

姓名	学号	班级	选题	论述	结论	总分
郭潇	2013301020099	物理弘毅班				

中子星模型的数值求解

郭潇 2013301020099 物理弘毅班

摘要：中子星是天体物理中十分重要的天体，它的理论建模相当复杂，需要用到广义相对论和统计物理，甚至强相互作用的理论等。本文试图在所学的能力范围之下，数值求解这一相当复杂的问题，在考虑了广义相对论产生的影响下，使用纯中子 Fermi 气体模型这一稍微简单一些的模型（但也不简单）构建了中子星的模型，使用精确的理论表达式代入微分方程求解了它的物理状态（即质量、压强、密度等物理量径向的分布），以及确定中子星的质量与其半径关系，并预言了中子星的质量上限和最小半径。

关键词：中子星，天体物理，TOV 方程，Fermi 气体模型，广义相对论。

目录

中子星模型的数值求解.....	1
1、 导言.....	1
2、中子星建模.....	2
2.1 质量连续性方程.....	2
2.2 TOV 方程.....	2
2.3 状态方程.....	3
3、数值求解.....	3
4、中子星的状态函数解.....	4
5、中子星的质量与半径关系.....	6
5.1 笔者的结果.....	6
5.2 Oppenheimer 和 Volkoff 的结果.....	7
5.3 两者间的比较.....	7
6、结论.....	8
附：程序代码.....	8
参考文献.....	10
致谢.....	11

1、导言

致密星是恒星演化末期的残骸，它不再燃烧核燃料，因此如果没有其他压强抵抗引力，在引力的吸引下它就会崩塌。

致密星分为三类：白矮星、中子星、黑洞，中子星是靠重子（中子、质子、超子等）和

夸克的简并压强来抵抗引力的吸引以达到平衡。由于中子星具有强引力场、强电磁场和极端的强相互作用与弱相互作用等，因此备受天体物理学家，甚至理论物理学家们的重视。

本文就考虑完全由中子构成的中子星模型，它以中子的简并压强来抵抗引力的吸引（忽略了强相互作用）。最终根据所给的模型可以确定中子星的质量与其半径关系，并预言了中子星的质量上限和最小半径。

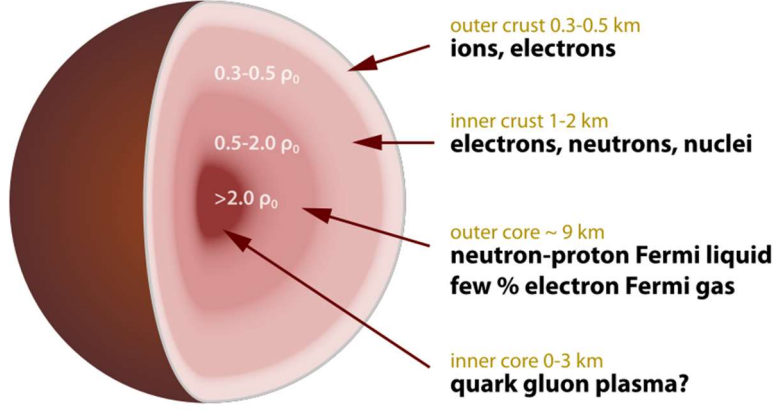


图 1 中子星可能的内部结构，图片来自维基百科^[5]

2、中子星建模

实际中子星的构成可能是相当复杂的，它也是分层的，如图 1，其间除了包含中子外，还包含一部分电子与质子、 π 介子，甚至超子等多种夸克物质，并且各种夸克物质之间还存在强相互作用，关于强相互作用的理论已超出笔者能力所及，为了简化问题，便于计算，我们忽略了强相互作用，考虑完全由中子构成的中子星，采用 Fermi 气体的模型计算其简并压强。

2.1 质量连续性方程

由中子星的球对称性，我们可以很容易得到

$$\frac{dm(r)}{dr} = 4\pi r^2 \rho(r)$$

其中 $m(r)$ 代表中子星从中心到半径 r 处的球体内包含的质量， $\rho(r)$ 是密度的径向分布函数。它揭示了 $m(r)$ 与 $\rho(r)$ 之间的关系。

