Term Project (1) - Orbit of the Sun

023426910029 胡旭凡 2023.11.26

(a) The Sun's orbit properties

$\Omega_g = A - B = 27.2\ km/s/kpc$
$\kappa = \sqrt{(-4B\Omega_g)} = 37\ km/s/kpc$

设相位角为ϕ\
半径：$R_g + A_x \cos\phi = X$ (1)
径向速度：$-\kappa A_x \sin\phi = U_\odot$ (2)
切向速度：$\dot{y} + \Omega_g R_g = V_{LSR} + V_\odot \Rightarrow -2\Omega_g A_x \cos\phi + \Omega_g R_g = V_{LSR} + V_\odot$ (3)

求解得：$R_g = 8.56\ kpc\ \phi = 3.83 = 220°\ A_x = 0.47\ kpc$
所以近日点为 $R_{peri} = R_g - A_x = 8.09\ kpc$，远日点为 $R_{apo} = R_g + A_x = 9.03\ kpc$

(b)Solar orbit integration

首先计算势能的梯度

In[∘]:= $\Phi b[x\_, y\_, z\_] := \dfrac{-Mb}{\sqrt{x^2+y^2+bb^2}}$

-Grad[Φb[x, y, z], {x, y, z}]
梯度

Out[∘]= $\left\{-\dfrac{Mb\,x}{(bb^2+x^2+y^2)^{3/2}}, -\dfrac{Mb\,y}{(bb^2+x^2+y^2)^{3/2}}, 0\right\}$

In[∘]:= $\Phi d[x\_, y\_, z\_] := \dfrac{-Md}{\sqrt{x^2+y^2+\left(ad+\sqrt{z^2+bd^2}\right)^2}}$;

-Grad[Φd[x, y, z], {x, y, z}]
梯度

Out[∘]= $\left\{-\dfrac{Md\,x}{\left(x^2+y^2+\left(ad+\sqrt{bd^2+z^2}\right)^2\right)^{3/2}}, -\dfrac{Md\,y}{\left(x^2+y^2+\left(ad+\sqrt{bd^2+z^2}\right)^2\right)^{3/2}}, -\dfrac{Md\,z\left(ad+\sqrt{bd^2+z^2}\right)}{\sqrt{bd^2+z^2}\left(x^2+y^2+\left(ad+\sqrt{bd^2+z^2}\right)^2\right)^{3/2}}\right\}$

In[∘]:= $\Phi h[x\_, y\_, z\_] := \dfrac{Mh}{ah}\left(\dfrac{1}{\gamma-1}\,\mathrm{Log}\left[\dfrac{1+\left(\sqrt{x^2+y^2+z^2}/ah\right)^{\gamma-1}}{1+(\Lambda/ah)^{\gamma-1}}\right] - \dfrac{(\Lambda/ah)^{\gamma-1}}{1+(\Lambda/ah)^{\gamma-1}}\right)$;
对数

-Grad[Φh[x, y, z], {x, y, z}]
梯度

Out[∘]= $\left\{-\dfrac{Mh\,x\left(\frac{\sqrt{x^2+y^2+z^2}}{ah}\right)^{-1+\gamma}}{ah\,(x^2+y^2+z^2)\left(1+\left(\frac{\sqrt{x^2+y^2+z^2}}{ah}\right)^{-1+\gamma}\right)}, -\dfrac{Mh\,y\left(\frac{\sqrt{x^2+y^2+z^2}}{ah}\right)^{-1+\gamma}}{ah\,(x^2+y^2+z^2)\left(1+\left(\frac{\sqrt{x^2+y^2+z^2}}{ah}\right)^{-1+\gamma}\right)}, -\dfrac{Mh\,z\left(\frac{\sqrt{x^2+y^2+z^2}}{ah}\right)^{-1+\gamma}}{ah\,(x^2+y^2+z^2)\left(1+\left(\frac{\sqrt{x^2+y^2+z^2}}{ah}\right)^{-1+\gamma}\right)}\right\}$

