Analysis of Equations of State for Neutron Star Modeling and Simulation

Joseph Nyhan

College of the Holy Cross

6 May 2022

Outline

Outline

- What is a neutron star?

Outline

- What is a neutron star?
- What is an equation of state (EoS)?

Outline

- What is a neutron star?
- What is an equation of state (EoS)? How does it fit into our model of a neutron star?

Outline

- What is a neutron star?
- What is an equation of state (EoS)? How does it fit into our model of a neutron star?
- How can we use an EoS to make macroscopic predictions about neutron stars?

Outline

- What is a neutron star?
- What is an equation of state (EoS)? How does it fit into our model of a neutron star?
- How can we use an EoS to make macroscopic predictions about neutron stars?
- What is the importance and role of an equation of state in a temporal simulation of a neutron star?

Outline

- What is a neutron star?
- What is an equation of state (EoS)? How does it fit into our model of a neutron star?
- How can we use an EoS to make macroscopic predictions about neutron stars?
- What is the importance and role of an equation of state in a temporal simulation of a neutron star?
- A derivation of an EoS and its predictions:

Outline

- What is a neutron star?
- What is an equation of state (EoS)? How does it fit into our model of a neutron star?
- How can we use an EoS to make macroscopic predictions about neutron stars?
- What is the importance and role of an equation of state in a temporal simulation of a neutron star?
- A derivation of an EoS and its predictions:
  - Quantum Hadrodynamics and the QHD-I parameter set

Introduction

Neutron Stars

Neutron Stars

- Dense core left behind after a supernovae explosion

Neutron Stars

- Dense core left behind after a supernovae explosion
- Made mostly of neutrons, protons, and electrons

Neutron Stars

- Dense core left behind after a supernovae explosion
- Made mostly of neutrons, protons, and electrons; overall, is neutral

Neutron Stars

- Dense core left behind after a supernovae explosion
- Made mostly of neutrons, protons, and electrons; overall, is neutral
- Radius: $\sim 10\,\mathrm{km}$

Neutron Stars

- Dense core left behind after a supernovae explosion
- Made mostly of neutrons, protons, and electrons; overall, is neutral
- Radius: $\sim 10\,\text{km}$; Mass: $\sim 1\odot$.

Neutron Stars

- Dense core left behind after a supernovae explosion
- Made mostly of neutrons, protons, and electrons; overall, is neutral
- Radius: $\sim 10\,\text{km}$; Mass: $\sim 1\,\odot$.
- Approximately the density of atomic nuclei ($\sim 10^{17}\,\text{kg/m}^3$)

Neutron Stars

- Dense core left behind after a supernovae explosion
- Made mostly of neutrons, protons, and electrons; overall, is neutral
- Radius: $\sim 10\,\mathrm{km}$; Mass: $\sim 1\,\odot$.
- Approximately the density of atomic nuclei ($\sim 10^{17}\,\mathrm{kg/m^3}$)
- Core held together by intense gravitational attraction

Neutron Stars

- Dense core left behind after a supernovae explosion
- Made mostly of neutrons, protons, and electrons; overall, is neutral
- Radius: $\sim 10\,\text{km}$; Mass: $\sim 1\odot$.
- Approximately the density of atomic nuclei ($\sim 10^{17}\,\text{kg}/\text{m}^3$)
- Core held together by intense gravitational attraction
  - Gravitational acceleration on Earth's Surface: $\approx 10\,\text{m}/\text{s}^2$

Neutron Stars

- Dense core left behind after a supernovae explosion
- Made mostly of neutrons, protons, and electrons; overall, is neutral
- Radius: $\sim 10\,\mathrm{km}$; Mass: $\sim 1\,\odot$.
- Approximately the density of atomic nuclei ($\sim 10^{17}\,\mathrm{kg/m^3}$)
- Core held together by intense gravitational attraction
  - Gravitational acceleration on Earth's Surface: $\approx 10\,\mathrm{m/s^2}$
  - Neutron star: $\approx 10^{12}\,\mathrm{m/s^2}$

Neutron Stars

- Dense core left behind after a supernovae explosion
- Made mostly of neutrons, protons, and electrons; overall, is neutral
- Radius: $\sim 10\,\mathrm{km}$; Mass: $\sim 1\,\odot$.
- Approximately the density of atomic nuclei ($\sim 10^{17}\,\mathrm{kg/m^3}$)
- Core held together by intense gravitational attraction
  - Gravitational acceleration on Earth's Surface: $\approx 10\,\mathrm{m/s^2}$
  - Neutron star: $\approx 10^{12}\,\mathrm{m/s^2}$ (escape velocity $\sim 100\,000\,\mathrm{km/s} =$

Neutron Stars

- Dense core left behind after a supernovae explosion
- Made mostly of neutrons, protons, and electrons; overall, is neutral
- Radius: $\sim 10\,\mathrm{km}$; Mass: $\sim 1\,\odot$.
- Approximately the density of atomic nuclei ($\sim 10^{17}\,\mathrm{kg/m^3}$)
- Core held together by intense gravitational attraction
  - Gravitational acceleration on Earth's Surface: $\approx 10\,\mathrm{m/s^2}$
  - Neutron star: $\approx 10^{12}\,\mathrm{m/s^2}$ (escape velocity $\sim 100\,000\,\mathrm{km/s} = c/3$)

Neutron Stars

- Dense core left behind after a supernovae explosion
- Made mostly of neutrons, protons, and electrons; overall, is neutral
- Radius: $\sim 10\,\mathrm{km}$; Mass: $\sim 1\,\odot$.
- Approximately the density of atomic nuclei ($\sim 10^{17}\,\mathrm{kg/m^3}$)
- Core held together by intense gravitational attraction
  - Gravitational acceleration on Earth's Surface: $\approx 10\,\mathrm{m/s^2}$
  - Neutron star: $\approx 10^{12}\,\mathrm{m/s^2}$ (escape velocity $\sim 100\,000\,\mathrm{km/s} = c/3$)
- Why are they interesting?

Neutron Stars

- Dense core left behind after a supernovae explosion
- Made mostly of neutrons, protons, and electrons; overall, is neutral
- Radius: $\sim 10\,\text{km}$; Mass: $\sim 1\,\odot$.
- Approximately the density of atomic nuclei ($\sim 10^{17}\,\text{kg}/\text{m}^3$)
- Core held together by intense gravitational attraction
  - Gravitational acceleration on Earth's Surface: $\approx 10\,\text{m}/\text{s}^2$
  - Neutron star: $\approx 10^{12}\,\text{m}/\text{s}^2$ (escape velocity $\sim 100\,000\,\text{km}/\text{s} = c/3$)
- Why are they interesting?
  - Smallest, densest observed stellar objects

Neutron Stars

- Dense core left behind after a supernovae explosion
- Made mostly of neutrons, protons, and electrons; overall, is neutral
- Radius: $\sim 10\,\mathrm{km}$; Mass: $\sim 1\,\odot$.
- Approximately the density of atomic nuclei ($\sim 10^{17}\,\mathrm{kg/m^3}$)
- Core held together by intense gravitational attraction
  - Gravitational acceleration on Earth's Surface: $\approx 10\,\mathrm{m/s^2}$
  - Neutron star: $\approx 10^{12}\,\mathrm{m/s^2}$ (escape velocity $\sim 100\,000\,\mathrm{km/s} = c/3$)
- Why are they interesting?
  - Smallest, densest observed stellar objects
  - Exotic physics

Equation of State (EoS)

What is an equation of state?

Equation of State (EoS)

What is an equation of state?

- A relationship between *energy density* (denoted ϵ)

Equation of State (EoS)

What is an equation of state?

- A relationship between *energy density* (denoted ϵ) and pressure (denoted P)

Equation of State (EoS)

What is an equation of state?

- A relationship between *energy density* (denoted ϵ) and pressure (denoted P)
  - $\epsilon = \epsilon(P)$

Equation of State (EoS)

What is an equation of state?

- A relationship between *energy density* (denoted ϵ) and pressure (denoted P)
  - $\epsilon = \epsilon(P) \quad \Leftrightarrow \quad P = P(\epsilon)$

Equation of State (EoS)

What is an equation of state?

- A relationship between *energy density* (denoted ϵ) and pressure (denoted P)
  - $\epsilon = \epsilon(P) \quad \Leftrightarrow \quad P = P(\epsilon)$
- Encodes the fundamental interparticle interactions within a neutron star

Equation of State (EoS)

What is an equation of state?

- A relationship between *energy density* (denoted ϵ) and pressure (denoted P)
  - $\epsilon = \epsilon(P) \quad \Leftrightarrow \quad P = P(\epsilon)$
- Encodes the fundamental interparticle interactions within a neutron star
- True EoS within a neutron star is unknown;

Equation of State (EoS)

What is an equation of state?

- A relationship between *energy density* (denoted ϵ) and pressure (denoted P)
  - $\epsilon = \epsilon(P) \quad \Leftrightarrow \quad P = P(\epsilon)$
- Encodes the fundamental interparticle interactions within a neutron star
- True EoS within a neutron star is unknown; multitude of candidates, each based on a slightly different model and fit of empirical data

Equation of State (EoS)

What is an equation of state?

- A relationship between *energy density* (denoted ϵ) and pressure (denoted P)
  - $\epsilon = \epsilon(P) \quad \Leftrightarrow \quad P = P(\epsilon)$
- Encodes the fundamental interparticle interactions within a neutron star
- True EoS within a neutron star is unknown; multitude of candidates, each based on a slightly different model and fit of empirical data
  - Models can be very complicated

Equation of State (EoS)

What is an equation of state?