2.2 TOV 方程

引力半径 $r_s = \frac{2GM}{c^2}$ 是标志引力场强弱的一个量，当天体半径 $r \ll r_s$ 时，属于弱引力场；而当天体半径接近 r_s 时，引力场足够强，这时广义相对论的效应就不能再被忽略掉了。由于中子星的质量半径比 M/R 很大，中子星的 $\frac{r_s}{R} = \frac{2GM}{Rc^2} \approx 0.1$ ，无疑地，我们需要用到广义相对论。^[1]

从球对称的时空度规

$$dS^2 = -e^{2\alpha(r)}dt^2 + e^{2\beta(r)}dr^2 + r^2d\theta^2 + r^2\sin^2\theta d\varphi^2$$

出发, 并作静态理想流体的假设, 于是有张量等式 $T_{\mu\nu} = (\rho + p)v_\mu v_\nu + pg_{\mu\nu}$ 和 $v^0v^0g_{00} = -1$, 再联立 Einstein 场方程 $G_{\mu\nu} = 8\pi T_{\mu\nu}$, 与能动量守恒方程 $\nabla_\mu T^{\mu\nu} = 0$, 经一番推导后, 最后补足常数可以得到 TOV 方程

$$\frac{dp(r)}{dr} = -(\rho(r) + \frac{p(r)}{c^2}) \frac{G(m(r) + \frac{4\pi r^3 p(r)}{c^2})}{r(r - \frac{2Gm(r)}{c^2})}$$

其中 $p(r)$ 是中子星半径为 r 处的压强, G 是万有引力常数。

TOV 方程与 Newton 的万有引力理论得出的结论相比

$$\frac{dp(r)}{dr} = -\frac{Gm(r)\rho(r)}{r^2}$$

要复杂得多, 并且 $\rho(r)$, $m(r)$, $p(r)$ 三者相互关联, 关系十分复杂。

2.3 状态方程

为求解中子星的状态函数的解, 即求解 $\rho(r)$, $m(r)$, $p(r)$ 三个函数, 我们还需要一个状态方程 $p = p(\rho(r))$ 来联系起它们。同时注意到前面两个方程都几乎是普适的, 它们与具体的物质形态、种类等都无关, 因此我们需要一个专门描述中子星的状态方程, 这也是我们的中子星模型的核心部分。

考虑到中子星用于对抗引力的压强来自于中子的简并压强(中子是 Fermion), 与白矮星以电子的简并压强对抗引力类似, 可以仿照白矮星的解法^[2], 从而得到中子星的 Fermi 气体模型, 只需将所有白矮星的相关参量全部替换成中子星的相关参量就可以, 比如将电子质量替换成中子质量等等。根据中子星的 Fermi 气体模型可以得到压强

$$p = \frac{m_n c^2}{3\pi^2 \lambda^3} I(\frac{p_F}{m_n c})$$

这里

$$\lambda = \frac{\hbar}{m_n c}$$

$$I(x) = \frac{3}{8} [x\sqrt{1+x^2} \left(\frac{2x^2}{3} - 1 \right) + \operatorname{arcsinh}(x)]$$

p_F 是 Fermi 动量, 并有

$$p_F = \hbar \left(\frac{3\pi^2 \rho}{m_n} \right)^{\frac{1}{3}}$$

代入上式, 最终可以得到十分复杂的状态方程 $p = p(\rho)$, 其中还包含了一个反双曲函数, 可见其复杂程度, 如果不作近似是几乎不可能解析求解的, 因此我们采用数值解法。

3、数值求解

将三个方程联立

$$\frac{dm(r)}{dr} = 4\pi r^2 \rho(r) \dots \dots \dots (1)$$

$$\frac{dp(r)}{dr} = -\left(\rho(r) + \frac{p(r)}{c^2}\right) \frac{G\left(m(r) + \frac{4\pi r^3 p(r)}{c^2}\right)}{r\left(r - \frac{2Gm(r)}{c^2}\right)} \dots \dots \dots (2)$$

