

Constraints on the Nuclear Equation of State induced by Binary Neutron Star Mergers

Zachariadou Anastasia

BSc Thesis Presentation, 2023



Physics Department, Aristotle University of Thessaloniki

Κεφάλαια

- Εισαγωγή: Μια σύντομη επισκόπηση των Αστέρων Νετρονίων
- Μαθηματικό Έποβαθρο
- Αποτελέσματα και Σχόλια
- Συμπεράσματα

Εισαγωγή: Μία σύντομη επισκόπηση των Αστέρων Νετρονίων

Ταυτότητα

Οι Αστέρες Νετρονίων γεννιούνται από τις εκρήξεις αστεριών μεγάλης μάζας ($8 - 25M_{\odot}$) όταν αυτά έχουν φτάσει στο τέλος της ζωής τους.

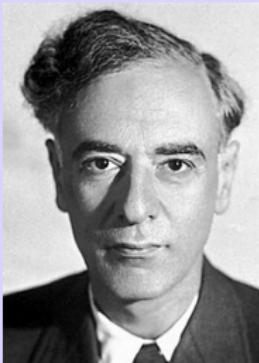
Είναι εξαιρετικά **πυκνά** αστρικά αντικείμενα, με **μικρές Ακτίνες**, $R = 10 - 12 \text{ km}$ και συγκριτικά **μεγάλες Μάζες** $1.1 - 2M_{\odot}$. Επίσης, χαρακτηρίζονται από πολύ ισχυρά μαγνητικά πεδία (εως και $B = 10^{12} - 10^{15} \text{ G}$) και πολυ υψηλή περιστροφική ταχύτητα.



Ιστορική Αναδρομή

- 1930: Ξεκινά η θεωρητική ενασχόληση με το αντικείμενο.
- 1932: Ο Landau ασχολείται με τη μέγιστη μάζα συμπυκνωμένων αστέρων και ο Chadwick ανακαλύπτει το νετρόνιο.
- 1933: Πρώτη εμφάνιση του όρου 'Αστέρας Νετρονίων' σε δημοσείευση των Baade και Zwicky.
- 1939: Ανάπτυξη των εξισώσεων TOV από Oppenheimer και Volkoff. Οδηγούν σε μια πρώτη εκτίμηση της μέγιστης μάζας.
- 1967: Ανακάλυψη των pulsars από τη Jocelyn Bell
- **Σήμερα:** Η ανίχνευση βαρυτικών κυμάτων ανοίγει νέους δρόμους για τη μελέτη Αστέρων Νετρονίων σε διαδικασίες συστήματα.

Ιστορική Αναδρομή



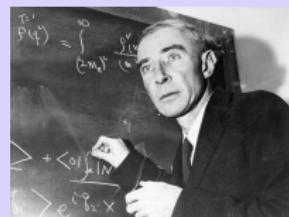
Lev Landau



Fritz Zwicky



Jocelyn Bell Burnell



J. Robert Oppenheimer

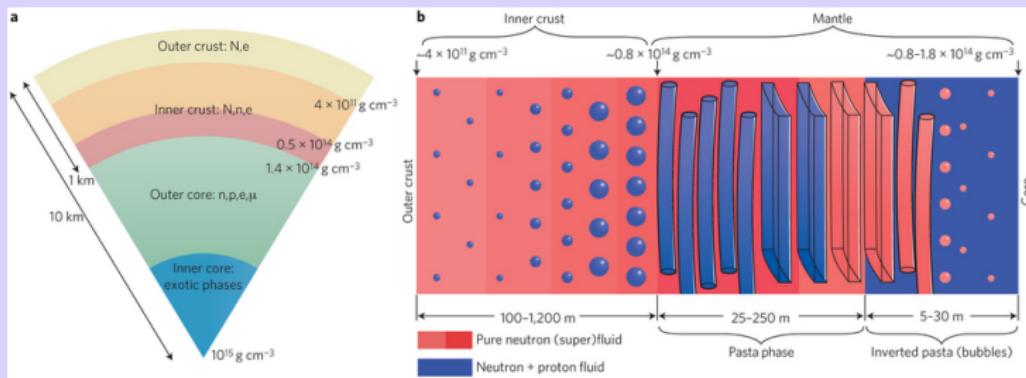
Δομή

- **Ατμόσφαιρα:** Το εξώτερο στρώμα του αστέρα, αποτελούμενο από υπέρλεπτή στρώση πλάσματος. Καθορίζει τις ακτινοβολιακές ιδιότητες του αστέρα (η/μ φάσμα ακτινοβολίας, θερμικά χαρακτηριστικά). Επίσης, παρέχει πληροφορίες για την επιφάνεια (θερμοκρασία, βαρύτητα, σύνθεση και μαγνητικό πεδίο). Το πάχος της διαφοροποιείται ανάλογα με τη θερμοκρασία.
- **Εξώτερη Κρούστα:** Ακριβώς κάτω από την ατμόσφαιρα. Αποτελείται από ιόντα και e. Με την αύξηση του βάθους, τα e γίνονται ρελατιβιστικά και οδηγούμαστε σε πλήρη ιονισμό των ατόμων και στον σχηματισμό ενός ισχυρά συσχετισμένου συστήματος Coulomb, το οποίο μπορεί να στερεοποιηθεί. Στην κρίσιμη τιμή πυκνότητας $4 \times 10^{11} g^{-3}$ τα νετρόνια αρχίζουν να 'δραπετεύουν', δημιουργώντας τελικά ένα ελεύθερο νετρονικό αέριο (neutron drip).

Δομή

- **Εσώτερη Κρούστα:** Έχει περίπου 1km έκταση και περιλαμβάνει ε, ελεύθερα n και πυρήνες ατόμων πλούσιους σε n . Με αύξηση της πυκνότητας αυξάνεται και το ποσοστό των ελεύθερων n το οποίο προκαλεί "softening" της Κ.Ε. Πιο χαμηλά, δημιουργείται 'μανδύας' και στα όρια με τον πυρήνα μένουν μόνο n και νουκλεόνια σε υπέρρευστη κατάσταση (;).
- **Εξώτερος Πυρήνας:** Εύρος πυκνοτήτων $0.5\rho_0 - 2\rho_0$, ($\rho_0 = 2.8 \times 10^{14} \text{g}^{-3}$) και έκταση αρκετών km . Έχει τη λεγόμενη πρεμ σύσταση. Κι εδώ μπορεί να υπάρχουν υπερρευστές καταστάσεις.
- **Εσωτερικός Πυρήνας:** $\rho > 2\rho_0$ (ίσως και $10 - 15\rho_0$). Σύνθεση και Κ.Ε υπό έρευνα: εξωτικά σωματίδια όπως υπερόνια (Σ, Λ), πιονικά ή καονικά συμπυκνώματα, ελεύθερη ύλη κουαρκ, μικτή ύλη. Αυτές οι υποθέσεις μπορεί να προκαλέσουν μεταβατικές φάσεις, επηρεάζοντας την Κ.Ε και τη συμπεριφορά του αστέρα.