- A relationship between *energy density* (denoted ϵ) and pressure (denoted P)
  - $\epsilon = \epsilon(P) \quad \Leftrightarrow \quad P = P(\epsilon)$
- Encodes the fundamental interparticle interactions within a neutron star
- True EoS within a neutron star is unknown; multitude of candidates, each based on a slightly different model and fit of empirical data
  - Models can be very complicated; often simplifications must be made to be solved practically

Equation of State (EoS)

What is an equation of state?

- A relationship between *energy density* (denoted ϵ) and pressure (denoted P)
  - $\epsilon = \epsilon(P) \quad \Leftrightarrow \quad P = P(\epsilon)$
- Encodes the fundamental interparticle interactions within a neutron star
- True EoS within a neutron star is unknown; multitude of candidates, each based on a slightly different model and fit of empirical data
  - Models can be very complicated; often simplifications must be made to be solved practically
- Often a tabulated list of P and ϵ values;

Equation of State (EoS)

What is an equation of state?

- A relationship between *energy density* (denoted ϵ) and pressure (denoted P)
  - $\epsilon = \epsilon(P) \quad \Leftrightarrow \quad P = P(\epsilon)$
- Encodes the fundamental interparticle interactions within a neutron star
- True EoS within a neutron star is unknown; multitude of candidates, each based on a slightly different model and fit of empirical data
  - Models can be very complicated; often simplifications must be made to be solved practically
- Often a tabulated list of P and ϵ values; however, in simulation work, analytical fits may be required

Using an Equation of State to Make Predictions

Using an EoS to Make Predictions

We want a way to understand the effects of an EoS on the observable properties of a star

Using an EoS to Make Predictions

We want a way to understand the effects of an EoS on the observable properties of a star

- e.g. total mass, total radius

Using an EoS to Make Predictions

We want a way to understand the effects of an EoS on the observable properties of a star

- e.g. total mass, total radius

We create *static solutions*; “images” of neutron star

Using an EoS to Make Predictions

We want a way to understand the effects of an EoS on the observable properties of a star

- e.g. total mass, total radius

We create *static solutions*; “images” of neutron star

- solve the Tolman-Oppenheimer-Volkoff (TOV) equations

Using an EoS to Make Predictions

We want a way to understand the effects of an EoS on the observable properties of a star

- e.g. total mass, total radius

We create *static solutions*; "images" of neutron star

- solve the Tolman-Oppenheimer-Volkoff (TOV) equations
- extract information about maximum max and radius allowed by the EoS

The Tolman-Oppenheimer-Volkoff (TOV) Equations

The Tolman-Oppenheimer-Volkoff (TOV) Equations

Used to describe a static (time independent) spherically symmetric star.

The Tolman-Oppenheimer-Volkoff (TOV) Equations

Used to describe a static (time independent) spherically symmetric star. Given by

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

where ϵ is energy density, P is pressure, and m is “mass.”

The Tolman-Oppenheimer-Volkoff (TOV) Equations

Used to describe a static (time independent) spherically symmetric star. Given by

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

where ϵ is energy density, P is pressure, and m is "mass." Use EoS to determine ϵ

The Tolman-Oppenheimer-Volkoff (TOV) Equations

Used to describe a static (time independent) spherically symmetric star. Given by

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

where ϵ is energy density, P is pressure, and m is "mass." Use EoS to determine $\epsilon = \epsilon(P)$.

The Tolman-Oppenheimer-Volkoff (TOV) Equations

Used to describe a static (time independent) spherically symmetric star. Given by

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

where ϵ is energy density, P is pressure, and m is "mass." Use EoS to determine $\epsilon = \epsilon(P)$.

- Initial conditions:

The Tolman-Oppenheimer-Volkoff (TOV) Equations

Used to describe a static (time independent) spherically symmetric star. Given by

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

where ϵ is energy density, P is pressure, and m is "mass." Use EoS to determine $\epsilon = \epsilon(P)$.

- Initial conditions:

$$m(r = 0) = 0, \quad P(r = 0) \equiv P_0 = \text{const.}$$

The Tolman-Oppenheimer-Volkoff (TOV) Equations

Used to describe a static (time independent) spherically symmetric star. Given by

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

where ϵ is energy density, P is pressure, and m is "mass." Use EoS to determine $\epsilon = \epsilon(P)$.

- Initial conditions:

$$m(r = 0) = 0, \quad P(r = 0) \equiv P_0 = \text{const}.$$

Each solution is uniquely defined by P_0, the *central pressure*.

The Tolman-Oppenheimer-Volkoff (TOV) Equations

Used to describe a static (time independent) spherically symmetric star. Given by

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

where ϵ is energy density, P is pressure, and m is "mass." Use EoS to determine $\epsilon = \epsilon(P)$.

- Initial conditions:

  $$m(r = 0) = 0, \quad P(r = 0) \equiv P_0 = \text{const.}$$

  Each solution is uniquely defined by P_0, the *central pressure*.
- Outer conditions: Let R, M to be the total radius and total mass of the star, respectively.

The Tolman-Oppenheimer-Volkoff (TOV) Equations

Used to describe a static (time independent) spherically symmetric star. Given by

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

where ϵ is energy density, P is pressure, and m is "mass." Use EoS to determine $\epsilon = \epsilon(P)$.

- Initial conditions:

  $$m(r=0) = 0, \quad P(r=0) \equiv P_0 = \text{const}.$$

  Each solution is uniquely defined by P_0, the *central pressure*.
- Outer conditions: Let R, M to be the total radius and total mass of the star, respectively. Defined by:

  $$P(R) = 0, \quad M = m(R).$$

TOV Equations: Computing a Solution

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

TOV Equations: Computing a Solution

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

- Specify a central pressure $P(r = 0) = P_0$

TOV Equations: Computing a Solution

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

- Specify a central pressure $P(r=0) = P_0$
- Begin at very small $r \approx 0$; (10^{-8})

TOV Equations: Computing a Solution

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

- Specify a central pressure $P(r = 0) = P_0$
- Begin at very small $r \approx 0$; (10^{-8})
- Use a numerical integration technique

TOV Equations: Computing a Solution

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

- Specify a central pressure $P(r = 0) = P_0$
- Begin at very small $r \approx 0$; (10^{-8})
- Use a numerical integration technique
  - The Runge-Kutta 4 Algorithm

TOV Equations: Computing a Solution

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

- Specify a central pressure $P(r = 0) = P_0$
- Begin at very small $r \approx 0$; (10^{-8})
- Use a numerical integration technique
  - The Runge-Kutta 4 Algorithm
  - In practice, use Scipy `solve_ivp`; faster due to optimized step size

TOV Equations: Computing a Solution

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

- Specify a central pressure $P(r = 0) = P_0$
- Begin at very small $r \approx 0$; (10^{-8})
- Use a numerical integration technique
  - The Runge-Kutta 4 Algorithm
  - In practice, use Scipy `solve_ivp`; faster due to optimized step size
- Integrate outwards until $P = 0$; use to define R, calculate M

TOV Equations: Computing a Solution

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

- Specify a central pressure $P(r = 0) = P_0$
- Begin at very small $r \approx 0$; (10^{-8})
- Use a numerical integration technique
  - The Runge-Kutta 4 Algorithm
  - In practice, use Scipy `solve_ivp`; faster due to optimized step size
- Integrate outwards until $P = 0$; use to define R, calculate M
- Store curves for $m(r), P(r)$,

TOV Equations: Computing a Solution

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

- Specify a central pressure $P(r = 0) = P_0$
- Begin at very small $r \approx 0$; (10^{-8})
- Use a numerical integration technique
  - The Runge-Kutta 4 Algorithm
  - In practice, use Scipy `solve_ivp`; faster due to optimized step size
- Integrate outwards until $P = 0$; use to define R, calculate M
- Store curves for $m(r), P(r)$, use to calculate $\epsilon(r)$

TOV Equations: Computing a Solution

$$\frac{dm}{dr} = 4\pi r^2 \epsilon, \quad \frac{dP}{dr} = -\frac{(4\pi r^3 P + m)(\epsilon + P)}{r^2(1 - 2m/r)}.$$

- Specify a central pressure $P(r = 0) = P_0$
- Begin at very small $r \approx 0$; (10^{-8})
- Use a numerical integration technique
  - The Runge-Kutta 4 Algorithm
  - In practice, use Scipy `solve_ivp`; faster due to optimized step size
- Integrate outwards until $P = 0$; use to define R, calculate M
- Store curves for $m(r), P(r)$, use to calculate $\epsilon(r)$
- *Note*: works well with tabulated EoSs; an interpolating function is often used

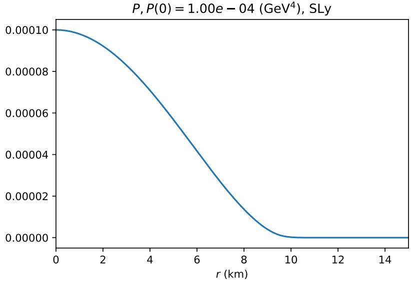
Static Solution: Example

Static Solution: Example

Use an EoS called "SLy" from [3]. A realistic equation of state from an analytical fit of empirical neutron star data.

Static Solution: Example

Use an EoS called "SLy" from [3]. A realistic equation of state from an analytical fit of empirical neutron star data.



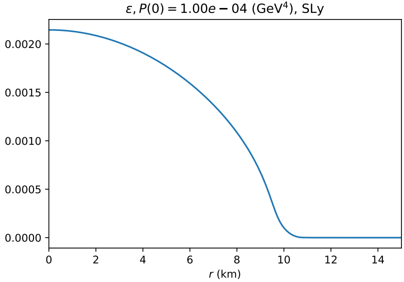


Figure: Example static solution for $P_0 = 10^{-4}\,\text{GeV}^4$ for an EoS called "SLy."