$$p(r) = p(\rho(r)) \dots \dots \dots (3)$$

但是由于(3)式十分复杂，如果要反解出 $\rho(r)$ 作为 $p(r)$ 的函数的精确表达式，也是几乎不可能的（除非用带参数的函数拟合，但这样就会丧失一部分精确性），这样的话，如果不对上述方程加以修改，直接离散化是没有办法迭代的。要实现迭代，首先我们必须能根据 $\rho(r_0)$, $m(r_0)$, $p(r_0)$ 求得 $\rho(r_0 + dr)$ 的值，再代入(1)(3)式，分别求得 $m(r_0 + dr)$, $p(r_0 + dr)$ ，就可以将迭代继续下去，从而求解出 $\rho(r)$, $m(r)$, $p(r)$ 三个函数，因此要求方程(2)必须是 $\rho(r)$ 的微分方程，这样才可以从方程(2)中得到 $\rho(r)$ 的递推式。于是我们需要对方程(2)作一些处理如下：

$$\begin{aligned} \frac{dp(\rho)}{d\rho} \frac{d\rho}{dr} &= -\left(\rho(r) + \frac{p(r)}{c^2}\right) \frac{G\left(m(r) + \frac{4\pi r^3 p(r)}{c^2}\right)}{r\left(r - \frac{2Gm(r)}{c^2}\right)} \\ \frac{d\rho}{dr} &= -\frac{\left(\rho(r) + \frac{p(r)}{c^2}\right) G\left(m(r) + \frac{4\pi r^3 p(r)}{c^2}\right)}{\frac{dp(\rho)}{d\rho} r\left(r - \frac{2Gm(r)}{c^2}\right)} \dots \dots \dots (2)' \end{aligned}$$

由于 $p = p(\rho)$ 的表达式十分复杂，其导数的表达式将更加复杂，但在数值计算中，我们可以将求导离散化

$$\frac{dp(\rho(r))}{d\rho(r)} \Big|_{r=r_0} = \frac{p(\rho(r_0)) - p(\rho(r_0 - dr))}{\rho(r_0) - \rho(r_0 - dr)}$$

，只要将 dr 取得足够小，就可以忽略掉数值计算离散化带来的误差。

这两个微分方程加上一个代数方程构成的微分方程组，相当于一个二阶的微分方程，可以用 Euler-Cromer 方法求解，思路如前所述，加上初值条件 $\rho(0) = \rho_0$, $m(0) = 0$, $p(0) = p(\rho_0)$ 。不过计算中为了避免出现 $r=0$ ，同时为了方便计算导数 $\frac{dp(\rho)}{d\rho}$ ，我依次给定了两个初

始条件，由于 $\frac{d\rho}{dr} \Big|_{r=0} = 0$ ，可知两个初始条件原本应当皆为 ρ_0 ，但这样在求离散化的导数

时，又会出现 $\frac{0}{0}$ ，所以我把第二个 ρ_0 乘以 0.9999 稍微调小一些，如此一石两鸟，就都没问题了。

具体的程序代码可见文章末尾或[点击此处 NeutronStar.py](#)

4、中子星的状态函数解

鉴于计算所用的理论公式全部由基本理论导出，并且模型相对实际的中子星比较简单，因此中子星的物理状态只有一个可调参量就是中子星的中心密度 ρ_0 ，给定中心密度 ρ_0 后，各个物理量的空间分布都可以利用该程序求解出来，最后还可以得到中子星的总质量与半径大小

关系。

选取一个典型的中子星的中心密度 $\rho_0 = 10^{18} \text{kg/m}^3$ ，步长 $dr = 10\text{m} = 0.01\text{km}$ （约为中子星总半径的千分之一，基本可以保证计算的精度），求解它的各物理量的径向分布如图 2 所示。