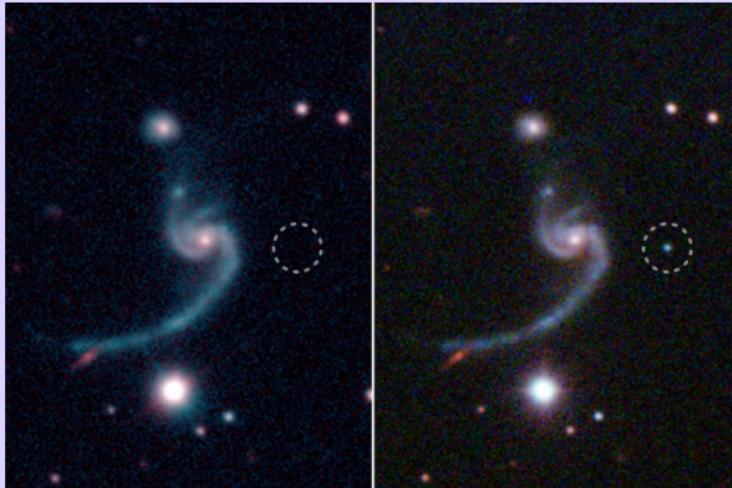
Δομή



Δομή Αστέρων Νετρονίων

Διαδικά Συστήματα Αστέρων Νετρονίων

Είναι σύνηθες φαινόμενο οι Αστέρες Νετρονίων να βρίσκονται σε Διαδικά Συστήματα (Binary Neutron Star Systems). Αυτά τα συστήματα αποτελούνται από δύο κοντινά αστέρια που περιστρέφονται σε τροχιά γύρω από κοινό κέντρο μάζας.



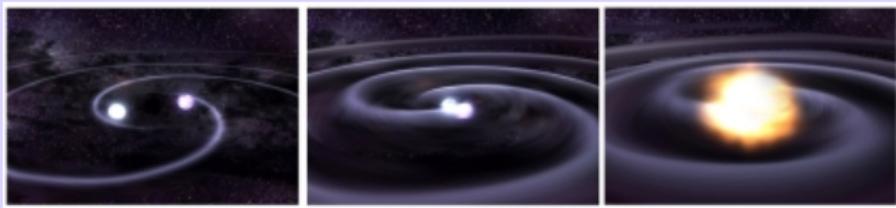
Γέννηση Διαδικού Συστήματος Αστέρων Νετρονίων

Συγχώνευση Αστέρων Νετρονίων Διαδικού Συστήματος

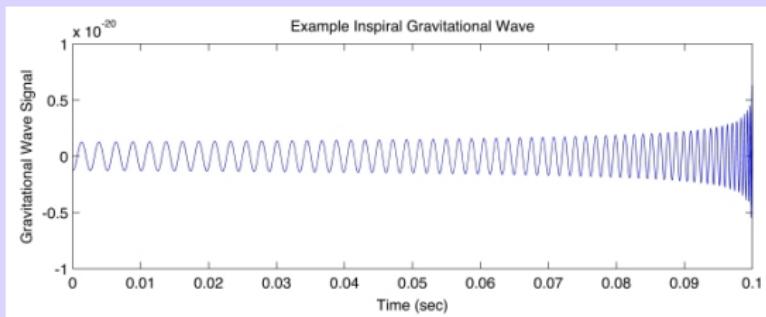
Η διαδικασία της συγχώνευσης εκτελείται σε 3 στάδια:

- ① **Inspiral:** Οι αστέρες περιστρέφονται ο ένας γύρω από τον άλλο με την απόστασή τους σταδιακά να μειώνεται και την ταχύτητα να αυξάνεται. Κατά τη φάση αυτή παράγονται βαρυτικά κύματα με μέγιστο συχνότητας στη σύγκρουση.
- ② **Συγχώνευση (Merger):** Η στιγμή της σύγκρουσης. Οι παλίρροιακές παράμετροι γίνονται σημαντικές. Το αποτέλεσμα της συγχώνευσης εξαρτάται από τα χαρακτηριστικά των αρχικών αστέρων
- ③ **Μετά τη Συγχώνευση:**
 - Γηραιότερες Αστέρι Νετρονίων
 - Μελανή Οπή
 - Σύστημα Μελανής Οπής-Τόρου

Συγχώνευση Αστέρων Νετρονίων Διαδικού Συστήματος



Στάδια BNS Merger



Σήμα από βαρυτικά κύματα κατά το inspiral

GW170817

Το **GW170817** πρόκειται για σήμα βαρυτικών κυμάτων που εντοπίστηκε από τους ανιχνευτές LIGO και Virgo στις 17 Αυγούστου 2017.

Αιτία του σήματος: Η συγχώνευση δύο αστέρων νετρονίων διαδικού συστήματος μαζών $1.36M_{\odot}$ και $1.27M_{\odot}$.

Τα βαρυτικά κύματα αποτελούν σημαντικό εργαλείο μελέτης των Αστέρων Νετρονίων. Άμεσα παρατηρήσιμο μέγεθος αποτελεί η **παλιρροιακή παραμόρφωση** λ που ποσοτικοποιείται μέσω του **tidal Love number** k_2 . Δίνει πληροφορίες για τη δομή των αστέρων, την Κ.Ε., αλλά και το **μέγεθός** τους.

Στη μελέτη μας χρησιμοποιήθηκαν δεδομένα του LIGO που αντλήθηκαν από το σήμα GW170817 με σκοπό τη σύγκριση με τα δικά μας αποτελέσματα.

Μαθηματικό Υπόβαθρο

Εξισώσεις ΤΟV

- Σύστημα διαφορικών εξισώσεων του οποίου η λύση δίνει πληροφορίες για τη δομή του αστέρα
- $G = c = 1$

Τδροστατική Ισορροπία:

$$\frac{d\bar{P}(r)}{dr} = -1.474 \frac{\bar{\mathcal{E}}(r)\bar{M}(r)}{r^2} \left(1 + \frac{\bar{P}(r)}{\bar{\mathcal{E}}(r)}\right) \left(1 + 11.2 \times 10^{-6} r^3 \frac{\bar{P}(r)}{\bar{M}(r)}\right) \\ \times \left(1 - 2.948 \frac{\bar{M}(r)}{r}\right)^{-1}$$

Σύνδεση Μάζας-Ακτίνας:

$$\frac{d\bar{M}(r)}{dr} = 11.2 \times 10^{-6} r^2 \bar{\mathcal{E}}(r),$$

Εξισώσεις TOV

Μετατροπές:

$$M(r) = \bar{M}(r) M_{\odot},$$

$$\mathcal{E}(r) = \bar{\mathcal{E}}(r) \mathcal{E}_0,$$

$$P(r) = \bar{P}(r) \mathcal{E}_0$$

$$\mathcal{E}_0 = 1 \text{ MeV} \cdot \text{fm}^{-3},$$

$$\frac{GM_{\odot}}{c^2} = 1.474 \text{ Km},$$

$$\frac{4\pi}{M_{\odot} c^2} = 0.7 \cdot 10^{-40} \frac{\text{s}^2}{\text{Kg} \cdot \text{Km}^2}.$$

Μάζα:

$$\bar{M}(R) = 11.2 \times 10^{-6} \int_0^R r^2 \bar{\mathcal{E}}(r) dr = b_0 \int \bar{\mathcal{E}}(r) dr.$$

Καταστατικές Εξισώσεις

- $T = 0K$
- Μορφή Καταστατικών Εξισώσεων: $\mathcal{E}(P)$
 - ① Stiff K.E: Έντονη άυξηση της πίεσης με την άυξηση της πυκνότητας
 - ② Soft K.E: Πιο σταδιακή άυξηση της πίεσης με την άυξηση της πυκνότητας
- P units: $[MeV \cdot fm^{-3}]$

