National Institute of Technology
Karnataka, Surathkal

Modelling of Neutron Stars

Shamant Shreedhar Dixit
(246PH027)

Outline

What are Neutron Stars?

↓

Equations of State

↓

Mass – Radius Relation

↓

Neutron Star Models

A Brief Introduction

- Main sequence → Red Giant → Supernova
- Collapsed core is called Neutron Star
- James Chadwick discovered neutron in 1932
- Walter Baade and Fritz Zwicky proposed existence of **neutron stars** in 1934
- Properties:
  - Mass: 1 ~ 3 $M_{\odot}$
  - Radius: 10 – 14 km
  - Mean Density: $> \ 10^{14}\ \mathrm{g/cm^3}$

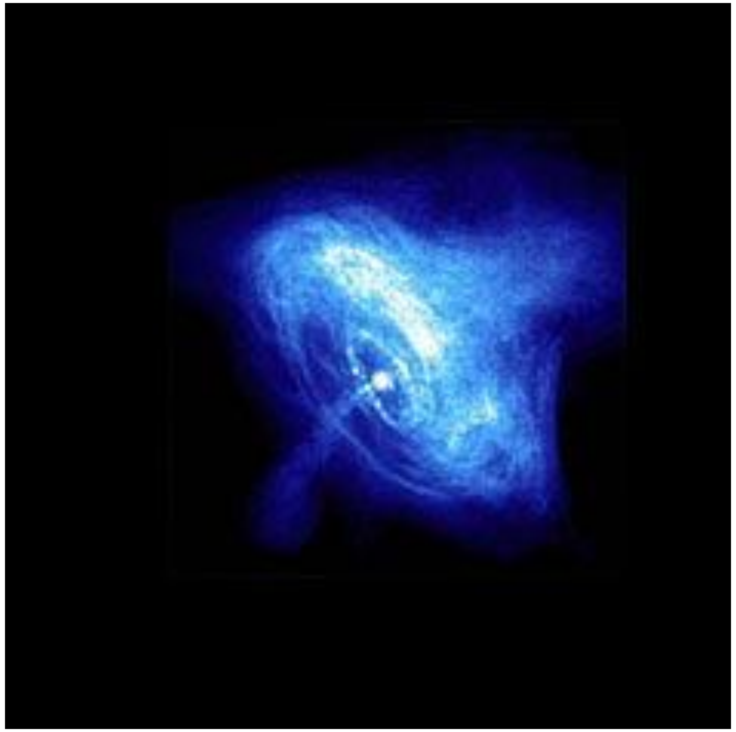

Figure 1: X-ray image of Crab Nebula (Supernova remnant) which has left a pulsar (central dot).

https://www.stsci.edu/~marel/black_holes/encyc_mod1_q7.html#:~:text=The%20Crab%20supernova%20produced%20a,Earth%2030%20times%20per%20second.

Preliminaries

- Newtonian formulation:

$$\frac{dp}{dr} = -\frac{G\rho(r)\mathcal{M}(r)}{r^2} = -\frac{G\epsilon(r)\mathcal{M}(r)}{c^2r^2}$$

$$\frac{d\mathcal{M}}{dr} = 4\pi r^2\rho(r) = \frac{4\pi r^2\epsilon(r)}{c^2}$$

$$\mathcal{M}(r) = 4\pi\int_0^r r'^2dr'\rho(r') = 4\pi\int_0^r r'^2dr'\epsilon(r')/c^2$$

- $G = 6.673 \times 10^{-8}\, dyne - cm^2/g^2$
- $\rho(r) - mass\ density\ (gm/cm^3)$
- $\epsilon(r) - energy\ density\ (ergs/cm^3)$ $\quad \epsilon(r) = \rho(r)c^2$