Static Solutions: $M(R)$ and $M(P_0)$ diagrams

Static Solutions: $M(R)$ and $M(P_0)$ diagrams

Single solutions don't tell much about star as a whole; instead, look at trends over lots of solutions

Static Solutions: $M(R)$ and $M(P_0)$ diagrams

Single solutions don't tell much about star as a whole; instead, look at trends over lots of solutions

1. Solve TOV equations for lots of P_0 values:

Static Solutions: $M(R)$ and $M(P_0)$ diagrams

Single solutions don't tell much about star as a whole; instead, look at trends over lots of solutions

1. Solve TOV equations for lots of P_0 values: $P_0 \in [10^{-6}, 10^{-1}](\mathrm{GeV}^4)$.

Static Solutions: $M(R)$ and $M(P_0)$ diagrams

Single solutions don't tell much about star as a whole; instead, look at trends over lots of solutions

1. Solve TOV equations for lots of P_0 values: $P_0 \in [10^{-6}, 10^{-1}](\mathrm{GeV}^4)$.
2. Calculate the total mass M and total radius R for each value of P_0

Static Solutions: $M(R)$ and $M(P_0)$ diagrams

Single solutions don't tell much about star as a whole; instead, look at trends over lots of solutions

1. Solve TOV equations for lots of P_0 values: $P_0 \in [10^{-6}, 10^{-1}](\mathrm{GeV}^4)$.
2. Calculate the total mass M and total radius R for each value of P_0
3. Plot $M(R)$ and $M(P_0)$

Static Solutions: $M(R)$ and $M(P_0)$ diagrams

Single solutions don't tell much about star as a whole; instead, look at trends over lots of solutions

1. Solve TOV equations for lots of P_0 values: $P_0 \in [10^{-6}, 10^{-1}](\text{GeV}^4)$.
2. Calculate the total mass M and total radius R for each value of P_0
3. Plot $M(R)$ and $M(P_0)$

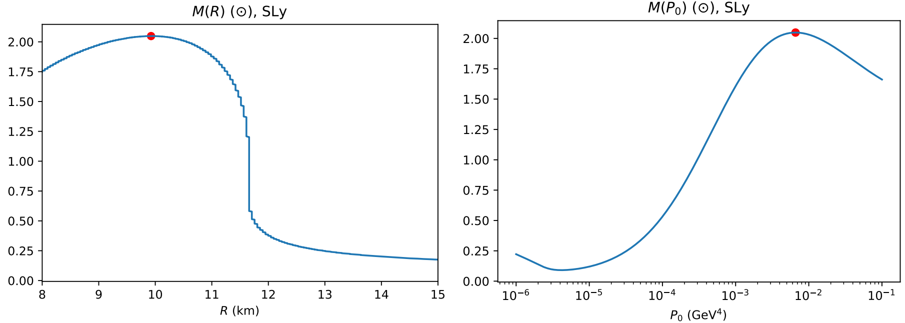
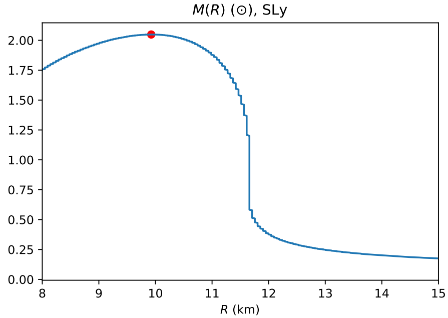
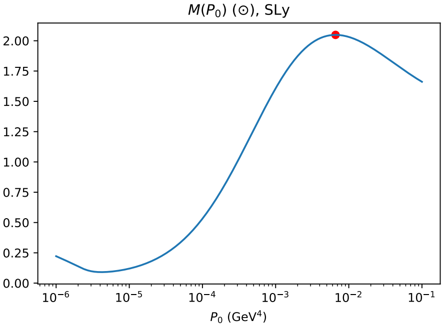


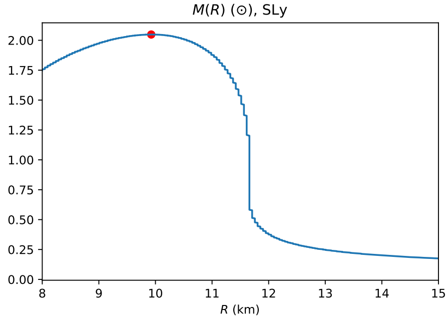
Figure: Example curves for EoS "SLy." $1\odot = 1.989 \times 10^{30}$ kg (solar mass)

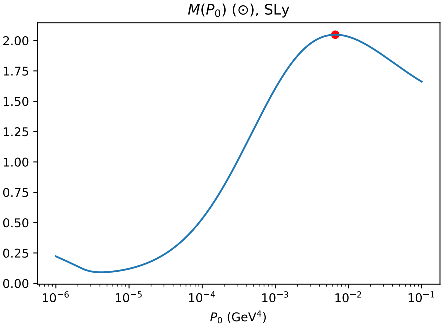
Critical Values of P, R, and M





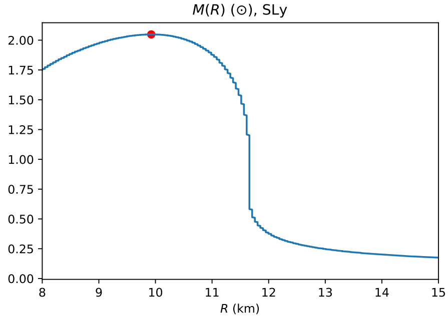
Critical Values of P, R, and M





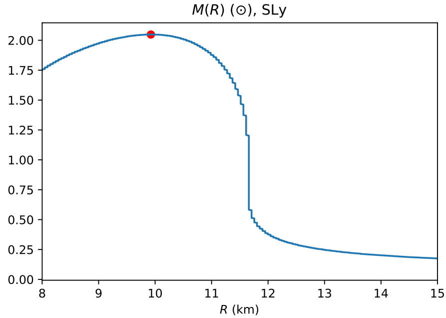
Three important values:

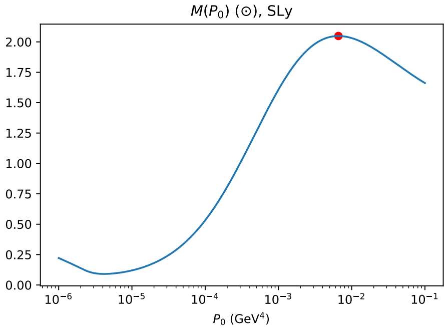
Critical Values of P, R, and M



Three important values: *critical pressure*, *critical mass*, and *critical radius*.

Critical Values of P, R, and M

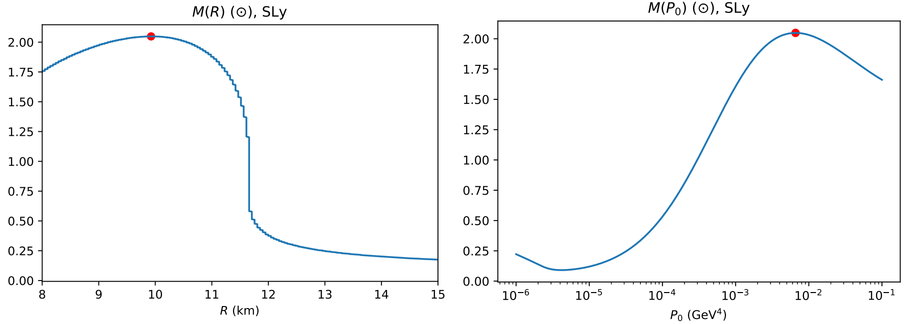




Three important values: *critical pressure*, *critical mass*, and *critical radius*.

- Determined by "peaks" of graph

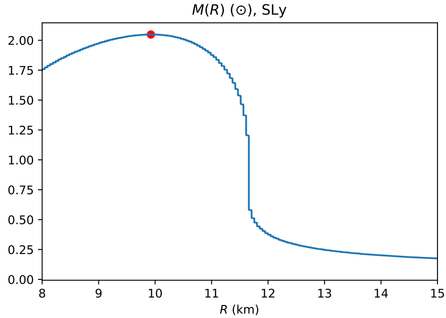
Critical Values of P, R, and M

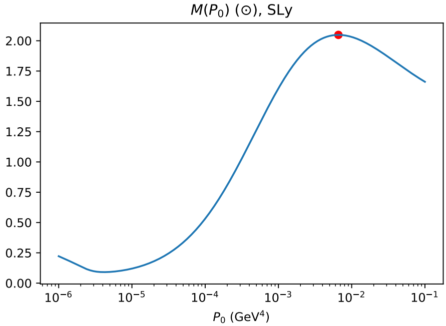


Three important values: *critical pressure*, *critical mass*, and *critical radius*.

- Determined by "peaks" of graph; calculated using an optimization routine

Critical Values of P, R, and M

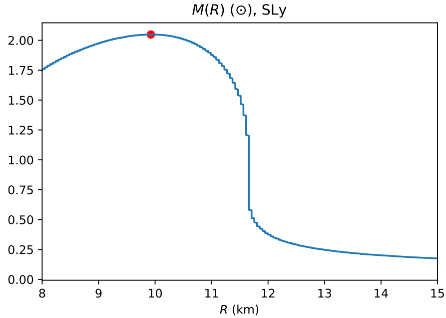


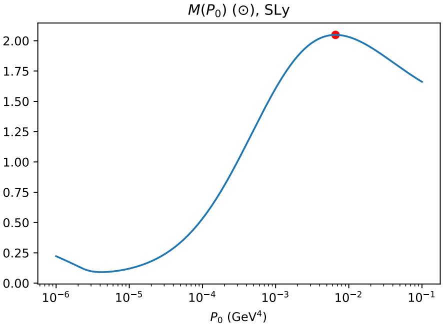


Three important values: *critical pressure*, *critical mass*, and *critical radius*.