In[∘]:= $3.2408\times10^{-16}\times3.15\times10^{7}\times10^{6}$

Out[∘]= 0.0102085

In[∘]:= $2\pi\,8.5/220$

Out[∘]= 0.242759

源代码由C实现，见附录

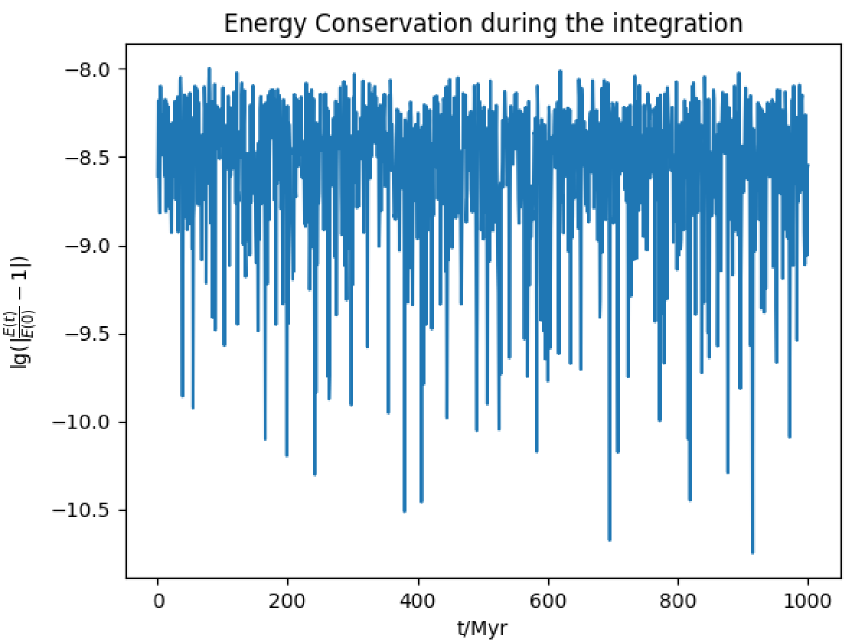
```
import numpy as np
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
```

```
d=np.loadtxt('pot.txt')
d.shape
```

```
(1001, 8)
```

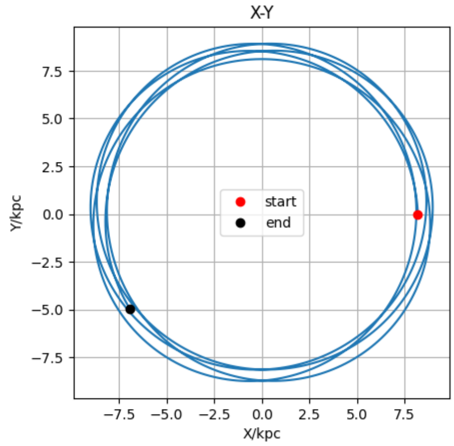
```
H=0.5*(np.power(d[:,4],2)+np.power(d[:,5],2)+np.power(d[:,6],2))+d[:,7]
```

```
plt.plot(d[:,0],np.log10(np.abs(H/H[0]-1)))
plt.ylim()
plt.xlabel('t/Myr')
plt.ylabel(r'$\lg(|\frac{E(t)}{E(0)}-1|)$')
plt.title('Energy Conservation during the integration')
plt.show()
```

