\beta\theta} \\
R^\phi{}_{\beta\phi\delta} &= \Gamma^\phi{}_{\beta\delta,\phi} - \Gamma^\phi{}_{\beta\phi,\delta} + \Gamma^\phi{}_{\mu\phi}\Gamma^\mu{}_{\beta\delta} - \Gamma^\phi{}_{\mu\delta}\Gamma^\mu{}_{\beta\phi}
\end{aligned}$$

$$
\begin{aligned}
R_{tt} &= R^{t}{}_{ttt} + R^{r}{}_{trt} + R^{\theta}{}_{t\theta t} + R^{\phi}{}_{t\phi t} \\
&= \Gamma^{r}{}_{tt,r} + \Gamma^{r}{}_{rr}\Gamma^{r}{}_{tt} - \Gamma^{r}{}_{tt}\Gamma^{t}{}_{tr} + \Gamma^{\theta}{}_{r\theta}\Gamma^{r}{}_{tt} + \Gamma^{\phi}{}_{r\phi}\Gamma^{r}{}_{tt} \\
&= \nu'' e^{-2\lambda+2\nu} + \nu' e^{-2\lambda+2\nu}(-2\lambda' + 2\nu') + \lambda'\nu' e^{-2\lambda+2\nu} - \\
&\qquad \nu'^2 e^{-2\lambda+2\nu} + \frac{1}{r}\nu' e^{-2\lambda+2\nu} + \frac{1}{r}\nu' e^{-2\lambda+2\nu} \\
&= e^{-2\lambda+2\nu}\left[\nu'' - 2\lambda'\nu' + 2\nu'^2 + \lambda'\nu' - \nu'^2 + \frac{2\nu'}{r}\right] \\
&= e^{-2\lambda+2\nu}\left[\nu'' - \lambda'\nu' + \nu'^2 + \frac{2\nu'}{r}\right]
\end{aligned}
$$

Similarly, the other components of the Ricci tensor will be given by,

$$
\begin{aligned}
R_{rr} &= \frac{2\lambda'}{r} + \lambda'\nu' - \nu'^2 - \nu'' \\
R_{\theta\theta} &= e^{-2\lambda}[-1 + e^{2\lambda} + r\lambda' - r\nu'] \\
R_{\phi\phi} &= e^{-2\lambda}\sin^2\theta[-1 + e^{2\lambda} + r\lambda' - r\nu'].
\end{aligned}
$$

The Ricci Scalar can then be calculated from the components of the Ricci tensor as well as the inverse metric tensor g^{ab} and will be given by,

$$
\begin{aligned}
R &= R_{tt}g^{tt} + R_{rr}g^{rr} + R_{\theta\theta}g^{\theta\theta} + R_{\phi\phi}g^{\phi\phi} \\
&= 2e^{-2\lambda}\left[-\frac{1}{r^2} + \frac{e^{2\lambda}}{r^2} - \frac{2\nu'}{r} - \nu'^2 + \frac{2\lambda'}{r} + \lambda'\nu' - \nu''\right].
\end{aligned}
$$

Using the field equations and the Bianchi Identities, we can use the geometric quantities we have calculated to derive the TOV equations for General Relativity.

We start with the Einstein field equations 2 and realise they contain the energy momentum tensor, which for a perfect fluid is given by,

$$
T^{\mu\nu} = (e + p)u^{\mu}u^{\nu} + pg^{\mu\nu}, \tag{19}
$$

where e is the energy density, p is the pressure and u^{μ} the 4-velocity. The only non-zero component of the 4-velocity will be u^{t} and this is due to the static configuration we have chosen [8]. If we then use the normalization condition $u_{\mu}u^{\mu} = -1$ we get,

$$
u^{t} = e^{-\nu}, \qquad u_{t} = -e^{\nu} \tag{20}
$$

And so using that we can calculate the non-zero components of $T^{\mu\nu}$ as follows,

$$
\begin{aligned}
T^{tt} &= (e + p)u^{t}u^{t} + pg^{tt} \\
&= (e + p)e^{-2\nu} + p(-e^{-2\nu}) \\
&= e \cdot e^{-2\nu}.
\end{aligned}
$$

And similarly,

$$\begin{aligned}
T^{rr} &= pg^{rr} \\
&= pe^{-2\lambda} \\
T^{\theta\theta} &= pg^{\theta\theta} \\
&= \frac{p}{r^2} \\
T^{\phi\phi} &= \frac{p}{r^2 \sin^2\theta}.
\end{aligned}$$