- Determined by "peaks" of graph; calculated using an optimization routine
- Maximum mass and radius predicted by EoS

Critical Values of P, R, and M

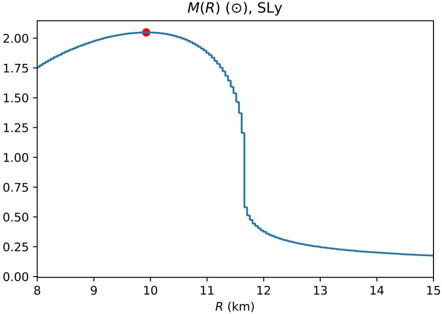


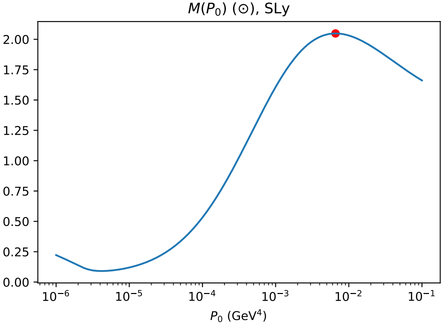


Three important values: *critical pressure*, *critical mass*, and *critical radius*.

- Determined by "peaks" of graph; calculated using an optimization routine
- Maximum mass and radius predicted by EoS
- Largest "stable" pressure

Critical Values of P, R, and M





Three important values: *critical pressure*, *critical mass*, and *critical radius*.

- Determined by "peaks" of graph; calculated using an optimization routine
- Maximum mass and radius predicted by EoS
- Largest "stable" pressure
- SLy: $M_{\max} = 2.05\,\odot$, $R_{\max} = 9.93\,\mathrm{km}$, and $P_{\mathrm{crit}} = 6.59 \times 10^{-3}\,\mathrm{GeV}^4$.

Predictions of Other Equations of State

Predictions of Other Equations of State

There are other realistic, analytical fit EoSs of a similar form SLy given in [3, 5]:

Predictions of Other Equations of State

There are other realistic, analytical fit EoSs of a similar form SLy given in [3, 5]: FPS, BSk19, BSk20, BSk21

Predictions of Other Equations of State

There are other realistic, analytical fit EoSs of a similar form SLy given in [3, 5]: FPS, BSk19, BSk20, BSk21

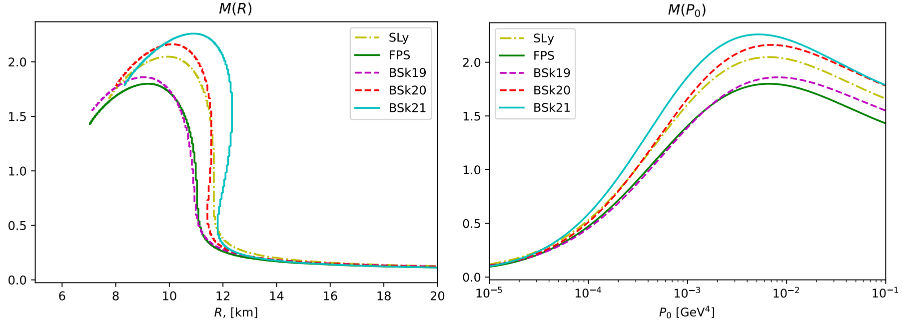


Figure: $M(R)$ and $M(P_0)$ for SLy family EoSs. M in $\odot$.

Predictions of Other Equations of State

There are other realistic, analytical fit EoSs of a similar form SLy given in [3, 5]: FPS, BSk19, BSk20, BSk21

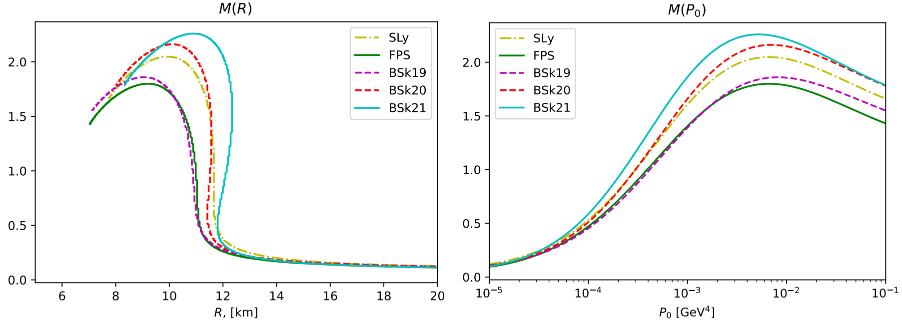


Figure: $M(R)$ and $M(P_0)$ for SLy family EoSs. M in $\odot$.

Critical values of the extreme EoSs:

Predictions of Other Equations of State

There are other realistic, analytical fit EoSs of a similar form SLy given in [3, 5]: FPS, BSk19, BSk20, BSk21

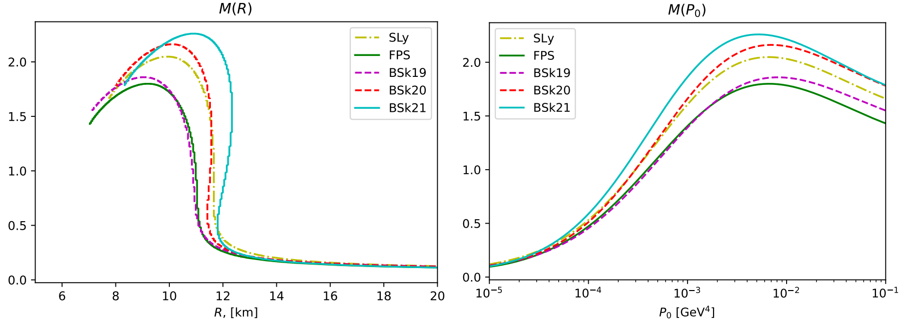


Figure: $M(R)$ and $M(P_0)$ for SLy family EoSs. M in $\odot$.

Critical values of the extreme EoSs:

- FPS: $R_{\text{max}} = 9.20\,\text{km}$, $M_{\text{max}} = 1.80\,\odot$, $P_{\text{crit}} = 6.67 \times 10^{-3}\,\text{GeV}^4$.
- BSk21: $R_{\text{max}} = 10.9\,\text{km}$, $M_{\text{max}} = 2.26\,\odot$, $P_{\text{crit}} = 5.17 \times 10^{-3}\,\text{GeV}^4$.

Temporal Simulations of Neutron Stars

Temporal Simulations of Neutron Stars: Background

Temporal Simulations of Neutron Stars: Background

- Neutron star as a spherically symmetric hydrodynamical system

Temporal Simulations of Neutron Stars: Background

- Neutron star as a spherically symmetric hydrodynamical system
- Three *primitive* variables: pressure P, energy density ϵ, and velocity v

Temporal Simulations of Neutron Stars: Background

- Neutron star as a spherically symmetric hydrodynamical system
- Three *primitive* variables: pressure P, energy density ϵ, and velocity v
- ϵ and P related by an EoS

Temporal Simulations of Neutron Stars: Background

- Neutron star as a spherically symmetric hydrodynamical system
- Three *primitive* variables: pressure P, energy density ϵ, and velocity v
- ϵ and P related by an EoS
- Define *conservative* variables Π, Φ in terms of primitive variables

$$\Pi = \frac{\epsilon + P}{1 - v} - P, \quad \Phi = \frac{\epsilon + P}{1 + v} - P.$$

Temporal Simulations of Neutron Stars: Background

- Neutron star as a spherically symmetric hydrodynamical system
- Three *primitive* variables: pressure P, energy density ϵ, and velocity v
- ϵ and P related by an EoS
- Define *conservative* variables Π, Φ in terms of primitive variables

$$\Pi = \frac{\epsilon + P}{1 - v} - P, \quad \Phi = \frac{\epsilon + P}{1 + v} - P.$$

- Π, Φ obey a conservation equation

$$\partial_t \vec{u} = -\frac{1}{r^2}\partial_r\left(r^2 \frac{\alpha}{a}\vec{f}^{(1)}\right) - \partial_r\left(\frac{\alpha}{a}\vec{f}^{(2)}\right) + \vec{s}, \quad \vec{u} = \begin{bmatrix} \Pi \\ \Phi \end{bmatrix},$$

where a, α are the *gravity* variables.