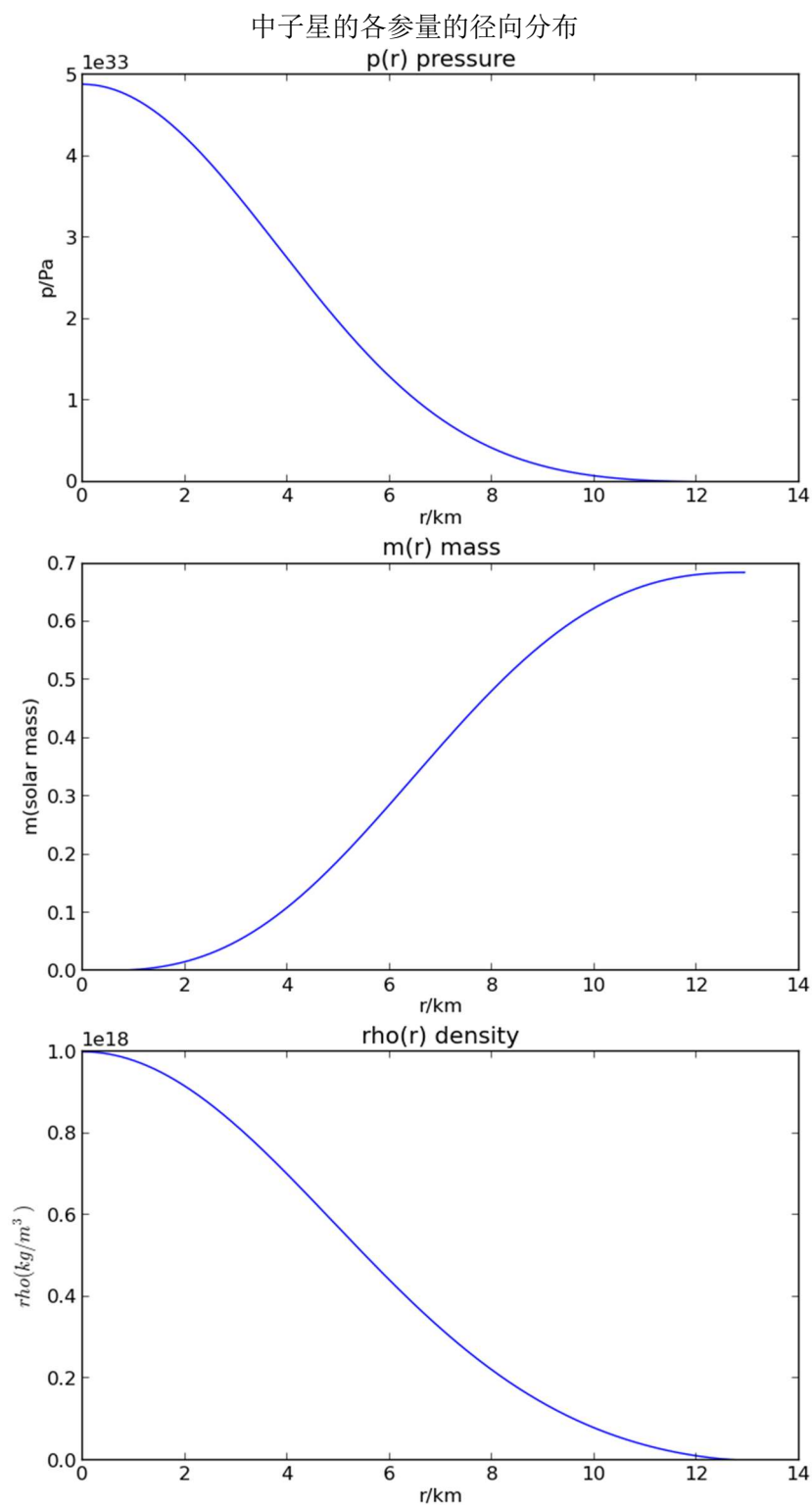


图 2 中子星各参量的径向分布

注：左上角的 10^{33} 代表所得纵坐标数值应乘以 10 的 33 次方

图 2 中半径 r 的单位是 km，质量 m 的数值为太阳质量的倍数，此外其他物理量的单位

均为国际单位，此次计算中大部分采用国际单位制（这样可以带来很多方便，避免了计算前后的冗杂的单位换算和无量纲化等，尽管有些量的数量级超大，而有些量却极小，但是好在python的精度高，并且都用科学计数法表示，有效数字的位数不是很多，不至于溢出或出错，而且实际上真实的物理系统必然是有有限的稳定解，笔者的计算结果也证明了这一点，确实是可以算出正确结果来的），最后还要说明的一点是我终止迭代计算，即确定中子星实际的半径大小的判据是 $\rho(r) = 0$ ，当发现 $\rho(r) \leq 0$ 时，立刻停止迭代计算。