Combining all the information we have, we can start with the time components of the Einstein field equation,

$$R_{tt} - \frac{1}{2} g_{tt} R = 8\pi T_{tt}, \tag{21}$$

So therefore, plugging in the time component of the Ricci tensor and metric tensor, as well as the Ricci scalar into equation 21 we are able to work out a differential equation involving the mass, which reads as,

$$\frac{dm}{dr} = 4\pi e r^2. \tag{22}$$

Looking now at the radial components of the field equations,

$$R_{rr} - \frac{1}{2} g_{rr} R = 8\pi T_{rr}, \tag{23}$$

we use the fact that $T_{rr} = 0$ and we can then similarly work out a differential equation for ν which is as follows,

$$\frac{d\nu}{dr} = \frac{m + 4\pi r^3 p}{r(r-2m)}. \tag{24}$$

We can then use the Bianchi Identities to extract a differential equation for the pressure because they contain the energy-momentum tensor, from Eq. 3 we have,

$$\nabla_c T^{cb} = \partial_c T^{cb} + \Gamma^c{}_{dc} T^{db} + \Gamma^b{}_{dc} T^{cd} = 0. \tag{25}$$

We can then use the non-zero components of T^{ab} that we calculated from Eq. 19. Taking the radial component T^{rr} we get,

$$\begin{aligned}
\nabla_r T^{rr} &= \partial_r T^{rr} + \Gamma^c{}_{rc} T^{rr} + \Gamma^r{}_{dc} T^{cd} = 0 \\
\therefore \partial_r T^{rr} &+ \Gamma^t{}_{rt} T^{rr} + \Gamma^r{}_{rr} T^{rr} + \Gamma^\theta{}_{r\theta} T^{rr} \Gamma^\phi{}_{r\phi} T^{rr} \\
&+ \Gamma^r{}_{rr} T^{rr} + \Gamma^r{}_{\theta\theta} T^{\theta\theta} \Gamma^r{}_{\phi\phi} T^{\phi\phi} + \Gamma^r{}_{tt} T^{tt} = 0,
\end{aligned}$$

with the above resulting from the indices c and d that are being summed over. We can then plug in the relevant components of the energy-momentum tensor and the Christoffel symbols as they have been already calculated, which will result in a differential equation for the pressure given by,

$$\frac{dp}{dr} = -(e+p)\frac{d\nu}{dr}. \tag{26}$$

The equations 22, 24 and 26 are what we call the TOV equations, and solving these is what will give us the various properties of the neutron star, namely the relation between the star's radius with its mass, pressure, sound speed, and more. A similar approach is taken when deriving the TOV equations for the metric $f(R)$ gravity, the derivation will not be undertaken here but can be found in [4] and [10], where using the field equations 6 and the metric 14, we can find the values,

$$\zeta = \zeta(R) \tag{27}$$

$$= \left\{ f' + \frac{r}{2} f'' Q \right\}^{-1} \tag{28}$$

$$\lambda = -\frac{1}{2} \ln \left(1 - \frac{2m}{r} \right). \tag{29}$$

Taking the derivative of λ will then allow us to get an equation for dm/dr as follows:

$$\begin{aligned} \frac{d\lambda}{dr} &= -\frac{1}{2} \left(\frac{r}{r-2m} \right) \left(\frac{m'r - 2m}{r^2} \right) \\ &= -\frac{1}{2} \left(\frac{m''r - 2m}{r(r-2m)} \right) \\ \Rightarrow m'r - 2m &= -2\lambda' r(r-2m) \\ \therefore m' &= \frac{1}{r}(2m - 2\lambda' r(r-2m)) \end{aligned}$$

So our differential equation for the mass with respect to the radius is,

$$\frac{dm}{dr} = \frac{2m}{r} - 2\frac{d\lambda}{dr}(r - 2m) \tag{30}$$

where,

$$\frac{d\lambda}{dr} = \frac{4\pi r e^{2\lambda}}{3f'}(2e + 3p) + \frac{(1 - e^{2\lambda})}{2r} + \frac{r(f + Rf')e^{2\lambda}}{12f'} - \frac{rf''}{2f'}\frac{d\nu}{dr}Q. \tag{31}$$

And along with Eq. 30 the remainder of the TOV equations for f(R) gravity are,

$$\frac{d\nu}{dr} = 4\pi\zeta r e^{2\lambda} p + \frac{\zeta(e^{2\lambda} - 1)f'}{2r} + \frac{\zeta r(f - Rf')e^{2\lambda}}{4} - \zeta f'' Q \tag{32}$$

$$\frac{dR}{dr} = Q \tag{33}$$

$$\frac{dQ}{dr} = (\frac{d\lambda}{dr} - \frac{d\nu}{dr} - \frac{2}{r})Q - \frac{f'''}{f''}Q^2 + \frac{e^{2\lambda}(2f - Rf')}{3f''} - \frac{8\pi e^{2\lambda}(e - 3p)}{3f''} \tag{34}$$

$$\frac{dp}{dr} = -(p + e)\frac{d\nu}{dr}. \tag{35}$$

Now that we have the TOV equations for both the GR case and $f(R)$, we make efforts to solve for the different quantities which in our case, the quantities are for neutron stars. In the following section we present the numerical methods used to solve the aforementioned TOV equations.

3 Numerical Methods

The Einstein field equations allow us to recover a set of differential equations which are difficult to solve analytically because of their nonlinearity. The differential equations recovered here are the TOV equations shown in Sec. 2. We therefore have to employ numerical methods in order to approximate as best as we can, solutions to the specified differential equations. There are a lot of numerical integration techniques at our disposal, but the one primarily used here is the Runge-Kutta method.