Π.χ:

$$\text{MDI-1} : E(P) = 15.55P^{0.666} + 76.71P^{0.247}$$

$$\text{HLPS-2} : E(P) = 172.858 \left(1 - e^{-\frac{P}{22.8644}}\right) + 2777.75 \left(1 - e^{-\frac{P}{1909.97}}\right) + 161.553$$

$$\text{Ska} : E(P) = 0.53928P^{1.01394} + 94.31452P^{0.35135}$$

Καταστατικές Εξισώσεις

- Πρέπει να ληφθεί υπόψιν και η Κρούστα

$$P \in (9.34375 \cdot 10^{-5}, 0.184] :$$

$$\mathcal{E}(P) = 0.00873 + 103.17338 \left(1 - e^{\frac{-P}{0.38527}}\right) + 7.34979 \left(1 - e^{\frac{-P}{0.01211}}\right)$$

$$P \in (4.1725 \cdot 10^{-8}, 9.34375 \cdot 10^{-5}] :$$

$$\mathcal{E}(P) = 0.00015 + 0.00203 \left(1 - e^{-P \cdot 344827.5}\right) + 0.10851 \left(1 - e^{-P \cdot 7692.3076}\right)$$

$$P \in (1.44875 \cdot 10^{-11}, 4.1725 \cdot 10^{-8}] :$$

$$\mathcal{E}(P) = 0.0000051 \left(1 - e^{-P \cdot 0.2373 \cdot 10^{10}}\right) + 0.00014 \left(1 - e^{-P \cdot 0.4020 \cdot 10^8}\right)$$

$$P < 1.44875 \cdot 10^{-11} :$$

$$\mathcal{E}(P) = 10^{c_0 + c_1 + c_2 + c_3 + c_4 + c_5}$$

Παλιρροιακές Παράμετροι: Tidal Love number k_2

- Άμεσα μετρήσιμο από GW
- Παράμετρος που ποσοτικοποιεί την απόκριση του αστέρα στις παλιρροιακές δυνάμεις
- Μεγαλύτερο k_2 ισοδυναμεί με μεγαλύτερο βαθμό παραμόρφωσης
- Αδιάστατο

$$k_2(\beta, y_R) = \frac{8\beta^5}{5} (1 - 2\beta)^2 [2 - y_R + 2\beta(y_R - 1)] \\ \times [2\beta(6 - 3y_R + 3\beta(5y_R - 8)) \\ + 4\beta^3 (13 - 11y_R + \beta(3y_R - 2) + 2\beta^2(1 + y_R))] \\ + 3(1 - 2\beta)^2 [2 - y_R + 2\beta(y_R - 1)] \ln(1 - 2\beta)]^{-1}$$

Compactness parameter: $\beta \propto M/R$

Παλιρροιακές Παράμετροι: y_R

$$r \frac{dy(r)}{dr} + y^2(r) + y(r)F(r) + r^2Q(r) = 0$$

Οριακές Συνθήκες: $y(0) = 2$, $y(R) \equiv y_R$

$F(r)$ και $Q(r)$ συναρτήσεις των $\mathcal{E}(r)$, $P(r)$, $M(r)$

Παλιρροιακές Παράμετροι: Παλιρροιακή Παραμόρφωση λ

- Άμεσα μετρήσιμο από GW
- Ποσοτικοποιεί την παραμόρφωση του αστέρα λόγω των παλιρροιακών δυνάμεων
- Παρέχει πληροφορίες για εσωτερική δομή και Κ.Ε. του αστέρα

$$\lambda = \frac{2}{3} k_2 R^5$$

Αδιάστατο:

$$\Lambda = \frac{\lambda}{M^5}$$

Αριθμητική Επίλυση με Python

Για κάθε Κ.Ε:

- Ταυτόχρονη επίλυση συστήματος TOV και διαφορικής για y_R (αριθμητική ολοκλήρωση από $r = 0$ ως $r = R$)
- Οριακές συνθήκες: $P(0) = P_c$, $M(0) = 0$, $y(0) = 2$
 - Χρησιμοποιούνται πολύ μικρές τιμές αντί για 0
 - $P_c = 1 - 1200 \text{ MeV} \cdot \text{fm}^{-3}$
- Στην επιφάνεια του αστέρα: $P(R) = 0$, $M(R) = M$, $y(R) = y_R$
- Από TOV προκύπτουν M , R , άρα και το β
- Από την άλλη εξίσωση προκύπτει το y_R
- Από τα παραπάνω, υπολογίζονται k_2 , λ , Λ
- Δημιουργούνται οι αντίστοιχες γραφικές παραστάσεις

Κώδικας Python: TOV solver

```
# -*- coding: utf-8 -*-
#%matplotlib inline
import numpy as np
import matplotlib . pyplot as plt
from scipy . integrate import solve_ivp
import os , json
import myeoslibs
from sympy import *

# Define right - hand system of equations
def tovrhs (r,z):
    M = z [0] # mass coordinate as part of array z
    P = z [1] # pressure coordinate as part of array z
    y = z [2] #yr parameter
    if P0 >= 0.184:
        e = myeoslibs.eoslib[ 'eosype' ](P) #Equation of state
        dd= deos(P)
    elif P0 >= 9.34375e-5:
        e= myeoslibs.eoslib[ 'eoscore' ](P) #1st crust equation
        dd= deoscore(P)
    elif P0 >= 4.1725e-8:
        e= myeoslibs.eoslib[ 'eoscore2' ](P) #2nd crust equation
        dd= deoscore2(P)
    elif P0 >= 1.44875e-11:
        e= myeoslibs.eoslib[ 'eoscore3' ](P) #3rd crust equation
        dd= deoscore3(P)
    else :
        e= myeoslibs.eoslib[ 'eoscore4' ](P) #4th crust equation
        dd= deoscore4(P)
    dddt = 11.2 * (10 ** (-6)) * (r** 2) * e #1st order derivative of Mass
    ddt = - 1.474 * (M/(r** 2)) * ((1 + 11.2 * (10 ** (-6)) * (r** 3)** 3) * P/M) * ((1 - 2.948 * M/r )** (-1)) #1st order derivative of Pressure
    F = (1 - 1.474 * 11.2 * (10 ** (-6)) * (r** 2) * (e-P)) * ((1 - 2.948 * M/r )** (-1)) #F part
    J = 1.474 * 11.2 * (10 ** (-6)) * (r** 2) * (5 * e*9* P + ((e+P)/(1/dd))) * ((1 - 2.948 * M/r )** (-1)) - 6* ((1 - 2.948 * M/r )** (-1)) - 4* ((1.474 ** 2) * (dydt - (-y* dydt)/r))
    dydt = [ddt , ddt , dydt ] # array of derivatives
    return dydt # return the array of derivatives

# initial conditions
Step = 0.0001;
G = 6.674*10**(- 33) ;
n = 100;
ic1 = np.arange(1.5 , 5 , 0.1)
ic2 = np.arange(5 , 1201 , 1)
ic=np.concatenate(( ic1 , ic2 ),axis = None )

# insert EOS and crust EOS
eostype = 'lattimer_soft'
pp = Symbol('pp')
eosfuncprime = myeoslibs.eossym[ eosype ]().diff(pp)
deos = lambdify (pp , eosfuncprime , 'numpy')

eosfuncprimecore = myeoslibs.eossym[ 'eoscore' ]().diff( pp )
deoscore = lambdify (pp , eosfuncprimecore , 'numpy')
eosfuncprimecore2 = myeoslibs.eossym[ 'eoscore2' ]().diff( pp )
deoscore2 = lambdify (pp , eosfuncprimecore2 , 'numpy')
eosfuncprimecore3 = myeoslibs.eossym[ 'eoscore3' ]().diff( pp )
deoscore3 = lambdify (pp , eosfuncprimecore3 , 'numpy')
eosfuncprimecore4 = myeoslibs.eossym[ 'eoscore4' ]().diff( pp )
deoscore4 = lambdify (pp , eosfuncprimecore4 , 'numpy')
```