由图2可见，中子星内部的压强和密度分布都是随半径增大而减小，中子星中心处变化率几乎为零，到中间下降最快，之后到边缘处又缓慢降到零。

5、中子星的质量与半径关系

5.1 笔者的结果

利用上面的程序，更进一步，改变中心密度 ρ_0 ，经过大量的计算后把相应的中子星的总质量与半径绘成曲线，我们就可以求解出中子星的质量与半径之间满足的关系。

图3中从左至右相应的初始条件中心密度 ρ_0 依次增大，值得注意的是：为满足平衡条件，中子星的M-R关系必须满足图3，但满足图3的中子星不一定是真实存在的，因为它还要满足稳定平衡的条件的！

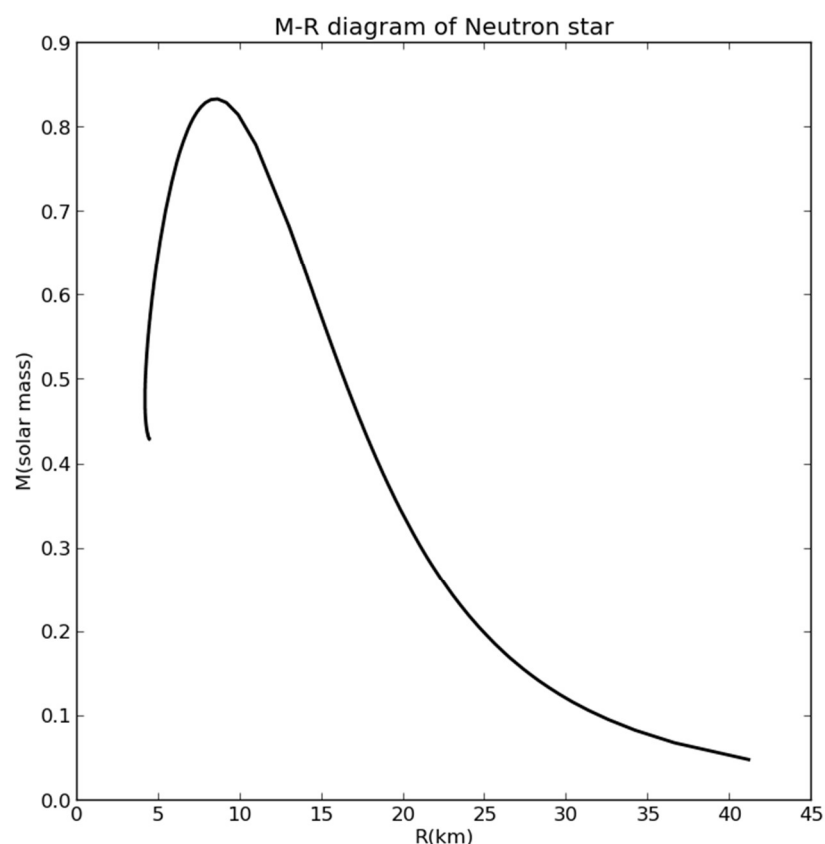


图3 中子星的质量与半径关系

这一单峰状曲线的右半侧中子星半径(或者说体积)随着质量增加而减小,这是正常的;

但左半侧半径（或体积）却随着质量的增加而增加！试想，如果中子星的质量稍微增大，引力也随之增大，只有它的半径减小才能提供更大的压强抵抗增大的引力，然而曲线左侧半径却随着质量的增加而增加，这样的体系是不稳定的，稍有扰动，它会迅速地坍缩或炸开。

因此曲线的峰值处，半径约 10km，质量约 0.85 倍太阳质量，这一点是中子星的极限状态，即是说，在这一模型下，中子星的最小半径约 10km，质量上限约 0.85 倍太阳质量。如果中子星的半径小于这一最小半径或者大于这一质量上限，它们在该模型下都不能稳定地存在。