3.1 The Runge-Kutta Method

The Runge-Kutta method is very powerful because it has a high-order local truncation error, but unlike most Taylor methods, there is no need to compute derivatives of functions within the method. The method of order four is the most commonly used Runge-Kutta method, and it is represented below by the difference equation form given in [3],

$$\begin{aligned} w_0 &= \gamma \\ k_1 &= hf(t_i, w_i) \\ k_2 &= hf\left(t_i + \frac{h}{2}, w_i + \frac{k_1}{2}\right) \\ k_3 &= hf\left(t_i + \frac{h}{2}, w_i + \frac{k_2}{2}\right) \\ k_4 &= hf(t_{i+1}, w_i + k_3) \\ w_{i+1} &= w_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4), \end{aligned}$$

where $w_0 = \gamma$ represents our initial condition and w_{i+1} is the update of our numerical solution through each iteration of $i = 0, 1, \dots, N-1$, with N representing the number of mesh-points in our chosen interval.

The differential equations we recovered from the Einsteins Field equations are coupled, and as such represent a system of differential equations. Henceforth we extend the Runge-Kutta method to solve for a system of DEs. The generalized version of the Runge-Kutta method as shown in [3], is as follows:

We partition our interval $[a, b]$ into N subintervals with meshpoints $t_i = a + jh$ for $j = 0, 1, \dots, N$ with $N > 0$ and $h = \frac{b-a}{N}$. If we have m differential equations in our system, we can represent them as,

$$\begin{aligned} \frac{du_1}{dt} &= f_1(t, u_1, u_2, \dots, u_m) \\ \frac{du_2}{dt} &= f_2(t, u_1, u_2, \dots, u_m) \\ &\vdots \\ \frac{du_m}{dt} &= f_m(t, u_1, u_2, \dots, u_m), \end{aligned}$$

for $a \leq t \leq b$ with initial conditions, $u_1(a) = \gamma_1, u_2(a) = \gamma_2, \dots, u_m = \gamma_m$. As such in the algorithm, we can use w_{ij} to denote the approximation to our solution $u_i(t_j)$ and as such the initial conditions can be denoted as $w_{1,0} = \gamma_1, w_{2,0} = \gamma_2, \dots, w_{m,0} = \gamma_m$.

We can therefore update our set of solutions by obtaining $w_{1,j+1}, w_{2,j+1}, \dots, w_{m,j+1}$ assuming we've found the values $w_{1,j}, w_{2,j}, \dots, w_{m,j}$ by calculating

$$
\begin{aligned}
k_{1,i} &= hf_i(t_j, w_{1,j}, w_{2,j}, \ldots, w_{m,j}) \\
k_{2,i} &= hf_i\left(t_j + \frac{h}{2}, w_{1,j} + \frac{1}{2}k_{1,1}, w_{2,j} + \frac{1}{2}k_{1,2}, \ldots, w_{m,j} + \frac{1}{2}k_{1,m}\right) \\
k_{3,i} &= hf_i\left(t_j + \frac{h}{2}, w_{1,j} + \frac{1}{2}k_{2,1}, w_{2,j} + \frac{1}{2}k_{2,2}, \ldots, w_{m,j} + \frac{1}{2}k_{2,m}\right) \\
k_{4,i} &= hf_i(t_j + h, w_{1,j} + k_{3,1}, w_{2,j} + k_{3,2}, \ldots, w_{m,j} + k_{3,m}) \\
w_{i,j+1} &= w_{i,j} + \frac{1}{6}(k_{1,j} + 2k_{2,i} + 2k_{3,i} + k_{4,i}).
\end{aligned}
$$

Now that the procedure has been outlined, we can then apply that to our system of differential equations 22, 24 and 26 derived from the Einstein field equations. We first denote our variables of interest using $u_i(r)$,

$$u_1(r) = m(r) \tag{36}$$

$$u_2(r) = p(r) \tag{37}$$

$$u_3(r) = \nu(r) \tag{38}$$

So therefore our system can be represented as follows,

$$
\begin{aligned}
u_1'(r) &= 4\pi r^2 e \\
u_2'(r) &= -\frac{(e + u_2)(u_1 + 4\pi r^3 u_2)}{r(r - 2u_1)} = -(e + u_2)u_3' \\
u_3'(r) &= \frac{u_1 + 4\pi r^3 u_2}{r(r - 2u_1)}.
\end{aligned}
$$

So following the sample code written in Python as seen in Appendix B, a function was defined that takes in the latest approximated solution and updates it using our system of DEs above $f(t, u_1, u_2, u_3) = [u_1', u_2', u_3']$. We then ran the function through a loop implementing the Runge-Kutta method. The loop is set to stop when the pressure reaches a value very close to zero, this is because we integrate from the center of the neutron star to its edge, by updating the radius with the value of $h = (b - a)/N$. As such, the pressure would be zero outside the neutron star, we however avoided stopping the loop at $p(r) = 0$ to avoid a division by zero error.