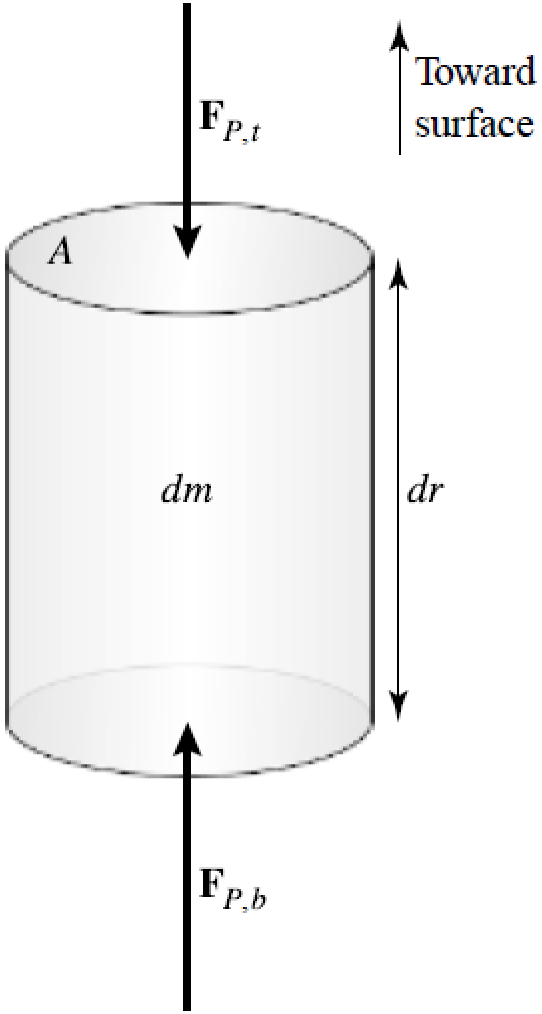

Figure 2: To illustrate **hydrostatic equilibrium** in a static star.

Preliminaries (contd..)

- General relativistic corrections:

$$\frac{dp}{dr} = -\frac{G\epsilon(r)\mathcal{M}(r)}{c^2r^2}\left[1+\frac{p(r)}{\epsilon(r)}\right]\left[1+\frac{4\pi r^3p(r)}{\mathcal{M}(r)c^2}\right]\left[1-\frac{2G\mathcal{M}(r)}{c^2r}\right]^{-1}$$

- **Tolman – Oppenheimer – Volkoff** (TOV) Equation
- Compactness factor, $GM/Rc^2 \sim 0.2\ - 0.3$ (neutron stars)

  $GM/Rc^2 \sim 10^{-4}$ (white dwarfs)

Equations of State (EoS)

- Relation between energy density $\epsilon(r)$ and pressure $p(r)$
- Fermi Gas Model:
  - Mass density of the star – $\rho = n m_N A/Z$
  - Fermi momentum – $k_F = \hbar \left(\frac{3\pi^2 \rho}{m_N} \frac{Z}{A} \right)^{1/3}$
  - Energy density – $\epsilon_n(k_F) = \frac{8\pi}{(2\pi\hbar)^3} \int_0^{k_F} \sqrt{k^2c^2 + m_n^2c^4}\, k^2\, dk$
  - Pressure density – $p_n(k_F) = \frac{8\pi}{3(2\pi\hbar)^3} \int_0^{k_F} \frac{k^4}{\sqrt{k^2c^2 + m_n^2c^4}}\, dk$

Equation of State (contd..)

- Polytrope: $$p = K\epsilon^{\gamma}$$
- In non relativistic limit, $k_F \ll m_n$, $$p = K_{\mathrm{non-rel}}\epsilon^{5/3}$$
- In relativistic limit, $k_F \gg m_n$, $$p = K_{rel}\epsilon$$
- Consider an Ansatz, $$\bar{\epsilon}(p) = A_{\mathrm{NR}}\bar{p}^{3/5} + A_{\mathrm{R}}\bar{p}$$ where

$$A_{\mathrm{NR}} = 2.4216\ , \quad A_{\mathrm{R}} = 2.8663$$ *[1]*

[1] Neutron Stars for Undergraduates (arXiV:0309041)

Neutron Star Model with Fermi Gas EoS

- Integrate the coupled first order differential equations
- From $r = 0$, to point where $\bar{p}(R) = 0$
- Runge – Kutta routine

$$\bar{\epsilon}(p) = A_{\mathrm{NR}}\bar{p}^{3/5} + A_{\mathrm{R}}\bar{p}$$

$$A_{\mathrm{NR}} = 2.4216\ , \quad A_{\mathrm{R}} = 2.8663$$

$$p = \epsilon_0\bar{p}\ , \quad \epsilon = \epsilon_0\bar{\epsilon}$$

```python
# Characteristic energy-density scale (fit for cold neutrons)
eps0 = 5.346e36              # erg cm^-3 (value from literature)

# EOS parameters for Fermi Gas model
A_NR = 2.4216  # non-relativistic limit coefficient
A_R = 2.8663   # relativistic limit coefficient

def eps_bar(p_bar):
    """Dimensionless energy density as function of dimensionless
pressure"""
    return A_NR * p_bar**(3/5) + A_R * p_bar

def eps_phys(p_phys):
    """Energy density (erg cm⁻³) from physical pressure"""
    p_bar = p_phys / eps0
    return eps0 * eps_bar(p_bar)
```

Listing 1: Python code to model Fermi Gas EoS with non relativistic and relativistic terms.