5.2 Oppenheimer 和 Volkoff 的结果

著名的物理学家 J. R. Oppenheimer 和 G.M. Volkoff 在 1939 年也类似地求解过同样的问题，他们得到结果如图 4，其中 t 是与密度有关的一个量，具体表达式比较复杂，可参见其原文^[4]，他们的得到的结论是中子星的质量上限大约是 0.75 倍太阳质量。

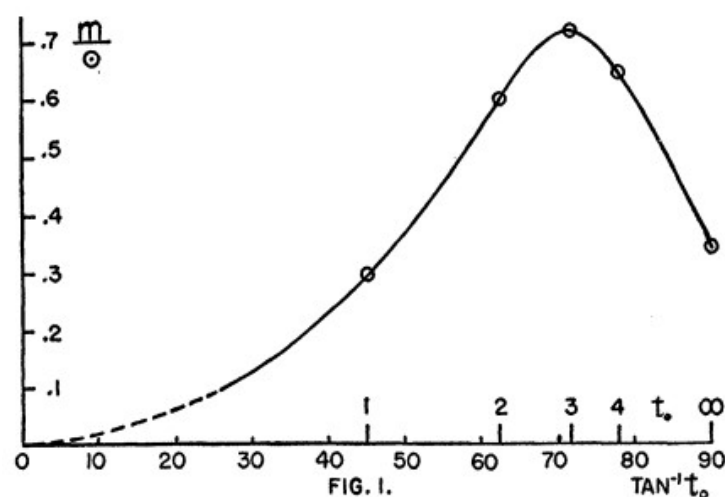


图 4 J. R. Oppenheimer 和 G.M. Volkoff 的结果，引自文献^[4]

5.3 两者间的比较

大体而言，笔者的计算结果与 Oppenheimer 他们的结果还是比较接近的，曲线形状也比较类似，峰值大小也相差不大。

不过虽然说差不多，但确实是有一定的差距，这应该不是简单的随机的计算误差导致的，计算误差不太可能有这么大。但是据笔者分析，笔者的结果与 Oppenheimer 他们的结果之所以有差距，可能是以下原因：

一是他们计算的数据点比较少（见图 4），所计算的最高的数据点可能不是恰好位于峰值位置，导致他们预言的质量上限结果可能有一定偏差。但笔者计算得出的是一串几乎连续的数据，笔者的计算误差（相对误差大约千分之一）绝对远小于 1 倍太阳质量，所以应该是他们的结果存在缺陷，当然这也可以原谅，毕竟他们那时（1939 年）的数值计算水平、速度等，都抵不上现在的计算机，（当时还没有计算机，他们的数值计算也是由人完成的）；

二是他们做这个问题的数值计算时具体的计算方法步骤与笔者的方法步骤不太一样，毕竟是做数值计算，不同的方法之间难免会有些差距也未必可知，由于笔者计算原理上坚持尽量能不作近似的地方就不作近似，所以笔者对自己的计算结果还是有信心的。

可以说，得益于计算机的进步，笔者进行了更精确的计算，更新了 Oppenheimer 他们的当年的计算结果。

6、结论

1、在中子星的这一问题中，由于 TOV 方程和精确的状态方程表达式都极为复杂，完全不作近似地解析求解几乎是不可能的，于是数值解法在这样具有复杂的表达式的问题中就展现出了巨大的威力，它可以保留所有的解析表达式，只作离散化的近似，从而逼近精确解。

2、由于理论公式全部是由基本理论导出，并且该模型对称性较高，比较简单，因此只有一个可调参量中心密度，给定任意中心密度后，各个物理量的径向分布都可以利用该程序求解出来，其结果类似于如图 2。

3、本文还求解了中子星的质量与半径之间的关系，结果如图 3，并预言了中子星的最小半径约 10km，质量上限约 0.85 倍太阳质量，这一结果与 Oppenheimer 等人的结果很接近，而笔者的数值计算更精确，更新了他们当年的结果。

附：程序代码

[NeutronStar.py](#)