Having applied the Runge-Kutta method to the TOV equations generated from the Einstein Field Equations in General Relativity, we do the same procedure for the TOV equations in $f(R)$ gravity. First we need to define the function $f(R)$ in Python along with it's derivatives as well as the function $\zeta(R)$ as shown in equations 30, 32, 33, 34 and 35 because they all depend on the Ricci scalar R, and we see in our $f(R)$ TOV equations that the Ricci scalar will be updated as well after every iteration in our loop.

From there we wish we to denote our variables of interest in our DEs in a convenient manner as we did for the GR case above.

$$
\begin{aligned}
u_1 &= m(r) \\
u_2 &= \nu(r) \\
u_3 &= R(r) \\
u_4 &= Q(r) \\
u_5 &= p(r)
\end{aligned}
$$

So then our system of differential equations can be represented as,

$$
\begin{aligned}
u_1'(r) &= \frac{2u_1}{r} - 2\frac{d\lambda}{dr}(r - 2u_1) \\
u_2'(r) &= 4\pi Z r e^{2\lambda} u_5 + \frac{Z(e^{2\lambda} - 1)f'}{2r} + \frac{Zr(f - u_3 f')e^{2\lambda}}{4} - Zf'' u_4 \\
u_3'(r) &= u_4 \\
u_4'(r) &= \left(\frac{d\lambda}{dr} - u_2' - \frac{2}{r}\right)u_4 - \frac{f'''}{f''}u_4^2 + \frac{e^{2\lambda}(2f - u_3 f')}{3f''} - \frac{8\pi e^{2\lambda}(e - 3u_5)}{3f''} \\
u_5'(r) &= -(u_5 + e)u_2'
\end{aligned}
$$

with e as the energy density[2], and $Z = 2(2f' + rf'' u_4)^{-1}$ represents the function ζ in Eq. 28 and $d\lambda/dr$ represents Eq. 31 and in conjunction with the above system can be written as,

$$
\frac{d\lambda}{dr} = \frac{4\pi r e^{2\lambda}}{3f'}(2e + 3u_5) + \frac{(1 - e^{2\lambda})}{2r} + \frac{r(f + u_3 f')e^{2\lambda}}{12f'} - \frac{rf''}{2f'}u_2' u_4.
$$

The system is yet again encompassed in a Python function that takes in the approximated solution and updates it [see sample code in Appendix B]. We once again run our function through the Runge-Kutta process and choose our loop to stop when it reaches a value very close to zero for the same motivation as the GR case.

3.2 The Shooting Method

Choosing the initial conditions for GR case is rather straightforward as will be shown in Sec. 4. When dealing with the $f(R)$ TOV equations however, we encounter a little complication. When the integration completes (i.e. at the boundary of the star), we require the Ricci scalar to be zero, or as close as possible to zero, so choosing the initial value for the Ricci scalar is not so trivial.

We employ a procedure known as the Shooting method in order to remedy this issue. This method becomes relevant in this case becasue we now have a boundary value problem and we wish to change one boundary to give results corresponding to the fixed boundary. Given a boundary value problem,

$$
y'' = f(x, y, y'), \tag{39}
$$

on the interval $a \leq x \leq b$, with boundary conditions $y(a) = \gamma,\ y(b) = \beta$, the shooting method say we approximate the solution to the BVP by using solutions to a sequence of initial value problems involving a parameter t [3]. So taking Eq. 39, we bring in the interval to $a \leq x \leq 1$, and use the initial conditions $y(a) = \gamma,\ y'(a) = t$. The value of $t = t_k$ needs to be chosen such that,

$$
\lim_{k \to \infty} y(b, t_k) = y(b) = \beta, \tag{40}
$$

where the function $y(x, t_k)$ will be the solution to the initial value problem, this will imply that $y(x)$ is the solution to the boundary value problem. The value of t is chosen via a root finding method, and here we use the method of Bisection. Starting with a value t_0, we iterate using the Bisection method until $y(b, t) - \beta \approx 0$, which implies $y(b, t)$ will be close enough to β as we require by Eq. 40.

We avoid using Newton's method, Secant Method or any other method of that kind to not have to deal with a division-by-zero error in case the derivative of $y(b, t)$ vanishes. Such issues do not affect the Bisection method so it is a reliable root finding method. Below the pseudo-code for the bisection method is presented.

[2] e is the energy density and not to be confused with the exponential as in $e^{2\lambda}$

Set interval $[p, q]$ such that
$t_k \leftarrow \frac{p+q}{2}$
if $g(t_k) \leq 0$ **then**
    $p \leftarrow t_k$
    if $g(t_k) \geq 0$ **then**
        $q \leftarrow t_k$
    end if
else
    if $g(t_k) = 0$ **then**
        $p = q$
    end if
end if

Unfortunately due to time constraints, the Shooting Method could not be implemented by the author, however results are still presented in Sec. 4 below using code provided by my supervisor.