Temporal Simulations of Neutron Stars: Background

- Neutron star as a spherically symmetric hydrodynamical system
- Three *primitive* variables: pressure P, energy density ϵ, and velocity v
- ϵ and P related by an EoS
- Define *conservative* variables Π, Φ in terms of primitive variables

$$\Pi = \frac{\epsilon + P}{1 - v} - P, \quad \Phi = \frac{\epsilon + P}{1 + v} - P.$$

- Π, Φ obey a conservation equation

$$\partial_t \vec{u} = -\frac{1}{r^2}\partial_r\left(r^2 \frac{\alpha}{a}\vec{f}^{(1)}\right) - \partial_r\left(\frac{\alpha}{a}\vec{f}^{(2)}\right) + \vec{s}, \quad \vec{u} = \begin{bmatrix} \Pi \\ \Phi \end{bmatrix},$$

where a, α are the *gravity* variables.
- $\vec{f}^{(1)} = \vec{f}^{(1)}(\vec{u}, v), \quad \vec{f}^{(2)} = \vec{f}^{(2)}(P), \quad \vec{s} = \vec{s}\,(\vec{u}, P, \epsilon, v, a, \alpha).$

Temporal Simulations of Neutron Stars: Background

- Neutron star as a spherically symmetric hydrodynamical system
- Three *primitive* variables: pressure P, energy density ϵ, and velocity v
- ϵ and P related by an EoS
- Define *conservative* variables Π, Φ in terms of primitive variables

$$\Pi = \frac{\epsilon + P}{1 - v} - P, \quad \Phi = \frac{\epsilon + P}{1 + v} - P.$$

- Π, Φ obey a conservation equation

$$\partial_t \vec{u} = -\frac{1}{r^2}\partial_r\left(r^2 \frac{\alpha}{a}\vec{f}^{(1)}\right) - \partial_r\left(\frac{\alpha}{a}\vec{f}^{(2)}\right) + \vec{s}, \quad \vec{u} = \begin{bmatrix} \Pi \\ \Phi \end{bmatrix},$$

  where a, α are the *gravity* variables.
- $\vec{f}^{(1)} = \vec{f}^{(1)}(\vec{u}, v), \quad \vec{f}^{(2)} = \vec{f}^{(2)}(P), \quad \vec{s} = \vec{s}\,(\vec{u}, P, \epsilon, v, a, \alpha).$
- Separate evolution equations for a, α

Temporal Simulations of Neutron Stars: Background

Temporal Simulations of Neutron Stars: Background

- Evolve a set of discrete spatial gridpoints $\rightarrow$ advanced numerical techniques

Temporal Simulations of Neutron Stars: Background

- Evolve a set of discrete spatial gridpoints $\rightarrow$ advanced numerical techniques
  - Finite differencing (for spatial derivatives) and the method of lines

Temporal Simulations of Neutron Stars: Background

- Evolve a set of discrete spatial gridpoints $\rightarrow$ advanced numerical techniques
  - Finite differencing (for spatial derivatives) and the method of lines
  - High-resolution shock-capturing methods

Temporal Simulations of Neutron Stars: Background

- Evolve a set of discrete spatial gridpoints $\rightarrow$ advanced numerical techniques
  - Finite differencing (for spatial derivatives) and the method of lines
  - High-resolution shock-capturing methods
  - Evolve through time using numerical integration (Runge-Kutta 3, Modified Euler's Method)

Temporal Simulations of Neutron Stars: Background

- Evolve a set of discrete spatial gridpoints $\rightarrow$ advanced numerical techniques
  - Finite differencing (for spatial derivatives) and the method of lines
  - High-resolution shock-capturing methods
  - Evolve through time using numerical integration (Runge-Kutta 3, Modified Euler's Method)
- Use EoSs that are analytical fits for numerical stability and root-finding abilities

Temporal Simulations of Neutron Stars: Background

- Evolve a set of discrete spatial gridpoints $\rightarrow$ advanced numerical techniques
  - Finite differencing (for spatial derivatives) and the method of lines
  - High-resolution shock-capturing methods
  - Evolve through time using numerical integration (Runge-Kutta 3, Modified Euler's Method)
- Use EoSs that are analytical fits for numerical stability and root-finding abilities
  - Determine P, ϵ, v numerically from Π, Φ

Temporal Simulations of Neutron Stars: Background

- Evolve a set of discrete spatial gridpoints $\rightarrow$ advanced numerical techniques
  - Finite differencing (for spatial derivatives) and the method of lines
  - High-resolution shock-capturing methods
  - Evolve through time using numerical integration (Runge-Kutta 3, Modified Euler's Method)
- Use EoSs that are analytical fits for numerical stability and root-finding abilities
  - Determine P, ϵ, v numerically from Π, Φ; need to be able to differentiate (e.g. Newton-Raphson Method)

Temporal Simulations of Neutron Stars: Background

- Evolve a set of discrete spatial gridpoints $\rightarrow$ advanced numerical techniques
  - Finite differencing (for spatial derivatives) and the method of lines
  - High-resolution shock-capturing methods
  - Evolve through time using numerical integration (Runge-Kutta 3, Modified Euler's Method)
- Use EoSs that are analytical fits for numerical stability and root-finding abilities
  - Determine P, ϵ, v numerically from Π, Φ; need to be able to differentiate (e.g. Newton-Raphson Method)
  - Extensive studies of realistic, analytical EoSs from [3, 5]; SLy family

Temporal Simulations of Neutron Stars

Temporal Simulations of Neutron Stars

- Use static solutions as initial data for temporal simulation

Temporal Simulations of Neutron Stars

- Use static solutions as initial data for temporal simulation
  - Above *critical pressure*:

Temporal Simulations of Neutron Stars

- Use static solutions as initial data for temporal simulation
  - Above *critical pressure*: unstable; below: stable

Temporal Simulations of Neutron Stars

- Use static solutions as initial data for temporal simulation
    - Above *critical pressure*: unstable; below: stable
- Stable solutions exhibit *radial oscillations*

Temporal Simulations of Neutron Stars

- Use static solutions as initial data for temporal simulation
  - Above *critical pressure*: unstable; below: stable
- Stable solutions exhibit *radial oscillations*
  - Evolve out to large t and perform a Fourier transform

Temporal Simulations of Neutron Stars

- Use static solutions as initial data for temporal simulation
  - Above *critical pressure*: unstable; below: stable
- Stable solutions exhibit *radial oscillations*
  - Evolve out to large t and perform a Fourier transform

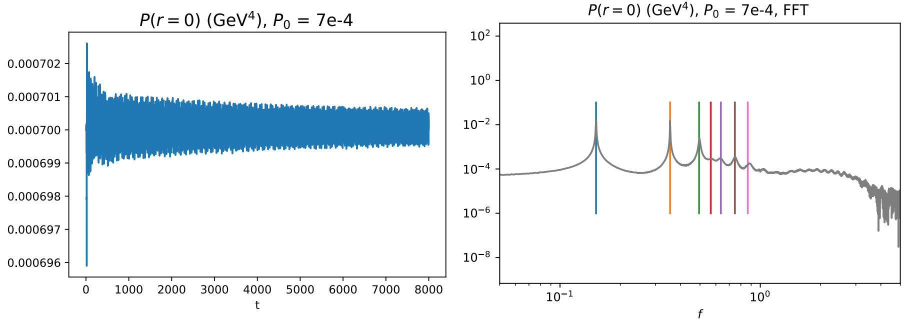


Figure: Plots of $P(r=0)$ for EoS "SLy" and initial $P_0 = 7 \times 10^{-4}\,\text{GeV}^4$. Colored lines on FFT plot represent predicted frequencies.

Temporal Simulations of Neutron Stars

- Use static solutions as initial data for temporal simulation
  - Above *critical pressure*: unstable; below: stable
- Stable solutions exhibit *radial oscillations*
  - Evolve out to large t and perform a Fourier transform

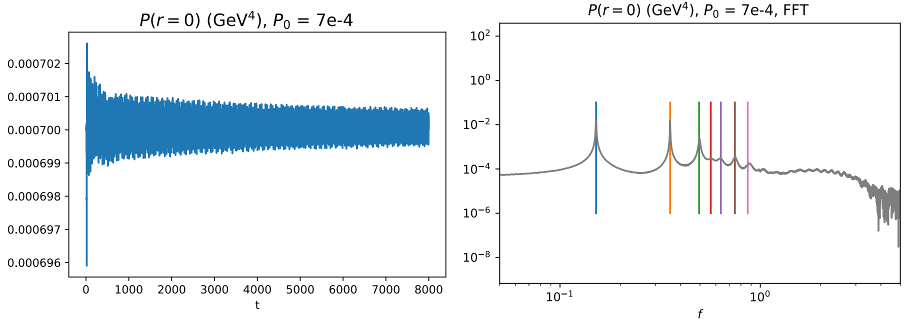


Figure: Plots of $P(r=0)$ for EoS "SLy" and initial $P_0 = 7 \times 10^{-4}\,\mathrm{GeV}^4$. Colored lines on FFT plot represent predicted frequencies.

- Radial oscillations differ by EoS;

Temporal Simulations of Neutron Stars

- Use static solutions as initial data for temporal simulation
  - Above *critical pressure*: unstable; below: stable
- Stable solutions exhibit *radial oscillations*
  - Evolve out to large t and perform a Fourier transform

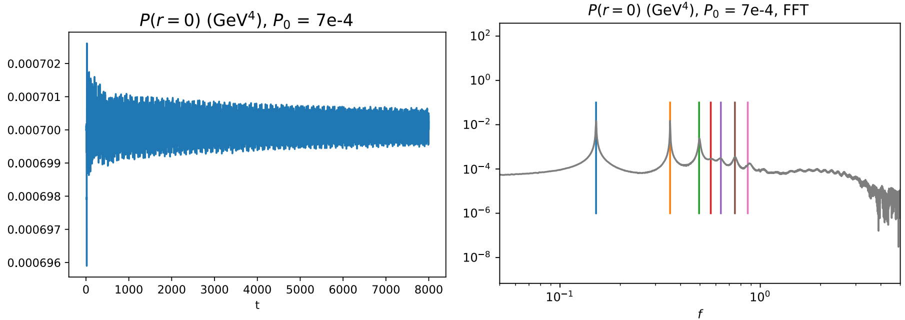


Figure: Plots of $P(r=0)$ for EoS "SLy" and initial $P_0 = 7 \times 10^{-4}\,\mathrm{GeV}^4$. Colored lines on FFT plot represent predicted frequencies.