Κώδικας Python: TOV solver

```
# setting up working directory
directory = os.path.join( os.getcwd() , eosstype )
if not os.path.exists( directory ):
    print(" path doesn't exist.trying to make ")
    os.makedirs( directory )

# SOLVING#
minmax = np.zeros (( len( ic ) , 6 ) );
j = 0;

for i in ic :
    z0 =[0.00000000001 , i ,2.]
    P0 = 1.0;
    rmin =-0.0000001;
    rmax =+0.1;
    Mf = np.array ([])
    Pf = np.array ([])
    R = np.array ([])
    yr = np.array ([])
    z0_old = np.array ([])
    print(i)
    while ( P0>1e-12 ) :
        res = solve_ivp ( tovrhs , ( rmin , rmax ) , z0 , method ='LSODA' , atol =10 ** -26 , rtol=10 ** -8 )
        #print(res.y[0][:-1])
        z0_old = z0 [1]
        z0 [0] = res.y [1][~np.isnan ( res.y [1] ) ][-1]
        z0 [0] = res.y [0][~np.isnan ( res.y [0] ) ][-1]
        z0 [2] = res.y [2][~np.isnan ( res.y [2] ) ][-1]
        if z0 [0]<0:
            break
        if ( z0_old == z0 [1] ) :
            break
        rmin = res.t[~np.isnan ( res.y [2] ) ][-1]
        rmax = rmin + 0.001
        P0 = z0 [1]
        Mf = np.append (Mf , res.y [0][~np.isnan ( res.y [0] ) ])
        Pf = np.append (Pf , res.y [1][~np.isnan ( res.y [1] ) ])
        yr = np.append (yr , res.y [2][~np.isnan ( res.y [2] ) ])
        R = np.append (R , res.t[~np.isnan ( res.y [2] ) ])
    if Pf [-1]<0:
        idx = np.argwhere ( Pf<0 ) [0 , 0]
        Pf = np.delete (Pf , np.s_ [ idx : : ] , 0)
        Mf = np.delete (Mf , np.s_ [ idx : : ] , 0)
        yr = np.delete (yr , np.s_ [ idx : : ] , 0)
        R = np.delete (R , np.s_ [ idx : : ] , 0)
        minmax [ j ] = R[-1] , max ( Mf ) ,min ( Pf ) , yr [-1] ,0 ,0
        j = j + 1;
    beta = 1.474 * minmax [ j-1 ][1] / minmax [ j-1 ][0]
    k2 = ( ( beta ** 5/5) * ((1 -2* beta )** 2)* (2*yr [-1]+2* beta * ( yr [-1]-1)) * (2*beta * (6-3* yr [-1]+3* beta * (5* yr [-1]-2)+2* ( beta ** 2)*(1+yr [-1])) )** 3* ((1 -2* beta )** 2)* (2*yr [-1]+2* beta * ( yr [-1]-1)) * np . log (1-2* beta ))** (1 -2/3* (( R[-1] ** 5/6) / k2) * 10** (- 36))
    minmax [ j-1 ][4] = k2
    minmax [ j-1 ][5] = 1

with open ( os . path . join ( directory , 'data_+' + eosstype +'.txt' ) , mode ='w') as f1 :
    myfile =----- data for EoS:' + eosstype +'\r\n'
    f1 . write ( myfile )
with open( os . path .join( directory , 'data_+' + eosstype +'.txt' ) , mode ='ab') as f1 :
    np . savetxt ( f1 , minmax , delimiter =",", header = "radius R (km)\t\t mass M (Msun )\t\t pressure P \t\t yr \t\t k2 \t\t lambda "
with open ( os . path . join ( directory , 'data_+' + eosstype +'.json' ) , 'w') as outfile :
    json . dump ( minmax . tolist () , outfile )
```

Κώδικας Python: Γραφήματα

```
@author: zaxari
"""
import pickle
import numpy as np
import matplotlib.pyplot as plt
import json
import os
from scipy.stats import pearsonr, spearmanr
from scipy.stats import linregress

# Define a list of EOSs
eos_list = ['MDI1', 'W', 'MFF1', 'APR1', 'MDI4', 'BGP', 'HM31', 'SKI4', 'MDI2', 'MDI3', 'MFF2', 'HM32', 'BL1', 'BL2']
# eos is a custom color palette with shades of blue, pink, purple, and red
custom_colors = [
    '#0099cc', '#ff66b2', '#9900cc', '#ff6666', '#3366ff', '#ff99cc', '#6633cc',
    '#ff3333', '#33ccff', '#ff6699', '#cc0066', '#00c0b0', 'g', '#9999ccfb',
    '#ffcc00', '#ccff66', '#ff9900', '#ff6600', '#ff3300', '#13cfef', '#4444ccfb'
]

# Define a list of line styles for each EOS
line_styles = ['-', '--', '-.', '-o', '-x', '-^', '->', '-<', '-s', '-d', '->.', '-<.', '-s.', '-d.', '->.-', '-<.-', '-s.-', '-d.-']

# Initialize lists to store data for each EOS
r_1_4_msun = []
lambda_1_4_msun = []
k2_1_4_msun = []
masses_list = []
radii_list = []
k2_values_list = []
lambda_values_list = []
yr_values_list = []
beta_values = [] # New list to store beta values
lambda_at_1_4_msun_list = [] # New list to store lambda at R(1.4M $\odot$ ) values
# Loop through each EOS
for i, eos in enumerate(eos_list):
    # Load data from the JSON file for the current EOS
    with open(os.getcwd() + f"\data_{eos}.json", 'r') as json_file:
        minmax_data = json.load(json_file)

    minmax = np.array(minmax_data)
    bound = np.where(minmax[:,1]==max(minmax[:,1]))[0][0]

    # Extract data for plotting
    radii = minmax[:,bound, 0] # Radius in km
    masses = minmax[:,bound, 1] # Mass in Msun
    k2_values = minmax[:,bound, 4] # k2 value
    lambda_values = minmax[:,bound, 5] # Lambda ( $\lambda$ ) values
    yr_values = minmax[:,bound, 3] # yr values (update if necessary)

    # Find the radius corresponding to 1.4M $\odot$  for the current EOS
    index_1_4_msun = np.argmax(np.abs(masses - 1.4))
    r_1_4_msun.append(radii[index_1_4_msun])
    lambda_1_4_msun.append(lambda_values[index_1_4_msun])
    k2_1_4_msun.append(k2_values[index_1_4_msun])
```