4 Results and Discussion

When numerically integrating the TOV equations for GR, we do so starting out at the center of the neutron star, i.e. when the radius $r = 0$, until the radius when the pressure $p = 0$ (i.e., when we reach the edge of the star) so that we may end up with the radius and mass of the simulated neutron star. Since the TOV equations are not defined at $r = 0$, we will instead begin the integration at a point very close to zero. For the initial conditions, the mass is of course initialized at zero at the center of the star, we take $\nu(0) = 1$ whereby this can be re-scaled after the integration if we need the solution to match with the Schwarszchild solution [8]. And to initialize the pressure, we use the polytropic equation of state [as discussed in Sec. 1.3]. The central density of the neutron star is given to be $\rho_c = 5.87 \times 10^{-4}$ and using that along with the constants $K = 217.858$ and $\Gamma = 2$ [8] we get the initial pressure to be,

$$p(r = 0) = p(\rho_c) = K\rho_c^{\Gamma},$$

with the energy density in terms of the pressure given by,

$$e = \rho + \frac{p}{\Gamma - 1} = \left(\frac{p}{K}\right)^{1/\Gamma} + \frac{p}{\Gamma - 1},$$

and can be initialized using the initial pressure.

For the $f(R)$ equations, we maintain the initial conditions for the variables corresponding with the GR case (m, ν, p) , and we take $Q = 0$ initially as well. For the Ricci scalar we are required to take a different route. We need the Ricci scalar to vanish on the boundary of the star so we need an initial R_c such that $R \approx 0$ when $r \geq r_{NS}$, where r_{NS} is the radius of the neutron star. This can be achieved using the shooting method as outlined in Sec. 3.2. In this instance we will start by using a predetermined $R_c = 0.01028206$ which can be verified when using the shooting method. So with this initial conditions in place, we can plot the solutions to the TOV equations for both the GR and $f(R)$ cases on the same set of axes as shown in the figures below.

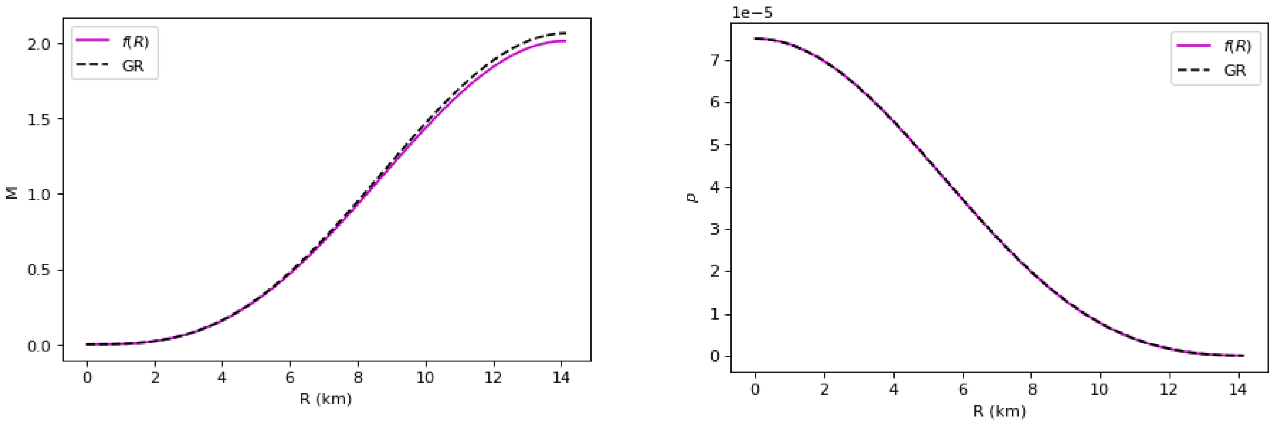

Figure 1: The figures above show the mass and pressure of the neutron star evolving with the radius of the star. This is calculated for both the GR and $f(R)$ cases and found that for the GR case, the mass of the neutron star is $m = 2.01364$ with a radius $R = 14.15$, and for the $f(R)$ case we got a slightly lower mass at $m = 2.01364$ and a radius of $R = 14.1195$.

It can be seen from the above figures that $f(R)$ gravity simulates a neutron star with a similar mass and radius to general relativity. The same can be attributed to the sound speed as well [as discussed in Sec. 1.3.1]. Using the results from Figure 1, we can calculate the sound speed using the relativistic sound speed equation 9.