- Radial oscillations differ by EoS; they could soon be measurable!

Simulations of Neutron Stars: Nearing Publication

"Dynamical evolution of fermion-boson stars with realistic equations of state" under Prof. Ben Kain, College of the Holy Cross

Simulations of Neutron Stars: Nearing Publication

"Dynamical evolution of fermion-boson stars with realistic equations of state" under Prof. Ben Kain, College of the Holy Cross

- Simulate presence of dark matter as a boson; superimpose a complex scalar field

Simulations of Neutron Stars: Nearing Publication

"Dynamical evolution of fermion-boson stars with realistic equations of state" under Prof. Ben Kain, College of the Holy Cross

- Simulate presence of dark matter as a boson; superimpose a complex scalar field
- First temporal simulations of these "mixed" stars using *realistic* equations of state;

Simulations of Neutron Stars: Nearing Publication

"Dynamical evolution of fermion-boson stars with realistic equations of state" under Prof. Ben Kain, College of the Holy Cross

- Simulate presence of dark matter as a boson; superimpose a complex scalar field
- First temporal simulations of these "mixed" stars using *realistic* equations of state; hope is to see how dark matter could affect observable properties of neutron stars

Simulations of Neutron Stars: Nearing Publication

"Dynamical evolution of fermion-boson stars with realistic equations of state" under Prof. Ben Kain, College of the Holy Cross

- Simulate presence of dark matter as a boson; superimpose a complex scalar field
- First temporal simulations of these "mixed" stars using *realistic* equations of state; hope is to see how dark matter could affect observable properties of neutron stars
- Interesting finding: there is a range of unstable static solutions that *migrate* to stable solutions

Derivation and Computation of an Equation of State

Computing an EoS: Quantum Hadrodynamics

Computing an EoS: Quantum Hadrodynamics

A theory of the quantum mechanical, interparticle interactions within a neutron star.

Computing an EoS: Quantum Hadrodynamics

A theory of the quantum mechanical, interparticle interactions within a neutron star.

- Formulation of nuclear interactions between *baryons* by the exchange of *mesons*

Computing an EoS: Quantum Hadrodynamics

A theory of the quantum mechanical, interparticle interactions within a neutron star.

- Formulation of nuclear interactions between *baryons* by the exchange of *mesons*
  - *baryons* are particles containing three quarks (e.g. protons, neutrons)

Computing an EoS: Quantum Hadrodynamics

A theory of the quantum mechanical, interparticle interactions within a neutron star.

- Formulation of nuclear interactions between *baryons* by the exchange of *mesons*
  - *baryons* are particles containing three quarks (e.g. protons, neutrons)
  - *mesons* are quark/anti-quark pairs

Computing an EoS: Quantum Hadrodynamics

A theory of the quantum mechanical, interparticle interactions within a neutron star.

- Formulation of nuclear interactions between *baryons* by the exchange of *mesons*
  - *baryons* are particles containing three quarks (e.g. protons, neutrons)
  - *mesons* are quark/anti-quark pairs
- Requires experimental input for constraint;

Computing an EoS: Quantum Hadrodynamics

A theory of the quantum mechanical, interparticle interactions within a neutron star.

- Formulation of nuclear interactions between *baryons* by the exchange of *mesons*
  - *baryons* are particles containing three quarks (e.g. protons, neutrons)
  - *mesons* are quark/anti-quark pairs
- Requires experimental input for constraint; implemented using *coupling constants*

Computing an EoS: Quantum Hadrodynamics

A theory of the quantum mechanical, interparticle interactions within a neutron star.

- Formulation of nuclear interactions between *baryons* by the exchange of *mesons*
  - *baryons* are particles containing three quarks (e.g. protons, neutrons)
  - *mesons* are quark/anti-quark pairs
- Requires experimental input for constraint; implemented using *coupling constants*
  - Models the strength of the interactions between particles

Computing an EoS: Quantum Hadrodynamics

A theory of the quantum mechanical, interparticle interactions within a neutron star.

- Formulation of nuclear interactions between *baryons* by the exchange of *mesons*
  - *baryons* are particles containing three quarks (e.g. protons, neutrons)
  - *mesons* are quark/anti-quark pairs
- Requires experimental input for constraint; implemented using *coupling constants*
  - Models the strength of the interactions between particles
  - Multiple *parameter sets* have been developed by fitting observed nuclear properties of nuclear matter

Computing an EoS: Quantum Hadrodynamics

A theory of the quantum mechanical, interparticle interactions within a neutron star.

- Formulation of nuclear interactions between *baryons* by the exchange of *mesons*
  - *baryons* are particles containing three quarks (e.g. protons, neutrons)
  - *mesons* are quark/anti-quark pairs
- Requires experimental input for constraint; implemented using *coupling constants*
  - Models the strength of the interactions between particles
  - Multiple *parameter sets* have been developed by fitting observed nuclear properties of nuclear matter
- Considered quite complicated to solve

Computing an EoS: Quantum Hadrodynamics

A theory of the quantum mechanical, interparticle interactions within a neutron star.

- Formulation of nuclear interactions between *baryons* by the exchange of *mesons*
  - *baryons* are particles containing three quarks (e.g. protons, neutrons)
  - *mesons* are quark/anti-quark pairs
- Requires experimental input for constraint; implemented using *coupling constants*
  - Models the strength of the interactions between particles
  - Multiple *parameter sets* have been developed by fitting observed nuclear properties of nuclear matter
- Considered quite complicated to solve; we introduce some simplifications in the QHD-I model

Quantum Hadrodynamics I (QHD-I)

We form the Lagrange Density for QHD-I:

Quantum Hadrodynamics I (QHD-I)

We form the Lagrange Density for QHD-I:

$$\begin{aligned}\mathscr{L} = \bar{\psi}[\gamma_\mu(i\partial^\mu - g_v V^\mu) - (M - g_\phi \phi)]\psi \\ + \frac{1}{2}\left(\partial_\mu \phi \partial^\mu \phi - m_\phi^2 \phi^2\right) - \frac{1}{4} V_{\mu\nu} V^{\mu\nu} + \frac{1}{2} m_v^2 V_\mu V^\mu,\end{aligned}$$

where $V_{\mu\nu} \equiv \partial_\mu V_\nu - \partial_\nu V_\mu$, $\partial_\mu \equiv \partial/\partial x^\mu$. The fields:

- Baryon field (protons and neutrons) $\psi(x^\mu)$, with mass M
- Scalar meson field: $\phi(x^\mu)$, with mass m_ϕ
- Vector meson field: $V^\mu(x^\mu)$, with mass m_v
- Experimental coupling constants: g_v and g_ϕ

Quantum Hadrodynamics I (QHD-I)

We form the Lagrange Density for QHD-I:

$$\mathcal{L} = \bar{\psi}[\gamma_\mu(i\partial^\mu - g_v V^\mu) - (M - g_\phi \phi)]\psi$$
$$+ \frac{1}{2}\left(\partial_\mu \phi \partial^\mu \phi - m_\phi^2 \phi^2\right) - \frac{1}{4} V_{\mu\nu} V^{\mu\nu} + \frac{1}{2} m_v^2 V_\mu V^\mu,$$

where $V_{\mu\nu} \equiv \partial_\mu V_\nu - \partial_\nu V_\mu$, $\partial_\mu \equiv \partial / \partial x^\mu$. The fields:

- Baryon field (protons and neutrons) $\psi(x^\mu)$, with mass M
- Scalar meson field: $\phi(x^\mu)$, with mass m_ϕ
- Vector meson field: $V^\mu(x^\mu)$, with mass m_v
- Experimental coupling constants: g_v and g_ϕ

From $\mathcal{L}$, we can determine ϵ and P, the EoS we desire.

QHD-I: Derivation of Equations of Motion

$$
\begin{aligned}
\mathscr{L} = &\bar{\psi}[\gamma_\mu(i\partial^\mu - g_v V^\mu) - (M - g_\phi \phi)]\psi \\
&+ \frac{1}{2}\left(\partial_\mu \phi \partial^\mu \phi - m_\phi^2 \phi^2\right) - \frac{1}{4} V_{\mu\nu} V^{\mu\nu} + \frac{1}{2} m_v^2 V_\mu V^\mu,
\end{aligned}
$$

QHD-I: Derivation of Equations of Motion

$$
\begin{aligned}
\mathcal{L} = {} & \bar{\psi}[\gamma_\mu(i\partial^\mu - g_v V^\mu) - (M - g_\phi \phi)]\psi \\
& + \frac{1}{2}\left(\partial_\mu \phi \partial^\mu \phi - m_\phi^2 \phi^2\right) - \frac{1}{4} V_{\mu\nu} V^{\mu\nu} + \frac{1}{2} m_v^2 V_\mu V^\mu,
\end{aligned}
$$

Applying the Euler-Lagrange equations for $\mathcal{L}$ over a classical field

$$
\partial_\nu \left(\frac{\partial \mathcal{L}}{\partial(\partial_\nu \varphi_\alpha)} \right) - \frac{\partial \mathcal{L}}{\partial \varphi_\alpha} = 0,
$$

for $\varphi_\alpha \in \{\phi, V^\mu, \psi\}$,

QHD-I: Derivation of Equations of Motion

$$
\begin{aligned}
\mathcal{L} = \bar{\psi}[\gamma_\mu(i\partial^\mu - g_v V^\mu) - (M - g_\phi \phi)]\psi \\
+ \frac{1}{2}\left(\partial_\mu \phi \partial^\mu \phi - m_\phi^2 \phi^2\right) - \frac{1}{4} V_{\mu\nu} V^{\mu\nu} + \frac{1}{2} m_v^2 V_\mu V^\mu,
\end{aligned}
$$

Applying the Euler-Lagrange equations for $\mathcal{L}$ over a classical field

$$
\partial_\nu \left(\frac{\partial \mathcal{L}}{\partial(\partial_\nu \varphi_\alpha)} \right) - \frac{\partial \mathcal{L}}{\partial \varphi_\alpha} = 0,
$$

for $\varphi_\alpha \in \{\phi, V^\mu, \psi\}$, we obtain the equations of motion:

$$
\begin{aligned}
&\partial_\nu \partial^\nu \phi + m_s^2 \phi = g_s \bar{\psi}\psi, \\
&\partial_\mu V^{\mu\nu} + m_\omega^2 V^\nu = g_v \bar{\psi}\gamma^\nu \psi, \\
&[\gamma_\mu(i\partial^\mu - g_v V^\mu) - (M - g_s \phi)]\psi = 0,
\end{aligned}
$$

QHD-I: RMF Simplifications

We introduce the *Relativistic Mean Field* (RMF) simplifications.