Initial Conditions and Setup

- Central pressure, $\bar{p}(0) = 0.01$
- Starting radius, $r = 0 \sim 0.01\ cm$
- Surface condition, $\bar{p}(R) = 0 \sim 10^{-8}$
- We want to find R and the corresponding maximum mass M_{max}
- Output:

Model	Radius (km)	M_{max} ($M_\odot$)
Newtonian	14.998	1.039
TOV	13.391	0.718

```python
# Central conditions and integration setup
p_c_bar = 0.01               # central dimensionless pressure
r_start = 1.0e-2             # starting radius (cm) - avoid r=0
eps_c = eps_phys(p_c)
rho_c = eps_c / c**2
M_start = (4.0/3.0) * np.pi * r_start**3 * rho_c  # initial mass

y0 = [p_c, M_start]
r_max = 2.0e7                # max integration radius (200 km)

# Surface detection condition
def surface(r, y):
    """Stop integration when pressure drops to 10^-8 eps0 (surface)"""
    return y[0] - 1.0e-8 * eps0
surface.terminal, surface.direction = True, -1

# Integrate both models
sol_newt = solve_ivp(
    newtonian_dim, (r_start, r_max), y0,
    events=surface, rtol=1e-8, atol=1e-10, max_step=5.0e4)

sol_tov = solve_ivp(
    tov_dim, (r_start, r_max), y0,
    events=surface, rtol=1e-8, atol=1e-10, max_step=5.0e4)
```

Listing 2: Python code to define initial conditions for Fermi Gas EoS model

Neutron Star Model with Fermi Gas EoS

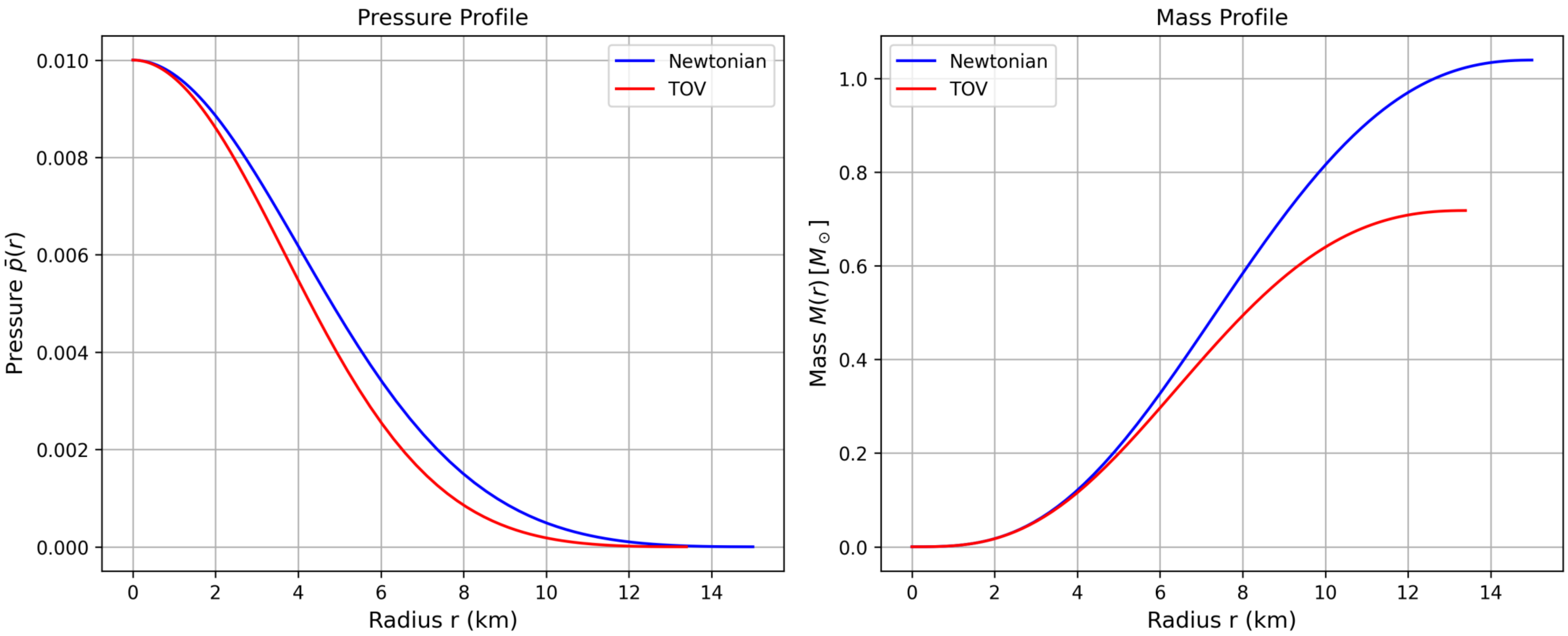

Figure 3: Pressure and mass profiles of a neutron star using a cold Fermi gas–based equation of state. The Newtonian model (blue) overestimates both radius and mass compared to the general relativistic TOV solution (red).