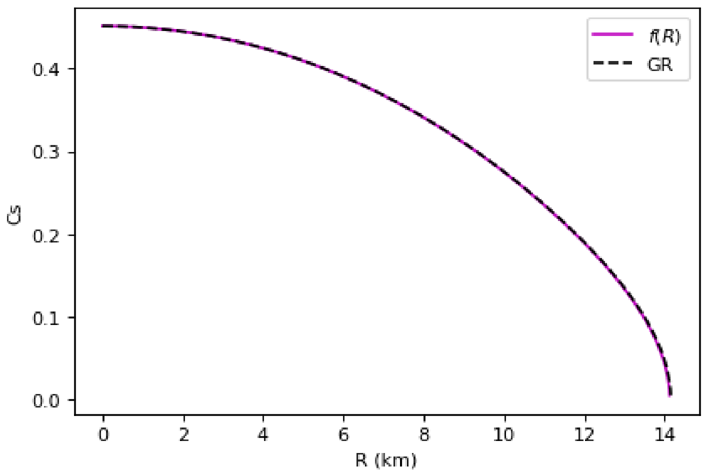

Figure 2: In this figure, the sound speed is calculated to be $Cs = 0.45122$ at the center of the star, and then decreases to vanish once we reach the edge of the star at a radius of $R \approx 14.1$. This trend holds for both the GR and $f(R)$ cases.

In both Figures 1 and 2, the simulation for neutron stars in $f(R)$ gravity results in the calculated quantities resembling those of neutron stars simulated in general relativity. We did take a few liberties in the process of achieving said results, the biggest being our choice of the parameter α. Using the model depicted by Eq. 10, we chose $\alpha = 0.3$. The choice of $\alpha = 0$ recovers Einstein's general relativity as already mentioned however, if we choose $\alpha < 0.3$, we run this risk of having a stiff system. We do however, test values of $\alpha > 0.3$ to see how neutron stars in $f(R)$ gravity evolve at different values of α.

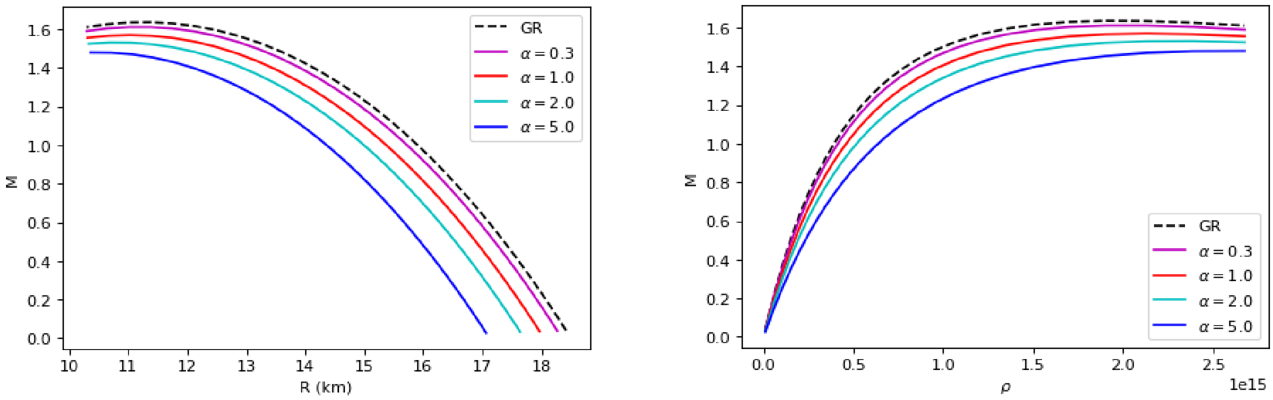

Figure 3: The figures above depict the evolution of neutron stars GR and in $f(R)$ gravity when the parameter α in $f(R) = R + \alpha R^2$ is varied.

In Figure 3, the Shooting method is implemented as discussed in Sec. 3.2, and an initial value for the Ricci scalar R_c is chosen such that the Ricci scalar vanishes at the edge of the neutron star. It can be seen that the maximal mass and radius of the neutron star decrease as the value of α is increased with the density at the middle of the star and at the edge remaining the same in all models. This implies that stars calculated with a softer equation of state (in contrast to a stiffer one), the maximum mass will be smaller and as such the central density will also be higher for stars with a softer equation of state.

5 Conclusion

Since the introduction of General Relativity in 1915 by Einstein, science has rapidly progressed to what we see it as today, and over a century later, GR is still a widely accepted fundamental theory of gravity. Modified gravity does however, present an opportunity to further understand our universe as well as GR itself and as shown in this paper using static spherically symmetric neutron stars, metric $f(R)$ gravity remains consistent with GR as well as expanding upon it.

We derived the TOV equations from the Einstein field equations and used those, along with the TOV equations recovered from the field equations of $f(R)$ gravity, to study properties of neutron stars. To visualize these properties, we numerically solved the TOV equations by using the Runge-Kutta integrator as well as the Shooting method for the $f(R)$ case in order to handle the initial Ricci scalar value R_c.

We then varied the parameter α in the Starobinksy R-squared model given by $f(R) = R + \alpha R^2$ and from the acquired results we concluded that the this is a viable model of gravity and can be tested with other observations. The Starobinksy model represents the simplest model of $f(R)$ gravity with the polytropic equation of state being the simplest equation of state, we have seen the merit of studying these simpler cases in this paper however, there are other much more complicated choices that could have been made and much of them are explored as one further studies modified gravity. There is also much interest and opportunity to study these $f(R)$ models of neutron stars in simulations of binary neutron star collisions in 3D numerical relativity.