QHD-I: RMF Simplifications

We introduce the *Relativistic Mean Field* (RMF) simplifications. We treat the interactions (exchange of mesons) as their average values:

QHD-I: RMF Simplifications

We introduce the *Relativistic Mean Field* (RMF) simplifications. We treat the interactions (exchange of mesons) as their average values:

$$\phi \to \langle \phi \rangle = \phi_0, \quad V_\mu \to \langle V_\mu \rangle = V_0, \quad \bar{\psi}\psi \to \left\langle \bar{\psi}\psi \right\rangle, \quad \bar{\psi}\gamma^\mu\psi \to \left\langle \bar{\psi}\gamma^0\psi \right\rangle,$$

where ϕ_0 and V_0 are constants.

QHD-I: RMF Simplifications

We introduce the *Relativistic Mean Field* (RMF) simplifications. We treat the interactions (exchange of mesons) as their average values:

$$\phi \to \langle \phi \rangle = \phi_0, \quad V_\mu \to \langle V_\mu \rangle = V_0, \quad \bar{\psi}\psi \to \left\langle \bar{\psi}\psi \right\rangle, \quad \bar{\psi}\gamma^\mu\psi \to \left\langle \bar{\psi}\gamma^0\psi \right\rangle,$$

where ϕ_0 and V_0 are constants. This allows us to simplify $\mathscr{L}$ considerably:

QHD-I: RMF Simplifications

We introduce the *Relativistic Mean Field* (RMF) simplifications. We treat the interactions (exchange of mesons) as their average values:

$$\phi \to \langle \phi \rangle = \phi_0, \quad V_\mu \to \langle V_\mu \rangle = V_0, \quad \bar{\psi}\psi \to \left\langle \bar{\psi}\psi \right\rangle, \quad \bar{\psi}\gamma^\mu\psi \to \left\langle \bar{\psi}\gamma^0\psi \right\rangle,$$

where ϕ_0 and V_0 are constants. This allows us to simplify $\mathcal{L}$ considerably:

$$\mathcal{L}_{\text{RMF}} = \bar{\psi}[i\gamma_\mu\partial^\mu - g_v\gamma_0 V_0 - (M - g_s\phi_0)]\psi - \frac{1}{2}m_s^2\phi_0^2 + \frac{1}{2}m_\omega^2 V_0^2,$$

QHD-I: RMF Simplifications

We introduce the *Relativistic Mean Field* (RMF) simplifications. We treat the interactions (exchange of mesons) as their average values:

$$\phi \to \langle \phi \rangle = \phi_0, \quad V_\mu \to \langle V_\mu \rangle = V_0, \quad \bar{\psi}\psi \to \left\langle \bar{\psi}\psi \right\rangle, \quad \bar{\psi}\gamma^\mu\psi \to \left\langle \bar{\psi}\gamma^0\psi \right\rangle,$$

where ϕ_0 and V_0 are constants. This allows us to simplify $\mathcal{L}$ considerably:

$$\mathcal{L}_{\text{RMF}} = \bar{\psi}[i\gamma_\mu\partial^\mu - g_v\gamma_0 V_0 - (M - g_s\phi_0)]\psi - \frac{1}{2}m_s^2\phi_0^2 + \frac{1}{2}m_\omega^2 V_0^2,$$

Applying the same simplifications to the equations of motions gives:

$$\begin{aligned}
&m_s^2\phi_0^2 = g_s \left\langle \bar{\psi}\psi \right\rangle \\
&m_\omega^2 V_0 = g_v \left\langle \bar{\psi}\gamma^0\psi \right\rangle \\
&[i\gamma_\mu\partial^\mu - g_v\gamma_0 V_0 - (M - g_s\phi_0)]\psi = 0
\end{aligned}$$

QHD-I: Solving for ϵ and P

QHD-I: Solving for ϵ and P

We can now solve for ϵ and P.

QHD-I: Solving for ϵ and P

We can now solve for ϵ and P. From [1], we have

$$\epsilon = \left\langle T^{00} \right\rangle, \quad P = \frac{1}{3} \left\langle T^{ii} \right\rangle,$$

where $T^{\mu\nu}$ is the energy momentum tensor, given by

$$T^{\mu\nu} = \frac{\partial \mathcal{L}}{\partial(\partial_\mu \varphi_\alpha)} \partial^\nu \varphi_\alpha - \mathcal{L} \eta^{\mu\nu}.$$

QHD-I: Solving for ϵ and P

We can now solve for ϵ and P. From [1], we have

$$\epsilon = \left\langle T^{00} \right\rangle, \quad P = \frac{1}{3} \left\langle T^{ii} \right\rangle,$$

where $T^{\mu\nu}$ is the energy momentum tensor, given by

$$T^{\mu\nu} = \frac{\partial \mathcal{L}}{\partial(\partial_\mu \varphi_\alpha)} \partial^\nu \varphi_\alpha - \mathcal{L} \eta^{\mu\nu}.$$

Using $\mathcal{L}_{\text{RMF}}$, we obtain

$$T^{\mu\nu}_{\text{RMF}} = i\bar{\psi}\gamma^\mu \partial^\nu \psi - \eta^{\mu\nu}\left(-\frac{1}{2} m_s^2 \phi_0^2 + \frac{1}{2} m_\omega^2 V_0^2\right).$$

QHD-I: Solving for ϵ and P

We can now solve for ϵ and P. From [1], we have

$$\epsilon = \left\langle T^{00} \right\rangle, \quad P = \frac{1}{3} \left\langle T^{ii} \right\rangle,$$

where $T^{\mu\nu}$ is the energy momentum tensor, given by

$$T^{\mu\nu} = \frac{\partial \mathcal{L}}{\partial(\partial_\mu \varphi_\alpha)} \partial^\nu \varphi_\alpha - \mathcal{L} \eta^{\mu\nu}.$$

Using $\mathcal{L}_{\text{RMF}}$, we obtain

$$T^{\mu\nu}_{\text{RMF}} = i\bar{\psi}\gamma^\mu \partial^\nu \psi - \eta^{\mu\nu} \left(-\frac{1}{2} m_s^2 \phi_0^2 + \frac{1}{2} m_\omega^2 V_0^2 \right).$$

This gives

$$\epsilon = \left\langle i\bar{\psi}\gamma^0 \partial^0 \psi \right\rangle + \frac{1}{2} m_s^2 \phi_0^2 - \frac{1}{2} m_\omega^2 V_0^2,$$

$$P = \left\langle i\bar{\psi}\gamma^i \partial^i \psi \right\rangle - \frac{1}{2} m_s^2 \phi_0^2 + \frac{1}{2} m_\omega^2 V_0^2.$$

QHD-I: Solving for ϵ and P

We can now solve for ϵ and P. From [1], we have

$$\epsilon = \left\langle T^{00} \right\rangle, \quad P = \frac{1}{3} \left\langle T^{ii} \right\rangle,$$

where $T^{\mu\nu}$ is the energy momentum tensor, given by

$$T^{\mu\nu} = \frac{\partial \mathcal{L}}{\partial(\partial_\mu \varphi_\alpha)} \partial^\nu \varphi_\alpha - \mathcal{L} \eta^{\mu\nu}.$$

Using $\mathcal{L}_{\text{RMF}}$, we obtain

$$T^{\mu\nu}_{\text{RMF}} = i\bar{\psi}\gamma^\mu \partial^\nu \psi - \eta^{\mu\nu} \left(-\frac{1}{2} m_s^2 \phi_0^2 + \frac{1}{2} m_\omega^2 V_0^2 \right).$$

This gives

$$\epsilon = \left\langle i\bar{\psi}\gamma^0 \partial^0 \psi \right\rangle + \frac{1}{2} m_s^2 \phi_0^2 - \frac{1}{2} m_\omega^2 V_0^2,$$

$$P = \left\langle i\bar{\psi}\gamma^i \partial^i \psi \right\rangle - \frac{1}{2} m_s^2 \phi_0^2 + \frac{1}{2} m_\omega^2 V_0^2.$$

The above expectation values are non-trivial and are derived in [1].