Κώδικας Python: Γραφήματα

```
# Plot M-R for each EOS
plt.figure(figsize=(10, 6), dpi=300)
with open("GW170817_MR1_maxmass_contour_data.pkl","rb") as f:
    ligo1 = pickle.load(f)
with open("GW170817_MR2_maxmass_contour_data.pkl","rb") as f:
    ligo2 = pickle.load(f)
ligo1['z_grid']=np.where(ligo1['z_grid']==0., ligo1['z_grid'],np.nan)
ligo2['z_grid']=np.where(ligo2['z_grid']==0., ligo2['z_grid'],np.nan)
#reloading invalid values so sorting is quicker
x1 = list(filter(lambda x: (not np.isnan(x)), ligo1['z_grid'].flatten()))
x2 = list(filter(lambda x: (not np.isnan(x)), ligo2['z_grid'].flatten()))
#function for getting the value of probability density where the confidence % cuts-off
def getProbCutoff(lst, confidence):
    lst.sort(reverse=True)
    S = sum(lst)
    csum = 0
    out = 0
    for i in lst:
        csum += i
        sprint("sum:",csum, " percentage:",csum/S, " i:", i)
        if csum/confidence*S:
            out = i
            break
    return out
#plots probability density
plt.pcolormesh(ligo1['xx'], ligo1['yy'], ligo1['z_grid'], shading='auto', cmap=plt.cm.PuBu)
plt.pcolormesh(ligo2['xx'], ligo2['yy'], ligo2['z_grid'], shading='auto', cmap=plt.cm.YlOrBr)
#plots 90% confidence region boundary
plt.contour(ligo1['xx'], ligo1['yy'], ligo1['z_grid'], [getProbCutoff(x1, 0.5)],linestyles = "dashed", linewidths = .5, colors=[ "w"])
plt.contour(ligo2['xx'], ligo2['yy'], ligo2['z_grid'], [getProbCutoff(x2, 0.5)],linestyles = "dashed", linewidths = .5, colors=[ "w"])
#plots 50% confidence region boundary
plt.contour(ligo1['xx'], ligo1['yy'], ligo1['z_grid'], [getProbCutoff(x1, 0.9)], linewidths = .5, colors=[ "gray"])
plt.contour(ligo2['xx'], ligo2['yy'], ligo2['z_grid'], [getProbCutoff(x2, 0.9)], linewidths = .5, colors=[ "gray"])
for i, eos in enumerate(eos_list):
    plt.plot(rad1_list[i], masses_list[i], label=eos, color=custom_colors[i], linestyle=linestyles[i], linewidth=line_thickness)

plt.xlim([8,20])
plt.xlabel('Radius (km)')
plt.ylabel('Mass (Msun)')
plt.title('Mass-Radius (M-R) Relation for Neutron Stars')
plt.grid()
plt.legend(title='EOS')
plt.savefig('plots/M_R_relation.png', dpi=300) # Save the graph
plt.show()

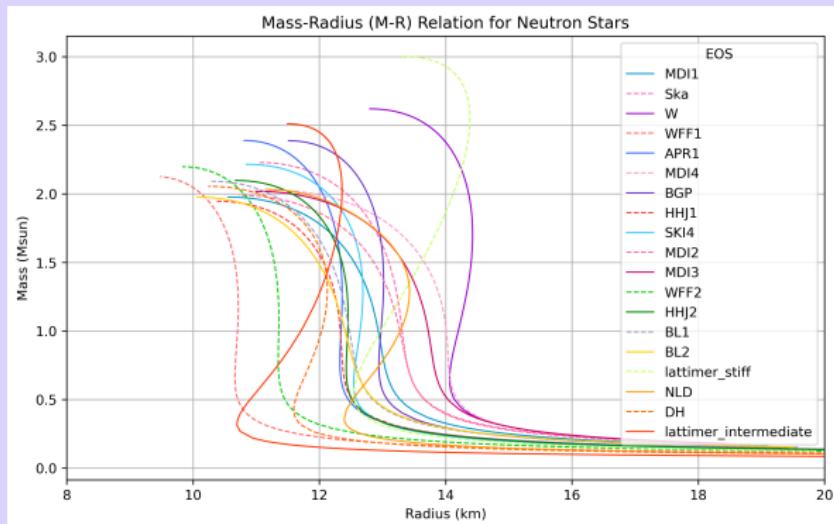
# Plot λ-M for each EOS
plt.figure(figsize=(10, 6), dpi=300)
for i, eos in enumerate(eos_list):
    plt.plot(masses_list[i], lambda_values_list[i], label=eos, color=custom_colors[i], linestyle=linestyles[i], linewidth=line_thickness)

plt.ylim([0,8])
plt.xlabel('Mass (Msun)')
plt.ylabel('Lambda (λ)')
plt.title('Lambda-Mass (λ-M) Relation for Neutron Stars')
plt.grid()
plt.legend(title='EOS')
plt.savefig('plots/lambda-M_relation.png', dpi=300) # Save the graph
plt.show()
```

Αποτελέσματα και Σχόλια

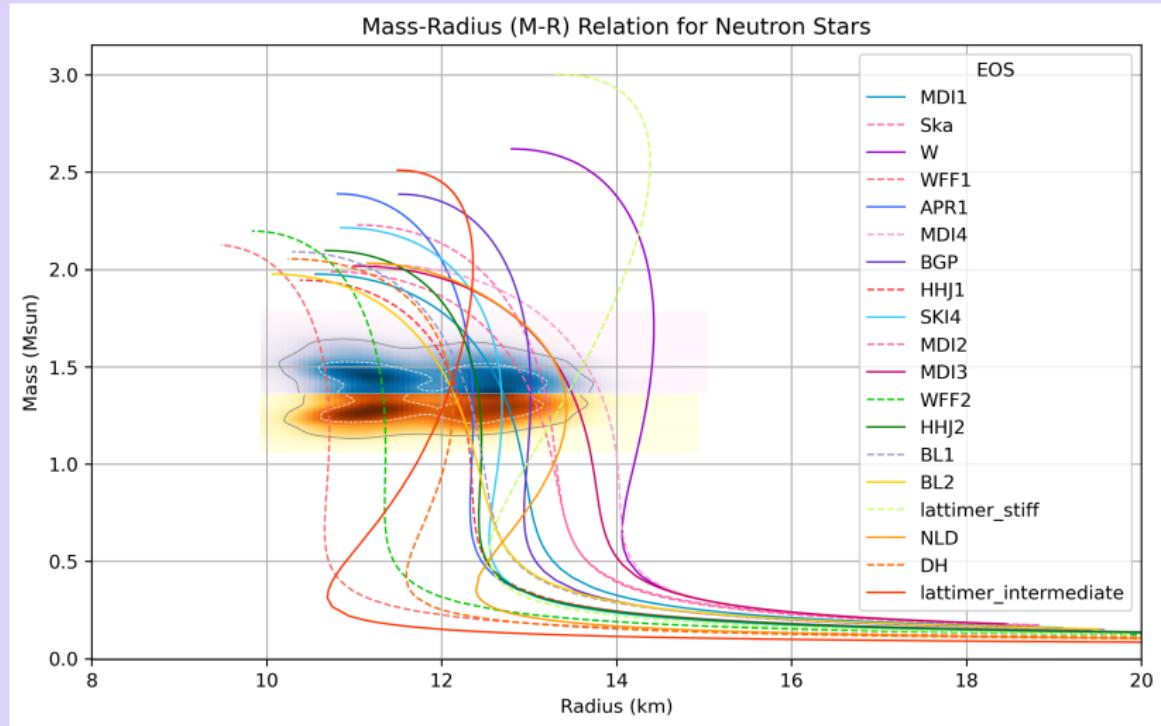
Αποτελέσματα: M-R

- $M_{max} \approx 2 - 2.5M_\odot$ από κάθε Κ.Ε. (συμφωνία με γνωστά δεδομένα)
- $R > 15km : M \rightarrow 0$
- Οι πιο stiff Κ.Ε. παράγουν γενικά μεγαλύτερες M και R



M-R

Αποτελέσματα: M-R



M-R με δεδομένα LIGO

Αποτελέσματα: M-R

Σύγκριση με δεδομένα από LIGO (BNS GW170817 event) :

- Παραμετροποιημένη μέθοδος
- Ελάχιστο όριο Μέγιστης Μάζας: $M_{max} = 1.97M_{\odot}$

Κ.Ε. εκτός Ορίων:

- ① **W:** Η πιο stiff, προβλέπει τις μεγαλύτερες M και R
- ② **MDI-4:** Λιγότερο stiff, αλλά κι αυτή προβλέπει μεγάλες τιμές R

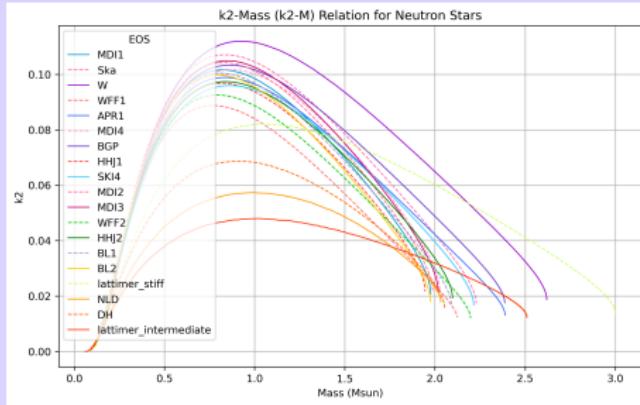
Οι υπόλοιπες 17 Κ.Ε. εντός περιοχών αξιοπιστίας 90% και 50%