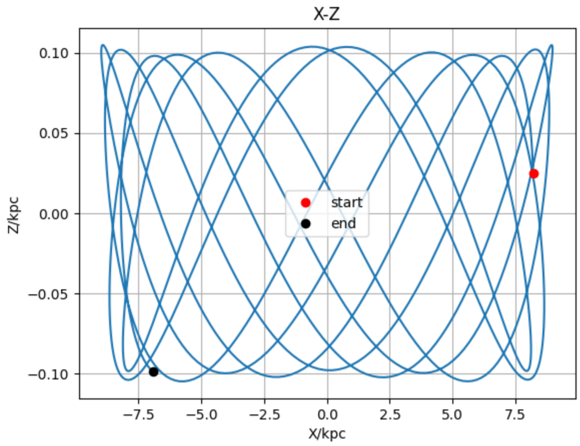


能量波动控制在了10^{-8}之内

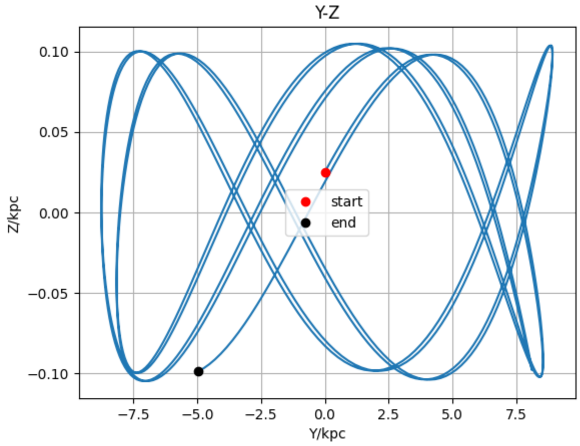
```
plt.plot(d[:,1],d[:,2])
plt.plot(d[0,1],d[0,2],'ro',label='start')
plt.plot(d[-1,1],d[-1,2],'ko',label='end')
plt.axis('scaled')
plt.title('X-Y')
plt.grid()
plt.xlabel('X/kpc')
plt.ylabel('Y/kpc')
plt.legend()
```




```
plt.plot(d[:,1],d[:,3])
plt.plot(d[0,1],d[0,3],'ro',label='start')
plt.plot(d[-1,1],d[-1,3],'ko',label='end')
#plt.axis('scaled')
plt.title('X-Z')
plt.grid()
plt.xlabel('X/kpc')
plt.ylabel('Z/kpc')
plt.legend()
```




```
plt.plot(d[:,2],d[:,3])
plt.plot(d[0,2],d[0,3],'ro',label='start')
plt.plot(d[-1,2],d[-1,3],'ko',label='end')
#plt.axis('scaled')
plt.title('Y-Z')
plt.grid()
plt.xlabel('Y/kpc')
plt.ylabel('Z/kpc')
plt.legend()
```




```python
def f(t,A,omega,phi,C):
    return A*np.sin(omega*t+phi)+C
```

```python
#拟合 R 的变化
r=np.sqrt(np.power(d[:,1],2)+np.power(d[:,2],2))
a0 = 0.47
a1 = 0.037
a2 = 0
a3 = np.mean(r)
p0 = [a0, a1, a2, a3]
para,_=curve_fit(f,d[:,0], r, p0=p0)
print(para)
```

```
[ 0.46870985  0.03968105 -2.16532523  8.59091903]
```

$\kappa = 39.7\ km/s/kpc\ A_x = 0.469\ kpc\ r_g = 8.59\ kpc$

近心点 8.12 kpc 远心点 9.06 kpc

```python
#拟合z方向振动
para,_=curve_fit(f,d[:,0],d[:,3],p0=[0.1,0.072,0,0])
para
```

```
array([ 0.10124885,  0.07314377,  0.33980469, -0.00039523])
```

$\nu = 73.1\ km/s/kpc$

```python
#拟合近似X的振动
para,_=curve_fit(f,d[:,0],d[:,1],p0=[8.5,0.028,np.pi/2,0])
para
```

```
array([8.59696521, 0.02871472, 1.64533089, 0.00913519])
```

$\Omega_g = 28.7\ km/s/kpc$

结果总结如下

	$\Omega_g[km/s/kpc]$	$\kappa[km/s/kpc]$	$\nu[km/s/kpc]$	r_{per}/kpc	r_{apo}
Integration	28.7	39.7	73.1	8.12	9.06
Calculation	27.2	37	-	8.09	9.03
Textbook	28.6	37	70	-	-

可以发现，积分结果略微比计算和课本值偏大

(c) $Z - v_z$ phase space

$E = \frac{J_z^2}{2r^2} + \frac{v_r^2+v_z^2}{2} + \Phi(r,z)\backslash$
在近日（远日）点，$v_r = 0$。由于 κ/ν 通常不为有理数，所以$|\Delta z|$ 没有确定值，我们可以取 $v_z = 0$ 估算上限：