QHD-I: Resulting Equations

QHD-I: Resulting Equations

After calculation the expectation values, we obtain the following equations:

QHD-I: Resulting Equations

After calculation the expectation values, we obtain the following equations:

$$\phi_0 = \frac{g_\phi}{m_\phi^2}\frac{1}{\pi^2}\int_0^{k_f} dk\, \frac{(M - g_\phi\phi_0)k^2}{\sqrt{k^2 + (M - g_\phi\phi_0)}},$$

$$V_0 = \frac{g_v}{m_v^2}\frac{k_f^3}{3\pi^2},$$

$$\epsilon = \frac{1}{2}m_\phi^2\phi_0^2 + \frac{1}{2}m_v^2 V_0^2 + \frac{1}{\pi^2}\int_0^{k_f} dk\, k^2\sqrt{k^2 + m^{*2}},$$

$$P = -\frac{1}{2}m_\phi^2\phi_0^2 + \frac{1}{2}m_v^2 V_0^2 + \frac{1}{3}\left(\frac{1}{\pi^2}\int_0^{k_f} dk\, \frac{k^4}{\sqrt{k^2 + m^{*2}}}\right).$$

where $m^* = (M - g_\phi\phi)$, the *reduced mass*.

QHD-I: Resulting Equations

After calculation the expectation values, we obtain the following equations:

$$\phi_0 = \frac{g_\phi}{m_\phi^2}\frac{1}{\pi^2}\int_0^{k_f} dk\, \frac{(M - g_\phi\phi_0)k^2}{\sqrt{k^2 + (M - g_\phi\phi_0)}},$$

$$V_0 = \frac{g_v}{m_v^2}\frac{k_f^3}{3\pi^2},$$

$$\epsilon = \frac{1}{2}m_\phi^2\phi_0^2 + \frac{1}{2}m_v^2 V_0^2 + \frac{1}{\pi^2}\int_0^{k_f} dk\, k^2\sqrt{k^2 + m^{*2}},$$

$$P = -\frac{1}{2}m_\phi^2\phi_0^2 + \frac{1}{2}m_v^2 V_0^2 + \frac{1}{3}\left(\frac{1}{\pi^2}\int_0^{k_f} dk\, \frac{k^4}{\sqrt{k^2 + m^{*2}}}\right).$$

where $m^* = (M - g_\phi\phi)$, the *reduced mass*. k_f, the Fermi wavenumber, is a free parameter.

Resulting Equations

Resulting Equations

Goal:

Resulting Equations

Goal: create a list of values that show us $\epsilon(P)$;

Resulting Equations

Goal: create a list of values that show us $\epsilon(P)$; each value of k_f gives us a different ϵ and P.

Resulting Equations

Goal: create a list of values that show us $\epsilon(P)$; each value of k_f gives us a different ϵ and P.

To produce the EoS:

Resulting Equations

Goal: create a list of values that show us $\epsilon(P)$; each value of k_f gives us a different ϵ and P.

To produce the EoS: (repeat the following)

Resulting Equations

Goal: create a list of values that show us $\epsilon(P)$; each value of k_f gives us a different ϵ and P.

To produce the EoS: (repeat the following)

- Choose a k_f value

Resulting Equations

Goal: create a list of values that show us $\epsilon(P)$; each value of k_f gives us a different ϵ and P.

To produce the EoS: (repeat the following)

- Choose a k_f value
- calculate ϕ_0 and V_0

Resulting Equations

Goal: create a list of values that show us $\epsilon(P)$; each value of k_f gives us a different ϵ and P.

To produce the EoS: (repeat the following)

- Choose a k_f value
- calculate ϕ_0 and V_0; use *rootfinding* for ϕ_0

Resulting Equations

Goal: create a list of values that show us $\epsilon(P)$; each value of k_f gives us a different ϵ and P.

To produce the EoS: (repeat the following)

- Choose a k_f value
- calculate ϕ_0 and V_0; use *rootfinding* for ϕ_0
- Using those values, calculate P and ϵ and store in a table

Resulting Equations

Goal: create a list of values that show us $\epsilon(P)$; each value of k_f gives us a different ϵ and P.

To produce the EoS: (repeat the following)

- Choose a k_f value
- calculate ϕ_0 and V_0; use *rootfinding* for ϕ_0
- Using those values, calculate P and ϵ and store in a table

We loop through k_f values until we have a large range of P values

$$P \in [10^{-20}, 10^{-1}](\text{GeV}^4).$$

$M(R)$ and $M(P_0)$ Curves for QHD-I

We use the tabulated values of P and ϵ to solve the TOV equations:

$M(R)$ and $M(P_0)$ Curves for QHD-I

We use the tabulated values of P and ϵ to solve the TOV equations:

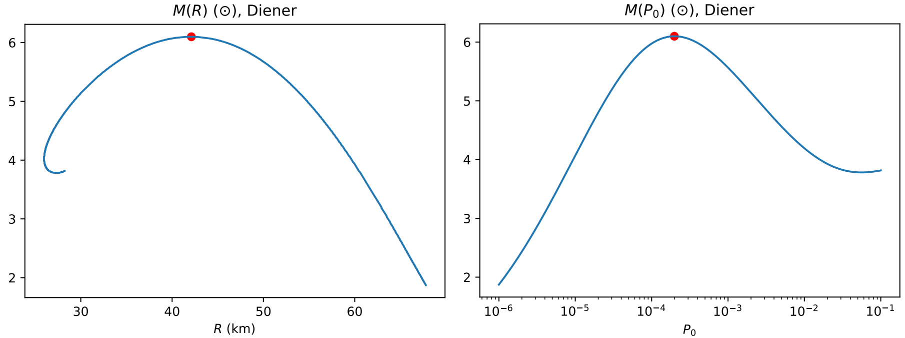


Figure: $M(R)$ and $M(P_0)$ curves for QHD-I EoS.

$M(R)$ and $M(P_0)$ Curves for QHD-I

We use the tabulated values of P and ϵ to solve the TOV equations:

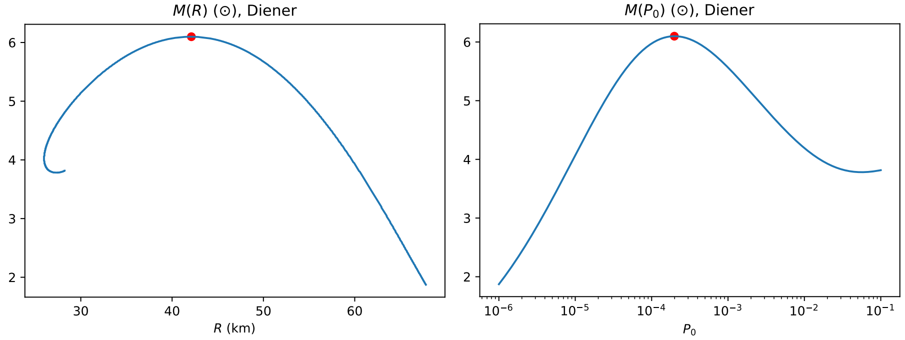


Figure: $M(R)$ and $M(P_0)$ curves for QHD-I EoS.

These curves give

$$M_{\max} = 6.1\,\odot, \quad R_{\max} = 42.1\,\text{km}, \quad P_{\text{crit}} = 1.98 \times 10^{-4}\,\text{GeV}^4.$$

Conclusion

- An equation of state is a relationship between energy density and pressure within a neutron star

Conclusion

- An equation of state is a relationship between energy density and pressure within a neutron star
- We use the TOV equations to predict the maximum mass and radius that a given EoS will produce

Conclusion

- An equation of state is a relationship between energy density and pressure within a neutron star
- We use the TOV equations to predict the maximum mass and radius that a given EoS will produce
- Within a temporal simulation of a neutron star:

Conclusion

- An equation of state is a relationship between energy density and pressure within a neutron star
- We use the TOV equations to predict the maximum mass and radius that a given EoS will produce
- Within a temporal simulation of a neutron star:
  - Static solutions are used as initial data

Conclusion

- An equation of state is a relationship between energy density and pressure within a neutron star
- We use the TOV equations to predict the maximum mass and radius that a given EoS will produce
- Within a temporal simulation of a neutron star:
  - Static solutions are used as initial data
  - We can predict radial oscillation frequencies of neutron stars, which could soon be measurable

Conclusion

- An equation of state is a relationship between energy density and pressure within a neutron star
- We use the TOV equations to predict the maximum mass and radius that a given EoS will produce
- Within a temporal simulation of a neutron star:
  - Static solutions are used as initial data
  - We can predict radial oscillation frequencies of neutron stars, which could soon be measurable
- We use the QHD-I parameter set and RMF simplifications to solve a system of equations and generate an equation of state

References I

Jacobus Petrus William Diener. “Relativistic mean-field theory applied to the study of neutron star properties”. PhD thesis. Stellenbosch University, 2008.

Norman K. Glendenning. *Compact Stars*. Springer, 1997.

P. Haensel and A. Y. Potekhin. “Analytical representations of unified equations of state of neutron-star matter”. In: *Astronomy & Astrophysics* 428.1 (Nov. 2004), pp. 191–197. ISSN: 1432-0746. DOI: 10.1051/0004-6361:20041722. URL: http://dx.doi.org/10.1051/0004-6361:20041722.

“Neutron Stars”. In: *COSMOS - The SAO Encyclopedia of Astronomy*. URL: https://astronomy.swin.edu.au/cosmos/n/neutron+star#:~:text=Neutrons%20stars%20are%20extreme%20objects,weigh%20around%20a%20billion%20tonnes..

References II

A. Y. Potekhin et al. "Analytical representations of unified equations of state for neutron-star matter". In: *Astronomy & Astrophysics* 560 (Dec. 2013), A48. ISSN: 1432-0746. DOI: 10.1051/0004-6361/201321697. URL: http://dx.doi.org/10.1051/0004-6361/201321697.