Αποτελέσματα: M-R

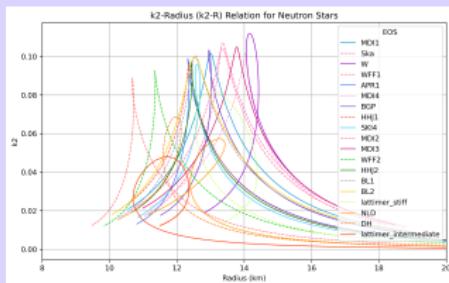
EOS	M_{max} (M_\odot)	$R(M_{max})$	$R(1.4M_\odot)$ (km)
MDI1	1.976194	10.556958	12.671
Ska	2.228849	11.035806	13.167
W	2.61987	12.8	14.378
WFF1	2.12574	9.479508	10.694
APR1	2.388921	10.810529	12.355
MDI4	2.026944	11.202	13.778
BGP	2.387047	11.513942	13.018
HHJ1	1.945446	10.381881	12.159
SKI4	2.214984	10.847	12.681
MDI2	1.988299	10.735	12.99
MDI3	2.018318	10.996	13.456
WFF2	2.197732	9.831483	11.32
HHJ2	2.097361	10.672908	12.407
BL1	2.091228	10.290629	12.272
BL2	1.975839	10.067	12.103
lattimer_stiff (HLPS-3)	3.003023	13.291725	13.479
NLD	2.032309	11.156	13.396
DH	2.054601	10.246879	12.115
lattimer_intermediate (HLPS-2)	2.510308	11.500784	12.109

Πίνακας με τιμές M_{max} , $R(M_{max})$, $R(1.4M_\odot)$ για κάθε Κ.Ε.

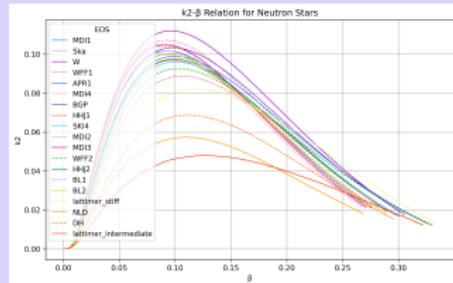
Αποτελέσματα: tidal Love number k_2



$$k_2 - M$$



$$k_2 - R$$



$$k_2 - \beta$$

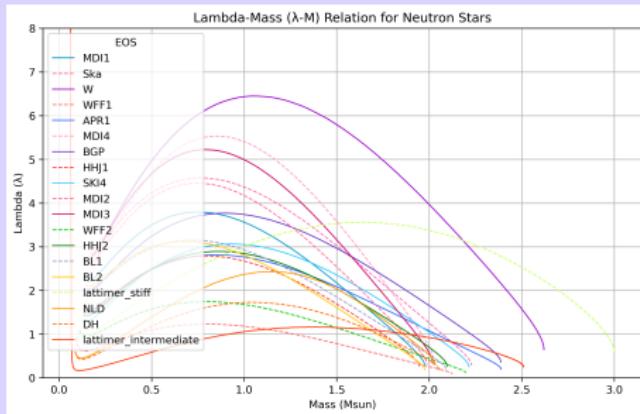
Αποτελέσματα: tidal Love number k_2

Πιο stiff Κ.Ε. παράγουν μεγαλύτερες τιμές για το k_2 . Δηλαδή, εν γένει:

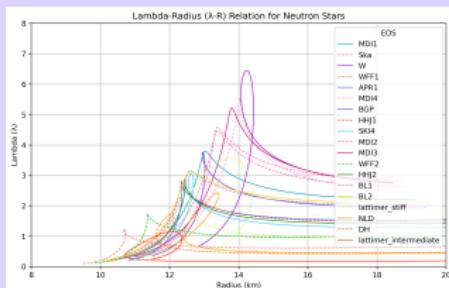
- Μεγαλύτερες M συνδέονται με μεγαλύτερες τιμές k_2
- Μεγαλύτερες R συνδέονται με μεγαλύτερες τιμές k_2
- $k_2 - \beta$ διάγραμμα έχει παρόμοια συμπεριφορά με $k_2 - M$
 $(\beta = M/R)$

Δεν παρατηρείται ιδιαίτερα μεγάλο εύρος τιμών του k_2 .

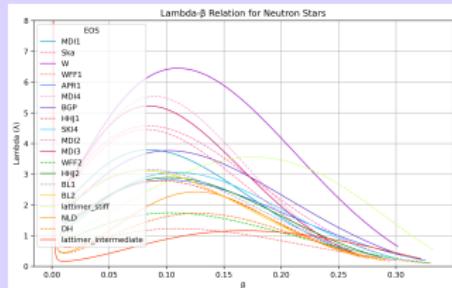
Αποτελέσματα: Παλλοιριακή παραμόρφωση λ



$$\lambda - M$$



$$\lambda - R$$



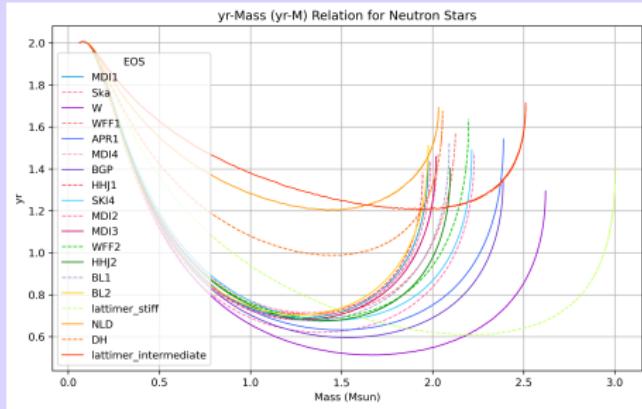
Αποτελέσματα: Παλλοιριακή παραμόρφωση λ

Το λ παρουσιάζει σαφώς πιο μεγάλο εύρος τιμών ανα τις διαφορετικές Κ.Ε., από $\lambda = 1$ ως και $\lambda = 6$ για M_{max} ($\lambda \propto k_2 \cdot R^5$)

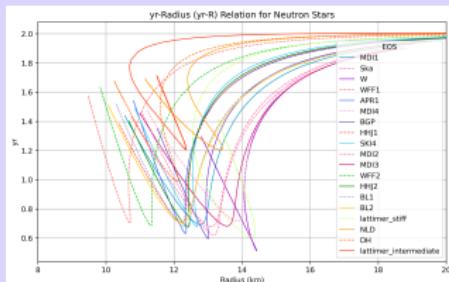
Πιο stiff Κ.Ε. παράγουν μεγαλύτερες τιμές λ . Δηλαδή εν γένει:

- Μεγαλύτερες M συνδέονται με μεγαλύτερες τιμές λ
- Μεγαλύτερες R συνδέονται με μεγαλύτερες τιμές λ (ισχυρή εξάρτηση)
- $\lambda - \beta$ διάγραμμα έχει παρόμοια συμπεριφορά με $\lambda - M$ ($\beta = M/R$)

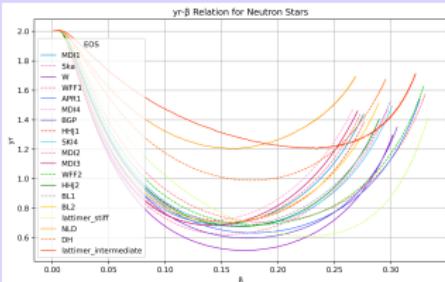
Αποτελέσματα: y_R



$$y_R - M$$



$$y_R - R$$



Αποτελέσματα: y_R

$$M = 0 : y_R = 2 \text{ (Οριακή συνθήκη)}$$

Πιο stiff Κ.Ε. παράγουν μικρότερες τιμές y_R . Δηλαδή εν γένει:

- Μεγαλύτερες M συνδέονται με μικρότερες τιμές y_R
- Μεγαλύτερες R συνδέονται με μικρότερες τιμές y_R
- $y_R - \beta$ διάγραμμα έχει παρόμοια συμπεριφορά με $y_R - M$
($\beta = M/R$)

Στατιστική Ανάλυση

Τελευταίο βήμα: Αναζήτηση **συστηματικής σχέσης**.