Acknowledgements

I would like to thank my supervisor, Dr. Bishop Mongwane, for guiding me through the process of writing a thesis and the patience when I was struggling with my code. I would also like to thank my Honours classmates for the emotional support throughout the completion of this thesis, as well as everyone who read and gave useful feedback. And lastly, I would like to thank all the family and friends who have supported me throughout the year.

References

[1] W. Baade and F. Zwicky. Remarks on super-novae and cosmic rays. *Physical Review*, 46(1):76, 1934.

[2] A. Briggs. What is a neutron star? `https://earthsky.org/astronomy-essentials/definition-what-is-a-neutron-star/`, 2020. Accessed: July 2022.

[3] R. L. Burden, J. D. Faires, and A. M. Burden. *Numerical analysis*. Cengage learning, 2015.

[4] S. Capozziello, M. De Laurentis, R. Farinelli, and S. D. Odintsov. Mass-radius relation for neutron stars in f(r) gravity. *Physical Review D*, 93(2):023501, 2016.

[5] V. Dzhunushaliev and V. Folomeev. On the linear stability of polytropic fluid spheres in r 2 gravity. *International Journal of Geometric Methods in Modern Physics*, 17(11):2050165, 2020.

[6] R. Giacconi, H. Gursky, F. R. Paolini, and B. B. Rossi. Evidence for x rays from sources outside the solar system. *Physical Review Letters*, 9(11):439, 1962.

[7] M. P. Hobson, G. P. Efstathiou, and A. N. Lasenby. *General relativity: an introduction for physicists*. Cambridge University Press, 2006.

[8] B. Mongwane. Relativistic Neutron Stars Beyond General Relativity. [notes sent by author].

[9] J. R. Oppenheimer and G. M. Volkoff. On massive neutron cores. *Physical Review*, 55(4):374, 1939.

[10] M. A. Resco, Á. de la Cruz-Dombriz, F. J. L. Estrada, and V. Z. Castrillo. On neutron stars in f(r) theories: Small radii, large masses and large energy emitted in a merger. *Physics of the dark universe*, 13:147–161, 2016.

[11] L. Rezzolla and O. Zanotti. *Relativistic hydrodynamics*. Oxford University Press, 2013.

[12] S. L. Shapiro and S. A. Teukolsky. *Black holes, white dwarfs, and neutron stars: The physics of compact objects*. John Wiley & Sons, 2008.

Appendices

A Non-zero Christoffel Symbols

Presented below are the calculations for the non-zero Christoffel symbols in an effort to calculate the curvature quantities from the given spacetime metric.

$$\begin{aligned}{\Gamma^r}_{tt} &= -\frac{1}{2}g^{rr}(g_{tt,r}) \\ &= -\frac{1}{2}(e^{-2\lambda})\frac{d}{dr}(-e^{2\nu}) \\ &= e^{-2\lambda}e^{2\nu}\nu'(r).\end{aligned}$$

$$\begin{aligned}{\Gamma^r}_{rr} &= \frac{1}{2}g^{rr}(g_{rr,r}) \\ &= -\frac{1}{2}(-e^{-2\lambda})\frac{d}{dr}(e^{2\lambda}) \\ &= \lambda'(r).\end{aligned}$$

$$\begin{aligned}{\Gamma^r}_{\theta\theta} &= \frac{1}{2}g^{rr}(-g_{\theta\theta,r}) \\ &= -\frac{1}{2}(e^{-2\lambda})\frac{d}{dr}(r^2) \\ &= -re^{-2\lambda}.\end{aligned}$$

$$\begin{aligned}{\Gamma^r}_{\phi\phi} &= \frac{1}{2}g^{rr}(-g_{\phi\phi,r}) \\ &= -\frac{1}{2}(e^{-2\lambda})\frac{d}{dr}(r^2\sin^2\theta) \\ &= -r\sin^2\theta e^{-2\lambda}.\end{aligned}$$

$$\begin{aligned}{\Gamma^\theta}_{r\theta} &= \frac{1}{2}g^{\theta\theta}(g_{\theta\theta,r}) \\ &= -\frac{1}{2}(\frac{1}{r^2})\frac{d}{dr}(r^2) \\ &= \frac{1}{r} = {\Gamma^\theta}_{\theta r}.\end{aligned}$$

$$\begin{aligned}{\Gamma^\theta}_{\phi\phi} &= \frac{1}{2}g^{\theta\theta}(-g_{\phi\phi,\theta}) \\ &= -\frac{1}{2}(\frac{1}{r^2})\frac{d}{d\theta}(r^2\sin^2\theta) \\ &= -\sin\theta\cos\theta.\end{aligned}$$

$$\begin{aligned}{\Gamma^\phi}_{r\phi} &= \frac{1}{2}g^{\phi\phi}(g_{\phi\phi,\theta}) \\ &= \frac{1}{2}(\frac{1}{r^2\sin^2\theta})\frac{d}{dr}(r^2\sin^2\theta) \\ &= \frac{1}{r} = {\Gamma^\phi}_{\phi r}.\end{aligned}$$

$$\begin{aligned}{\Gamma^\phi}_{\theta\phi} &= \frac{1}{2}g^{\phi\phi}(g_{\phi\phi,\theta}) \\ &= \frac{1}{2}(\frac{1}{r^2\sin^2\theta})\frac{d}{d\theta}(r^2\sin^2\theta) \\ &= \frac{\sin\theta\cos\theta}{\sin^2\theta} \\ &= \cot\theta = {\Gamma^\phi}_{\phi\theta}.\end{aligned}$$

B Python Code Samples

Shown below is the Python code written to numerically solve the TOV equations for both cases of general relativity and metric $f(R)$ gravity. The code can be run with any python installation and the additional libraries numpy and matplotlib.