Mass – Radius Relation (Fermi Gas EoS)

- Plot of mass M and radius R of neutron star for a range of central pressures $\bar{p}(0)$
- Stars to the right of maximum R are stable
- Result agrees with Oppenheimer and Volkoff's seminal 1939 result [2], $M \sim 0.7\ M_{\odot}$
- Limitation: Ignores nucleon – nucleon interaction, pure neutron gas

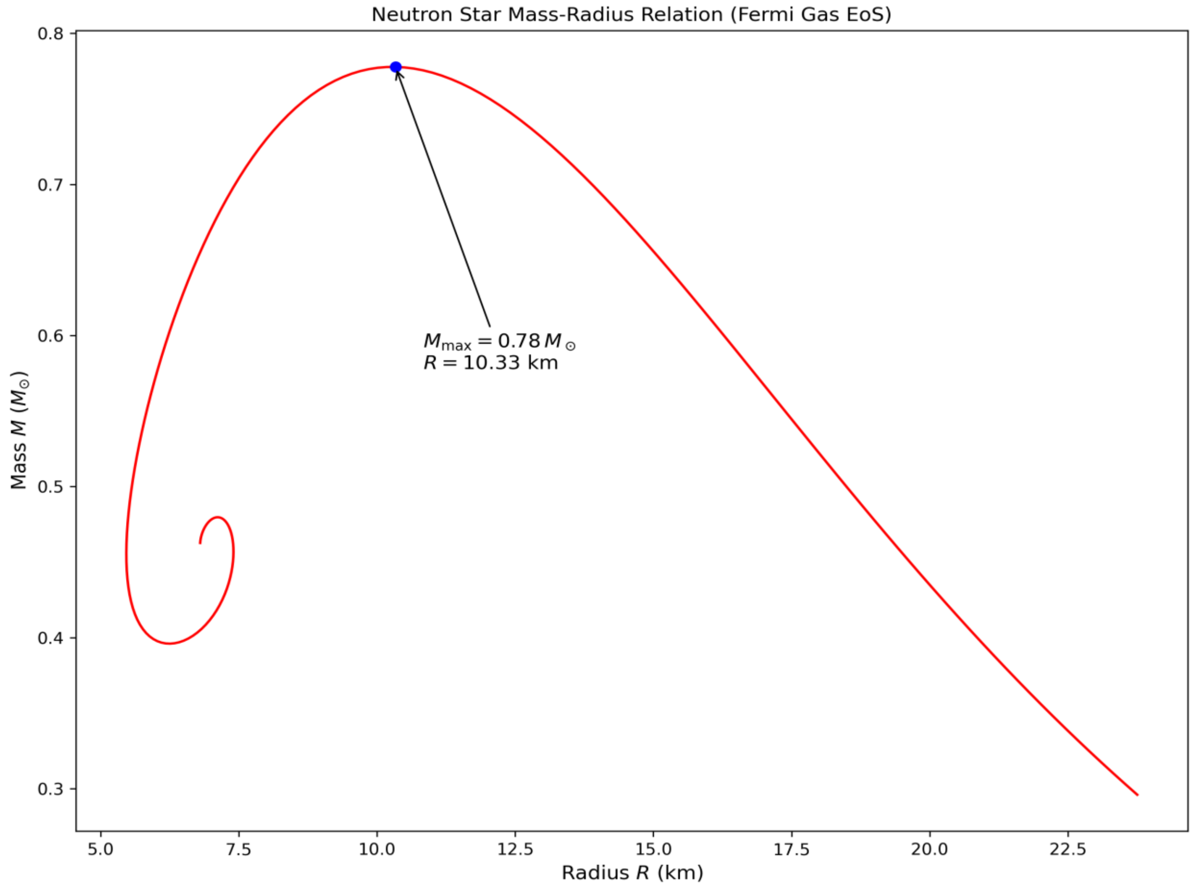

Figure 4: The mass M and radius R for pure neutron stars, using a Fermi gas EoS. The stars of low mass and large radius are solutions of the TOV equations.

[2] Oppenheimer & Volkoff(1939), On Massive Neutron Cores

Nuclear Interactions

- Solve TOV equations with a new EoS which considers nuclear interaction in pure neutron matter

$$\bar{\epsilon}(\bar{p}) = (\kappa_0 \epsilon_0)^{-1/2} \bar{p}^{1/2} = A_0 \bar{p}^{1/2}\,, \quad A_0 = 0.8642$$

- Nuclear Incompressibility, $K_0 = 363\ MeV\ (200 \sim 400\ MeV)$
- The more incompressible something is, the more mass it can support
- Limitations: Still pure neutron matter!