注：尽管此处含有超链接，为确保万一，仍然在下方复制粘贴了自己编写的程序代码

```
'''
```

```
Neutron star: pure neutron Fermi gas model
```

```
Author: GUO Xiao
```

```
2013301020099
```

```
SI units
```

```
'''
```

```
import numpy as np
```

```
from pylab import *
```

```
from math import *
```

```
global c, m_n, hbar, lam, G, msun
```

```
hbar=1.05457e-34
```

```
c=2.9792458e8
```

```
m_n=1.675e-27
```

```
lam=hbar/(m_n*c)#lambda
```

```
G=6.67e-11
```

```
msun=1.989e30
```

```
def p_F(rho):
```

```
    pF=hbar*(3*pi**2*rho/m_n)**(1.0/3.0)#1.0! not 1
```

```
    return pF
```

```
def lp(x):
```

```
    l=3.0/8.0*(x*sqrt(1+x**2)*(2*x**2/3.0-1)+asinh(x))
```

```
    return l
```



```

def p_(rho):#pressure   equation of state
    p_id=m_n*c**2/(3*pi**2*lam**3)*lp(p_F(rho)/(m_n*c))
    return p_id

def NS(rho0):
    dr=10#(m)
    r=[0,dr]
    rho=[rho0,0.9999*rho0]#actually drho=0 at r=0, such that avoid infinite
    p=[p_(rho0),p_(0.9999*rho0)]
    m=[0]
    m.append(m[0]+(4*pi*r[1]**2*rho[1])*dr)
    i=1
    #f=open('NS.txt','a')
    while 1:
        dp_drho=(p[i]-p[i-1])/(rho[i]-rho[i-1])
        print 'dp_drho=',dp_drho
        rho.append(rho[i]-
dr/dp_drho*G*(rho[i]*c**2+p[i])*(4*pi*r[i]**3*p[i]+m[i]*c**2)/(r[i]*c**2*(r[i]*c**2-2*m[i]*G)))
        if rho[-1]<=0 :
            break
        p.append(p_(rho[i+1]))
        r.append(r[i]+dr)
        m.append(m[i]+4*pi*r[i+1]**2*rho[i+1]*dr)
        print r[-1],p[-1],m[-1],'rho=',rho[-1]
        i=i+1
    rho.pop(-1)
    #print >> f,'/n'
    r=array(r)/1000.0#km
    m=array(m)/msun
    R=r[-1]
    M=m[-1]
    #print >> f,R,M
    #f.close()
    return [R,M,r,p,m,rho]

NS1=NS(1e18)
r=NS1[2]
p=NS1[3]
m=NS1[4]
rho=NS1[5]
figure(figsize=[8,15])
subplot(311)
plot(r,p)

```

```

title('p(r) pressure')
xlabel('r/km')
ylabel('p/Pa')
subplot(312)
plot(r,m)
title('m(r) mass')
xlabel('r/km')
ylabel('m(solar mass)')
subplot(313)
plot(r,rho)
title('rho(r) density')
xlabel('r/km')
ylabel('$\rho(\text{kg/m}^3)$')
savefig('Neutron star .png')
show()

#R-M diagram
RNS=[]
MNS=[]
for i in range(1,500):
    N=NS(i*2e15)
    RNS.append(N[0])
    MNS.append(N[1])
for i in range(1,500):
    N=NS(i*1e18)
    RNS.append(N[0])
    MNS.append(N[1])

figure(figsize=[8,8])
plot(RNS,MNS,'-',color='black',linewidth=2)
xlabel('R(km)')
ylabel('M(solar mass)')
xlim(0,)
ylim(0,)
title('M-R diagram of Neutron star')
savefig('Neutron Star R-M diagram .png')
show()
-----END-----

```

参考文献

- [1]方励之等.相对论天体物理的基本概念.上海科学技术出版社.1982
- [2]R. R.Silbar, S. Reddy. Neutron stars for undergraduates. American journal of physics.72.892(2004)
- [3] R. M. Wald. General Relativity. The university of Chicago press. 1984

[4]J. R. Oppenheimer, G.M. Volkov. On massive neutron cores. Physical Review.55.374-381(1939)

[5] https://en.wikipedia.org/wiki/Neutron_star

致谢

本文是在与陈洋遥和李明皓同学的合作讨论之后完成的，感谢他们提供的帮助和支持！
但同时也声明除理论上参考过一些文献上的部分内容外，本文的全部核心内容尤其程序的构思、编写、计算结果等系为笔者原创。