B.1 General Relativity

```
def f(r_,u1,u2,u3):
    e = ((u2/K)**(1/G))+(u2/(G-1))
    return [4*np.pi*(r_**2)*e, -((e+u2)*(u1+4*np.pi*(r_**3)*u2))/(r_*(r_-2*u1)),
        ((u1+4*np.pi*(r_**3)*u2))/(r_*(r_-2*u1))]
```

```
while w[1] > 1e-8 :
    for j in range(m):
        K1[j] = h*f(r,w[0],w[1],w[2])[j]
```

```
    for j in range(m):
        K2[j] = h*f(r+h/2,w[0]+K1[0]/2,w[1]+K1[1]/2,w[2]+K1[2]/2)[j]
    for j in range(m):
        K3[j] = h*f(r+h/2,w[0]+K2[0]/2,w[1]+K2[1]/2,w[2]+K2[2]/2)[j]
    for j in range(m):
        K4[j] = h*f(r+h,w[0]+K3[0],w[1]+K3[1],w[2]+K3[2])[j]
    for j in range(m):
        w[j] = w[j]+(K1[j]+2*K2[j]+2*K3[j]+K4[j])/6
    r = r+h
```

B.2 Metric f(R) Gravity

```
# Metric f(R) gravity functions
def f(R_):                              # f(R)
    return R_ + Alpha*R_**2
def fp(R_):                             # f'(R)
    return 1 + 2*Alpha*R_
def fpp(R_):                            # f''(R)
    return 2*Alpha
def fppp(R_):                           # f'''(R)
    return 0.0
def zeta(r_, R_, Q_):
    return 1/(fp(R_) + (r_/2)*fpp(R_)*Q_)
```

```
def odes(r_, u1, u2, u3, u4, u5):   # m = u1, \nu = u2, R = u3, Q = u4, p = u5
    e = ((u5/K)**(1/G))+(u5/(G-1))  # e = (p/\Kappa)^{1/\Gamma} + p/(\Gamma - 1)
    Lambda = -(1/2)*np.log(1 - 2*(u1/r_)) # \lambda -(1/2)ln(1 - 2(m/r))
    exp = np.exp(2*Lambda)          # e^{2\lambda}
    Z = zeta(r_, u3, u4)            # zeta
    dnudr = (4*pi*Z*r_*exp*u5 + (Z*(exp-1)*fp(u3))/(2*r_) +
         (Z*r_*(f(u3)-u3*fp(u3))*exp)/4 - Z*fpp(u3)*u4)
                                # d\nu / dr
    dLdr = ((4*pi*r_*exp/(3*fp(u3)))*(2*e + 3*u5) + (1 - exp)/(2*r_) +
    (r_*(f(u3) + u3*fp(u3))*exp)/(12*fp(u3)) - ((r_*fpp(u3))/(2*fp(u3)))*dnudr*u4)
                                # d\lambda / dr
    return [(u1/(r_)) + (r_ - 2*u1)*dLdr, dnudr, u4,
        (dLdr - dnudr - 2/r_)*u4 - (fppp(u3)/fpp(u3))*u4*u4 +
        (exp*(2*f(u3) - u3*fp(u3)))/(3*fpp(u3)) -
        (8*pi*exp*(e - 3*u5))/(3*fpp(u3)), -(e+u5)*dnudr]
```

```
while w[4] > 1e-12 :
    for j in range(m):
        K1[j] = h*odes(r,w[0],w[1],w[2],w[3],w[4])[j]
    for j in range(m):
        K2[j] = h*odes(r+h/2,w[0]+K1[0]/2,w[1]+K1[1]/2,
```

```
        w[2]+K1[2]/2,w[3]+K1[3]/2,w[4]+K1[4]/2)[j]
for j in range(m):
    K3[j] = h*odes(r+h/2,w[0]+K2[0]/2,w[1]+K2[1]/2,
        w[2]+K2[2]/2,w[3]+K2[3]/2,w[4]+K2[4]/2)[j]
for j in range(m):
    K4[j] = h*odes(r+h,w[0]+K3[0],w[1]+K3[1],w[2]+K3[2],
        w[3]+K3[3],w[4]+K3[4])[j]
for j in range(m):
    w[j] = w[j]+(K1[j]+2*K2[j]+2*K3[j]+K4[j])/6
r = r+h
```