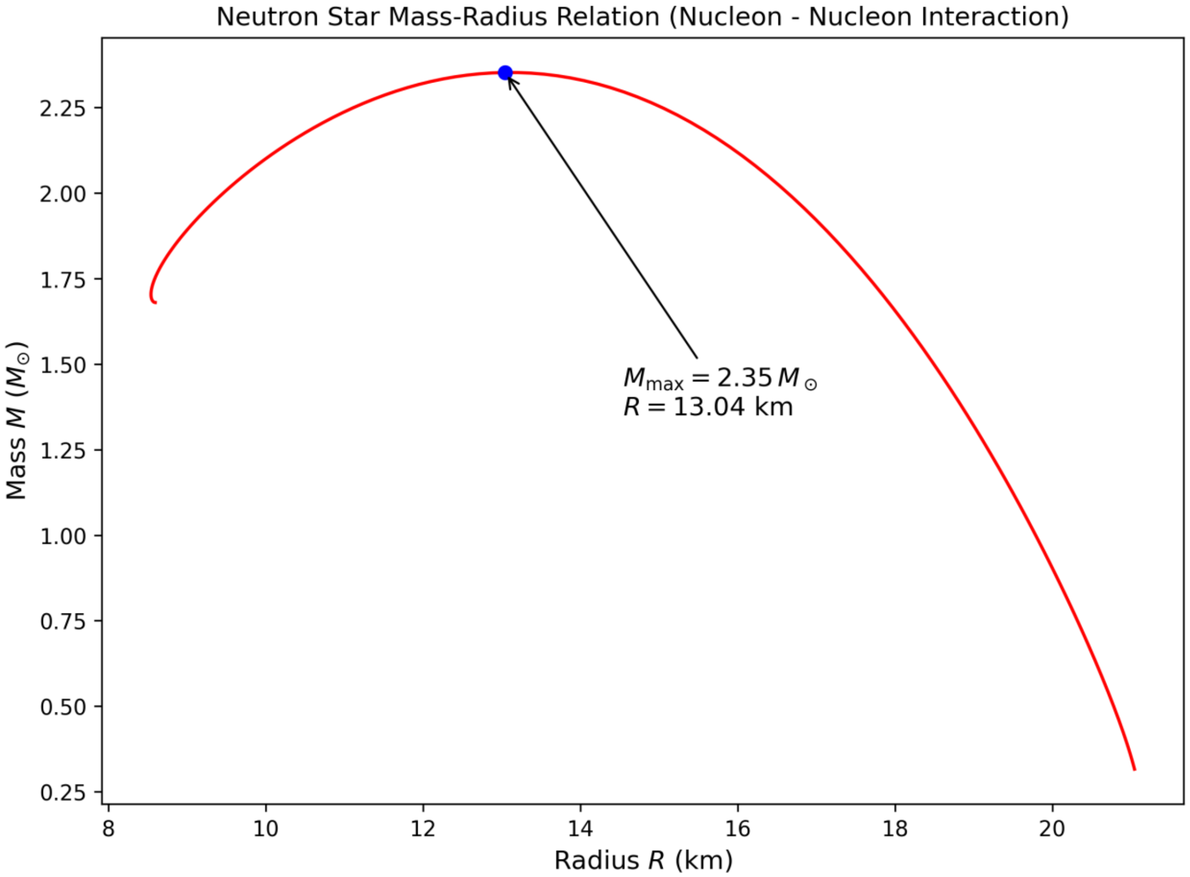

Figure 5: The mass M and radius R for pure neutron stars using an EoS which contains nucleon-nucleon interactions. Only those stars to the right of the maximum are stable against gravitational collapse.

Equation of State (EoS) Tables

- Analytic equations fail to account for non-linearities in the densities, nuclear interactions, phase transitions
- EoS tables available on CompOSE (CompStar Online Supernovae Equations of State)
- Commonly used models to study mass – radius relation:
  SLy, APR, BSk series