Μελετήθηκε η σχέση της $R(1.4M_{\odot})$ με τα μεγέθη k_2 , λ και Λ .

Επιλέχθηκε αυτή η τιμή για τη μάζα, καθώς έχει βρεθεί πως είναι η πιο πιθανή τιμή μαζας για Αστέρα Νετρονίων.

Μέθοδοι:

- ① Correlation Analysis
- ② Linear Regression

Στατιστική Ανάλυση

EOS	$R(1.4M_{\odot})$ (km)	k_2	λ	Λ
MDI1	12.671	0.077011	2.512634	0.469412
Ska	13.167	0.085126	3.365247	0.634381
W	14.378	0.098633	6.05392	1.141928
WFF1	10.694	0.063174	0.882599	0.164862
APR1	12.355	0.078651	2.261741	0.417152
MDI4	13.778	0.08157	4.04562	0.739985
BGP	13.018	0.085065	3.176832	0.584304
HHJ1	12.159	0.072847	1.933852	0.356211
SKI4	12.681	0.077939	2.552968	0.468251
MDI2	12.99	0.079779	2.947513	0.555037
MDI3	13.456	0.082147	3.619888	0.673405
WFF2	11.32	0.068652	1.274697	0.236154
HHJ2	12.407	0.076816	2.255832	0.42192
BL1	12.272	0.075385	2.095955	0.387039
BL2	12.103	0.072563	1.882353	0.351743
lattimer_stiff (HLPS-3)	13.479	0.078131	3.472447	0.665218
NLD	13.396	0.051405	2.215155	0.404503
DH	12.115	0.057858	1.50834	0.277762
lattimer_intermediate (HLPS-2)	12.109	0.044523	1.157831	0.211274

Πίνακας με τιμές $R(1.4M_{\odot})$, k_2 , λ , Λ για κάθε Κ.Ε.

Στατιστική Ανάλυση: Correlation Analysis

Οι συνθήκες που τέθηκαν για την ερμηνεία των αποτελεσμάτων:

Pearson's correlation coefficient	Pearson's p value	Linear Correlation
≥ 0.7	$< \alpha$	Strong & Positive
≥ 0.7	$\geq \alpha$	Not statistically Significant
[0.3, 0.7)	$< \alpha$	Moderate & Positive
[0.3, 0.7)	$\geq \alpha$	Not statistically Significant
< 0.3		Weak or None

Διαστήματα ερμηνείας για την Correlation Analysis μέθοδο

Τέθηκε επίπεδο στατιστικής σημαντικότητας $\alpha = 0.05$

Στατιστική Ανάλυση: Correlation Analysis

```
Pearson's correlation coefficient (R(1.4M $\odot$ )-k2): 0.544204
Pearson's p-value (R(1.4M $\odot$ )-k2): 0.016002
Spearman's rank correlation coefficient (R(1.4M $\odot$ )-k2): 0.731579
Spearman's p-value (R(1.4M $\odot$ )-k2): 0.000371
Pearson's correlation coefficient (R(1.4M $\odot$ )-λ): 0.902636
Pearson's p-value (R(1.4M $\odot$ )-λ): 0.000000
Spearman's rank correlation coefficient (R(1.4M $\odot$ )-λ): 0.938596
Spearman's p-value (R(1.4M $\odot$ )-λ): 0.000000
Interpretation for R(1.4M $\odot$ )-k2: Moderate positive linear correlation
Interpretation for R(1.4M $\odot$ )-λ: Strong positive linear correlation
```

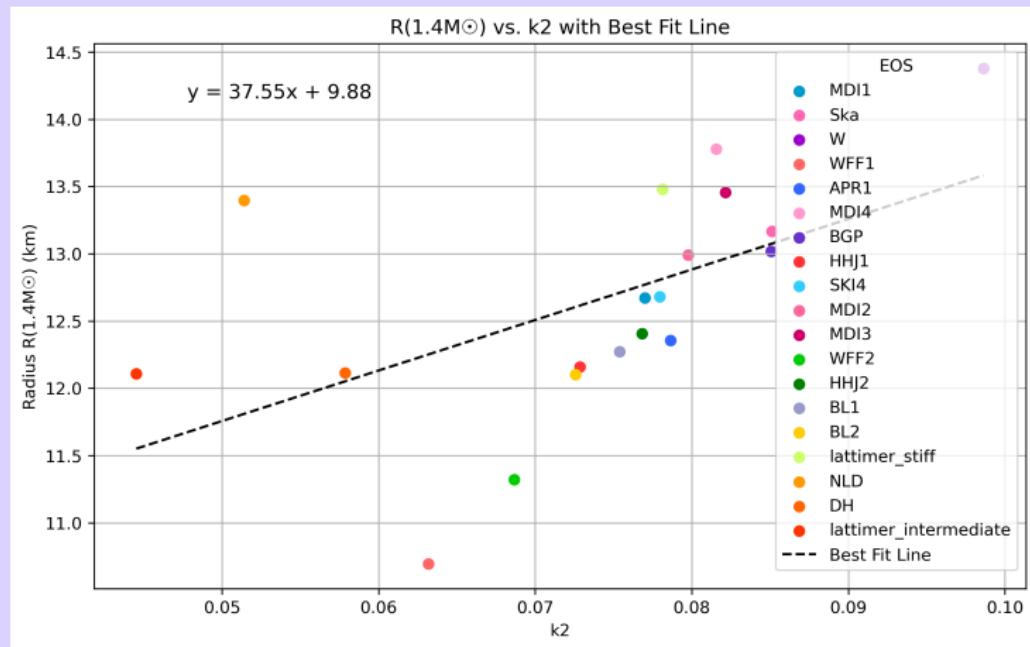
Αποτελέσματα κώδικα (k_2, λ)

```
Pearson's correlation coefficient (R(1.4M $\odot$ )-Λ): 0.898276
Pearson's p-value (R(1.4M $\odot$ )-Λ): 0.000000
Spearman's rank correlation coefficient (R(1.4M $\odot$ )-Λ): 0.938596
Spearman's p-value (R(1.4M $\odot$ )-Λ): 0.000000
Interpretation for R(1.4M $\odot$ )-Λ: Strong positive linear correlation
```

Αποτελέσματα κώδικα (Λ)

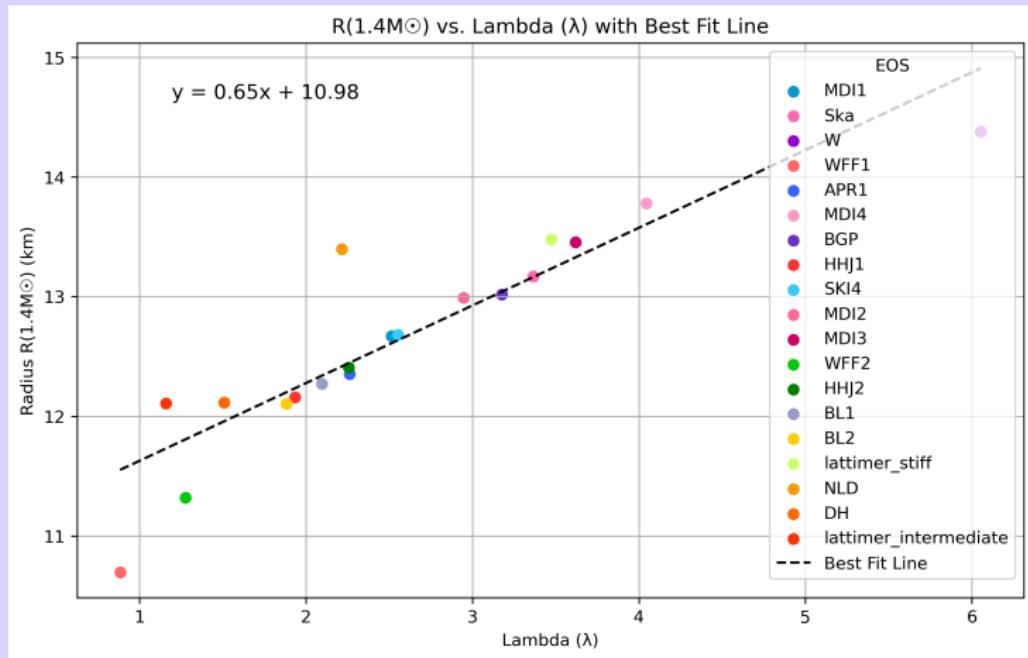
- $R(1.4M_{\odot}) - k_2$: Μέτρια, θετική γραμμική συσχέτηση
- $R(1.4M_{\odot}) - \lambda$: Ισχυρή, θετική γραμμική συσχέτηση
- $R(1.4M_{\odot}) - \Lambda$: Ισχυρή, θετική γραμμική συσχέτηση