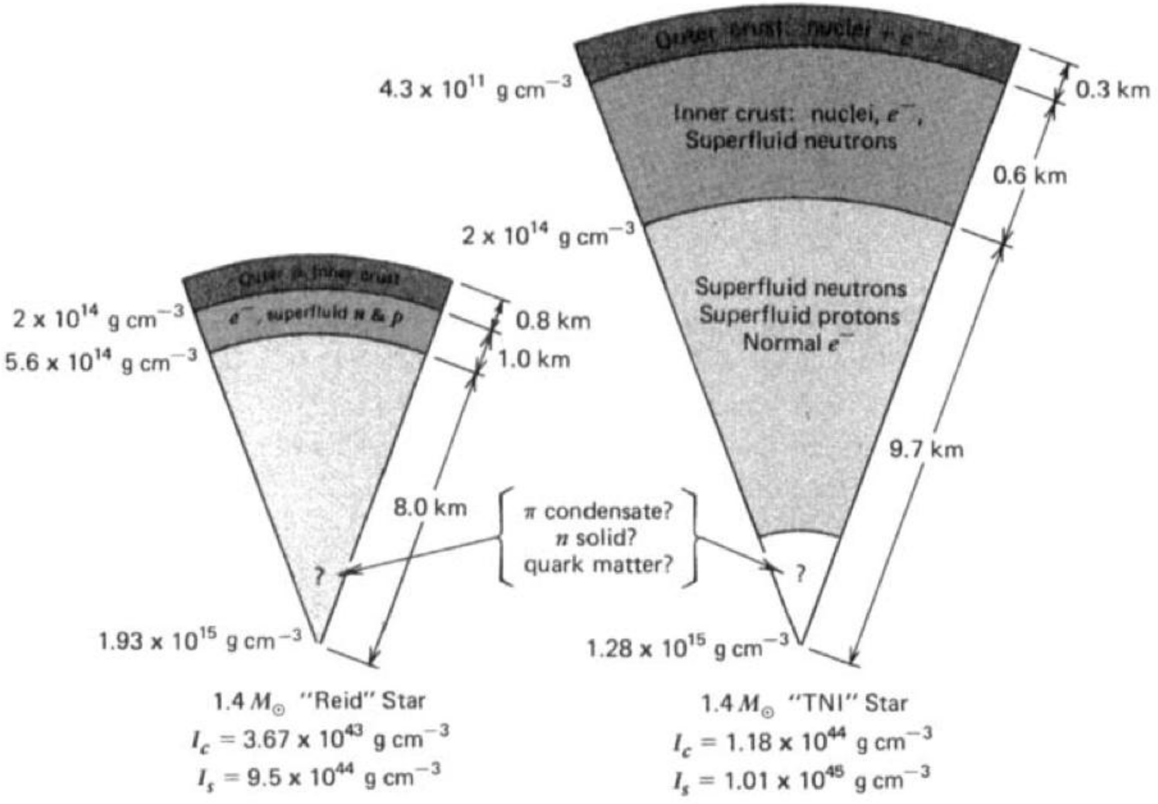

Figure 6: Cross section of 1.4 $M_{\odot}$ ‘Reid’ and ‘TNI’ neutron star models illustrating different regions of a neutron star. [3]

[3] Teukolsky & Shaprio, The Physics of Compact Objects

Neutron Star Model using Skyrme Lyon (SLy) EoS

- Interpolate the EoS data to get relations between pressure, mass density and baryon density
- Cubic Spline interpolation
- Integrate the TOV equations
- From $r = 0$, to point where $\bar{p}(R) = 0$
- For a given central density, calculate $p(0)$ using interpolated EoS
- Runge – Kutta routine

```python
# Load data
sly_data = np.genfromtxt("SLy.txt", delimiter="  ")
baryon_density = sly_data[:, 1]  # fm^-3
mass_density = sly_data[:, 2]    # g/cm^3
pressure = sly_data[:, 3]        # dyne/cm^2

# Create interpolation functions
pressure_from_density = CubicSpline(mass_density, pressure)
density_from_pressure = CubicSpline(pressure, mass_density)
baryon_density_from_pressure = CubicSpline(pressure, baryon_density)
```

Listing 3: Python code for Cubic Spline interpolation of Sly EoS table.

Mass – Radius Relation (SLy EoS)

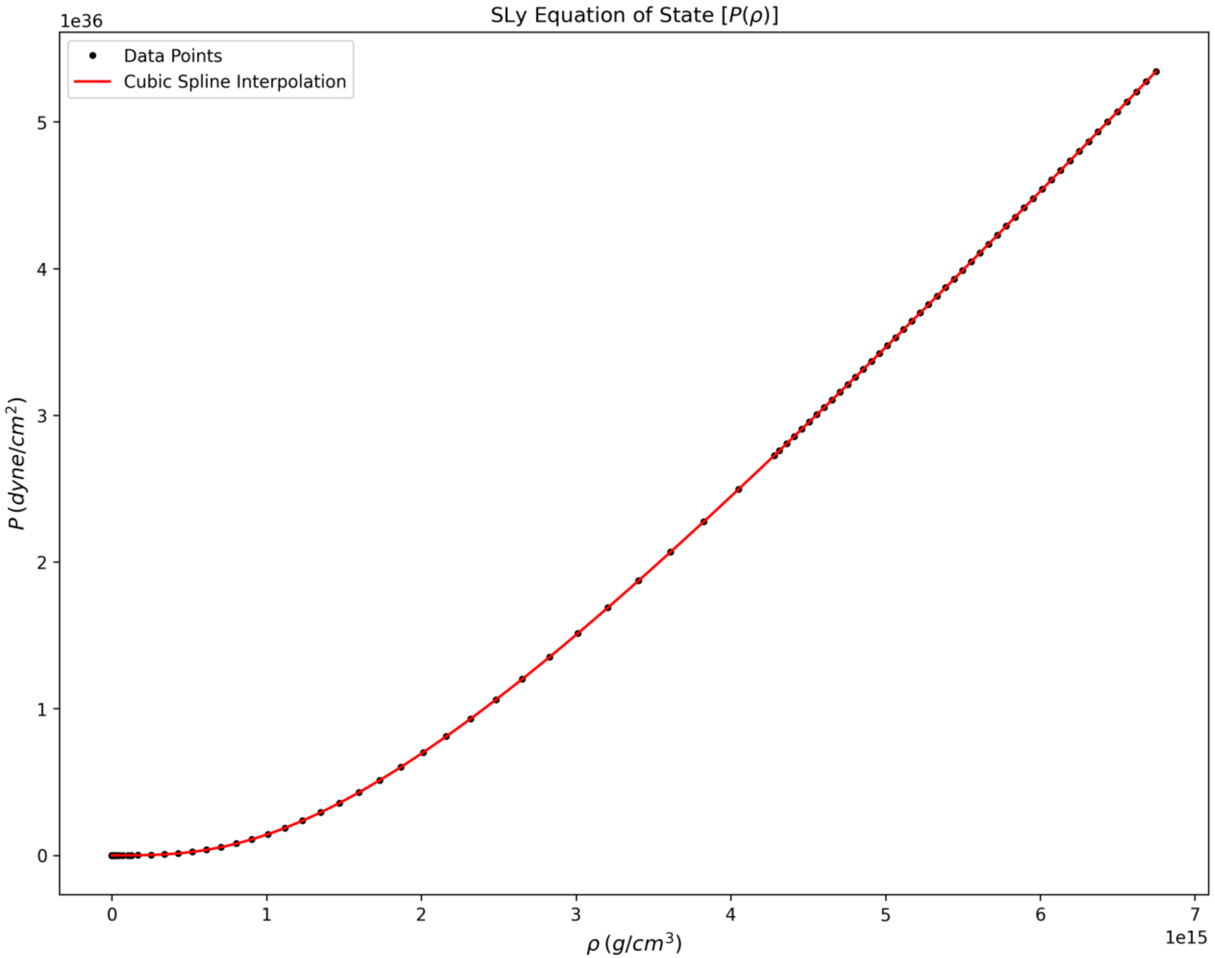

Figure 7: Cubic Spline interpolation of pressure vs mass density of SLy EoS data.

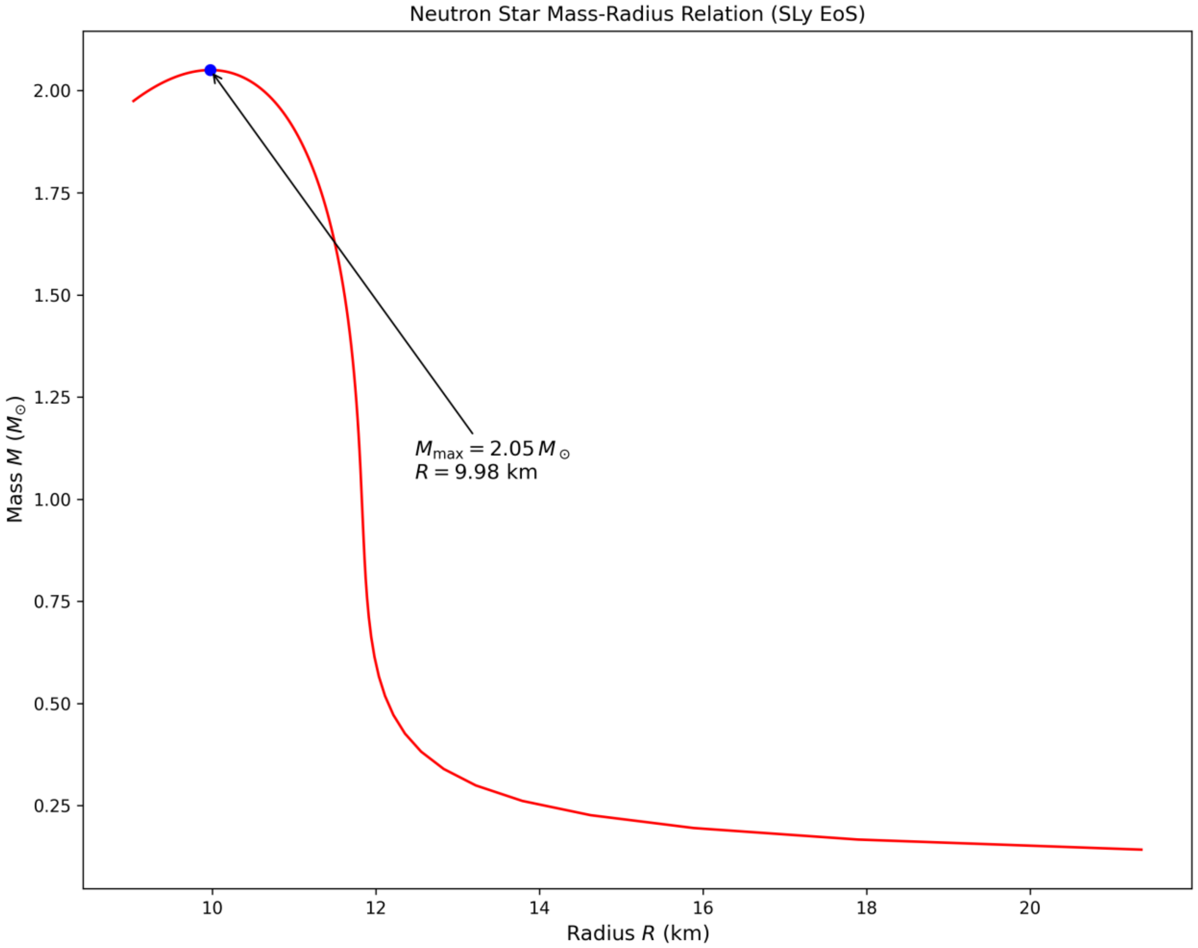

Figure 8: The mass M and radius R for neutron stars using SLy EoS which is a more realistic EoS.