Στατιστική Ανάλυση: Linear Regression



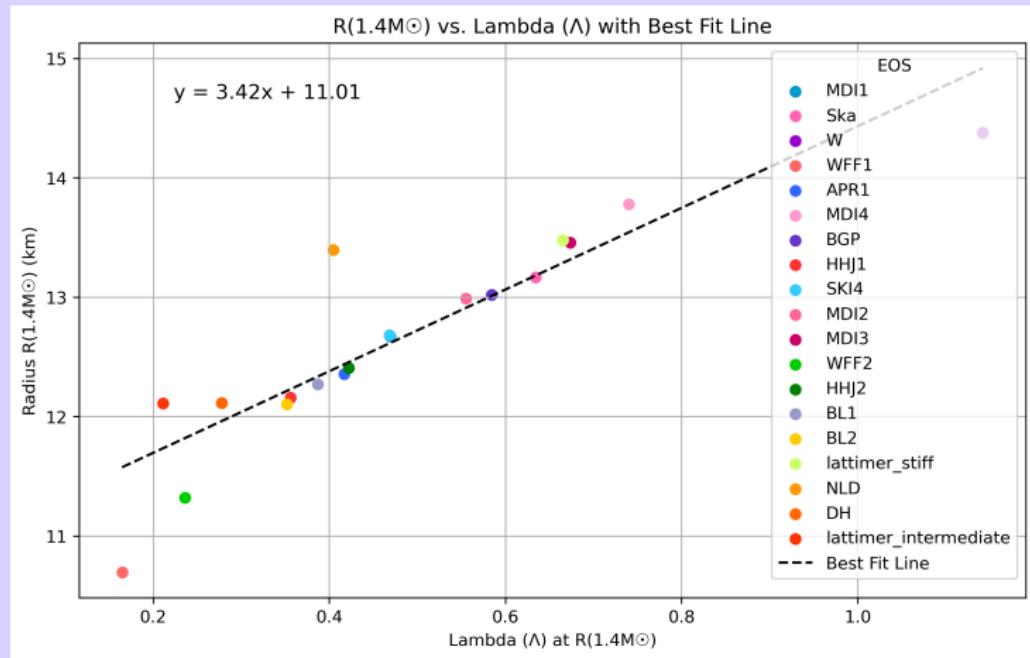
$R(1.4M_{\odot}) - k_2$ best fit

Στατιστική Ανάλυση: Linear Regression



$$R(1.4M_{\odot}) - \lambda \text{ best fit}$$

Στατιστική Ανάλυση: Linear Regression



$$R(1.4M\odot) - \Lambda \text{ best fit}$$

Στατιστική Ανάλυση: Linear Regression

Οι καλύτερες ευθείες που προέκυψαν είναι:

$$R_{1.4} = 37.55k_2 + 9.88$$

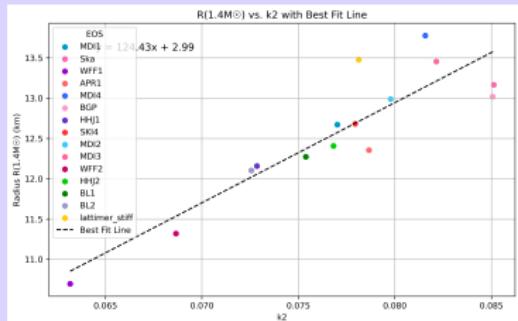
$$R_{1.4} = 0.65\lambda + 10.98$$

$$R_{1.4} = 3.42\Lambda + 11.01$$

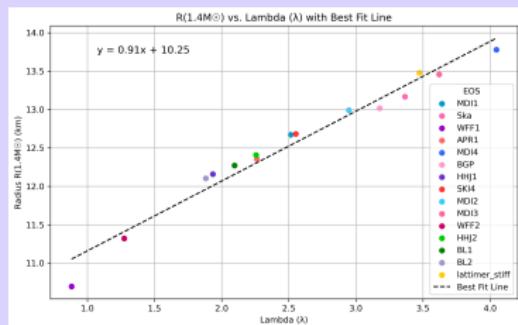
Παρατηρώντας τις γραφικές παραστάσεις:

- Υπάρχει σχέση γραμμικότητας με την ακτίνα $R(1.4M_\odot)$ ειδικά για τα λ και Λ
- Τα σημεία που 'χαλανε' τη γραμμικότητα αντιστοιχούν στις πιο stiff και πιο soft K.E.

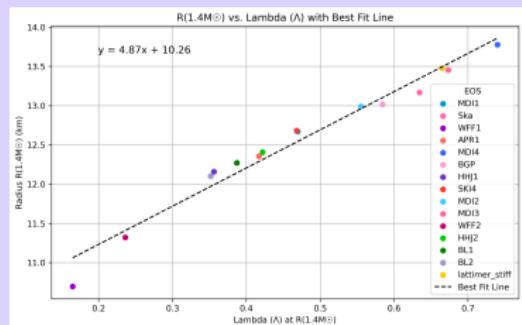
Στατιστική Ανάλυση: Linear Regression



$R(1.4M_\odot) - k_2$ best fit v.2



$R(1.4M_\odot) - \lambda$ best fit v.2



$R(1.4M_\odot) - \Lambda$ best fit v.2

Συμπεράσματα

Συμπεράσματα

- Έγινε χρήση κώδικα python για την επίλυση των συστήματος TOV και εξισώσεων για τον υπολογισμό παλιρροιακών παραμέτρων
- Βρέθηκαν M και R για κάθε K.E. και μέσω του $M - R$ διαγράμματος παρατηρήσαμε ότι οι πιο stiff K.E. προβλέπουν γενικά μεγαλύτερες τιμές Μάζας και Ακτίνας.
- Οι K.E. που χρησιμοποιήθηκαν αναπαράγουν μέγιστη μάζα τουλάχιστον $M_{max} = 2M_{\odot}$: συνέπεια με τα όρια που προκύπτουν από παρατηρήσεις.
- Το $M - R$ διαγράμμα μελετήθηκε σε συνδυασμό με contours παρατηρήσεων από το LIGO. 2 K.E. βρέθηκαν εκτός επιπέδων εμπιστοσύνης, (η μία ήταν η πιο stiff): ένδειξή πως ίσως πρέπει να αποκλειστούν

- Έγιναν διαγράμματα για τη μελέτη συμπεριφοράς των tidal parameters σε σχέση με M , R , β . Οι stiffer K.E. παράγουν μεγαλύτερες τιμές για k_2 και λ σε αντίθεση με τις πιο soft. Επιβεβαιώθηκε ακόμη η έντονη εξάρτηση του λ από την ακτίνα R
- Έγινε αναζήτηση συστηματικής μεταξύ του $R(1.4M_{\odot})$ και των k_2 , λ , Λ . Οι δύο τελευταίες εμφάνισαν καλή σχέση γραμμικότητας.

Σας ευχαριστώ για την προσοχή σας!