Conclusion

- TOV equations with different EoS clarify mass – radius relation
- Realistic neutron star models (Mass: $1 \sim 3\ M_{\odot}$) require n-p-e mixture and nucleon interactions
- Better models can be generated by incorporating rotations
- Study of EoS is an active field of research in Nuclear Physics
- Code will be made available on Github

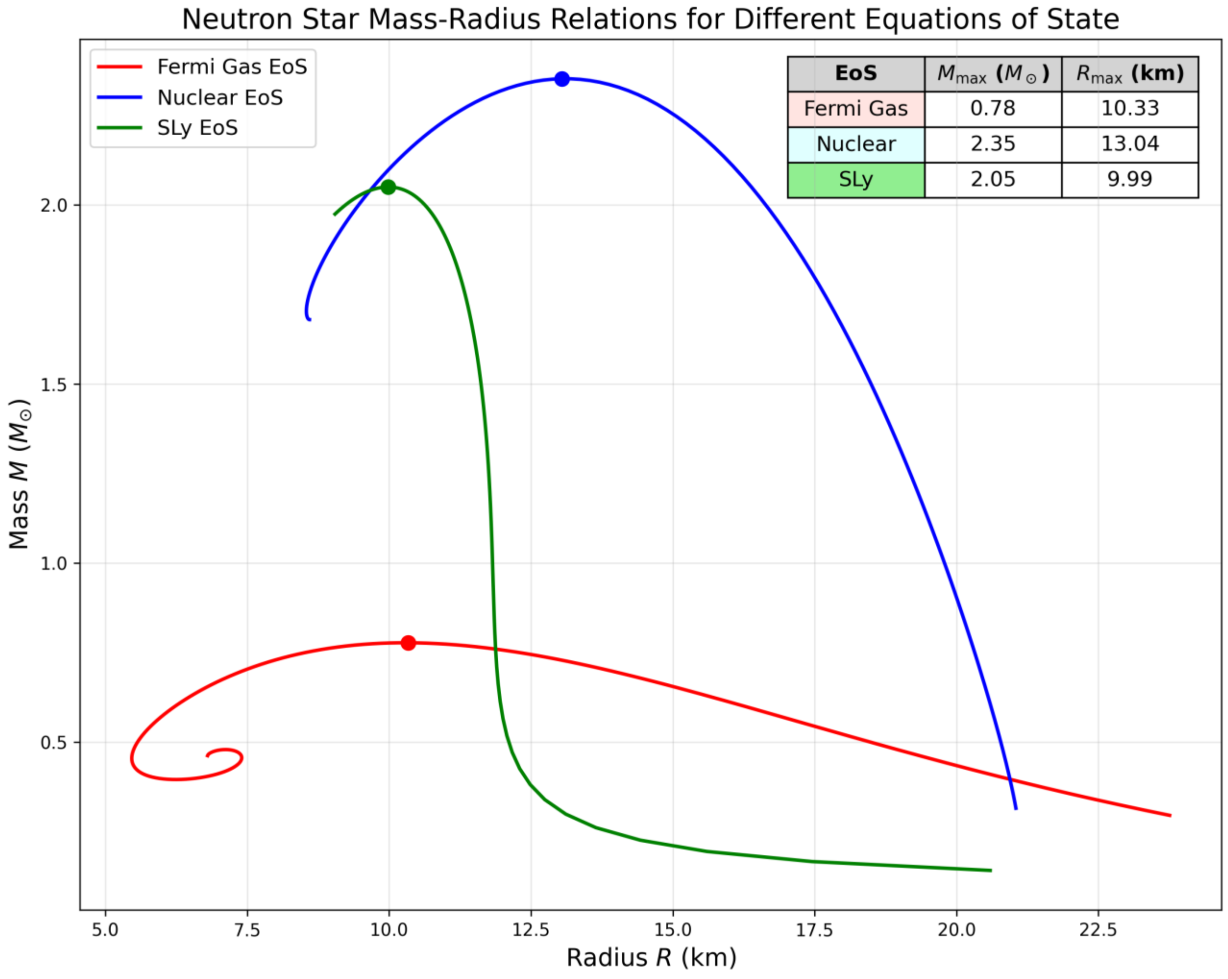

Figure 9: Mass-radius curves for neutron stars with three different equations of state. Maximum masses and corresponding radii are indicated by dots on each curve.

Thank You

Github: https://github.com/shamantdixit