$E = \frac{J_z^2}{2r^2} + \frac{v_r^2+v_z^2}{2} + \Phi(r,z) \approx \frac{J_z^2}{2r^2} + \Phi(r,0) + \frac{1}{2}\Phi_{zz}(r,0)z^2 \Rightarrow z = \sqrt{\frac{2(E-J_z^2/(2r^2)-\Phi(r,0))}{\Phi_{zz}(r,0)}}$

```
1 H[0]
```

```
1 -1594.5241855
```

通过Mathematica计算得知，$|\Delta z|_{per} = 0.108\ kpc, |\Delta z|_{apo} = 0.0867\ kpc$

In[1]:= $\Phi[r\_, z\_] := \frac{-\text{Mb}}{\sqrt{r^2+z^2+\text{bb}^2}} + \frac{-\text{Md}}{\sqrt{r^2+\left(\text{ad}+\sqrt{z^2+\text{bd}^2}\right)^2}} + \frac{\text{Mh}}{\text{ah}}\left(\frac{1}{\gamma-1}\text{Log}\left[\frac{1+\left(\sqrt{r^2+z^2}/\text{ah}\right)^{\gamma-1}}{1+(\Lambda/\text{ah})^{\gamma-1}}\right] - \frac{(\Lambda/\text{ah})^{\gamma-1}}{1+(\Lambda/\text{ah})^{\gamma-1}}\right);$

```
Mb = 409;
Md = 2856;
Mh = 1018;
bb = 0.23;
ad = 4.22;
bd = 0.292;
ah = 2.562;
Λ = 200;
γ = 2;
```

In[11]:= $25.224^2/2$

Out[11]= 318.125

In[12]:= $\text{zper} = \sqrt{\frac{2\left(-1594.5241855 - 318.125088\text{`} \times 8.2^2/8.12^2 - \Phi[8.12, 0]\right)}{\text{D}[\Phi[r, z], \{z, 2\}]\ /.\ \{r \to 8.12, z \to 0\}}}$

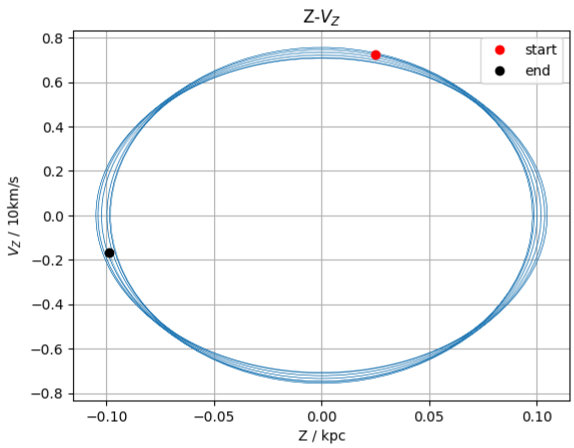
Out[12]= 0.108156

In[14]:= $\text{zapo} = \sqrt{\frac{2\left(-1594.5241855 - 318.125088\text{`} \times 8.2^2/9.06^2 - \Phi[9.06, 0]\right)}{\text{D}[\Phi[r, z], \{z, 2\}]\ /.\ \{r \to 9.06, z \to 0\}}}$

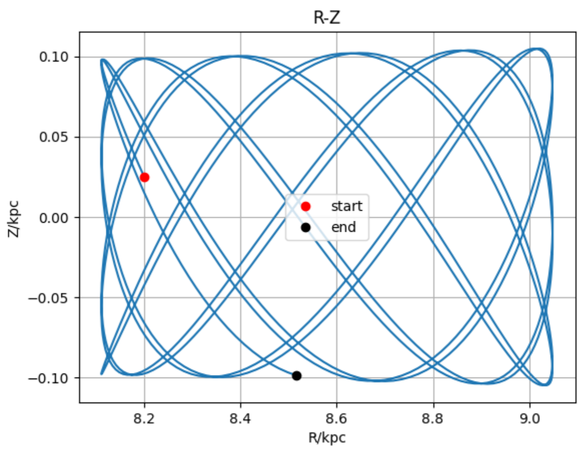
Out[14]= 0.0867166

(d) $Z - v_z$ phase space

```python
plt.plot(d[:,3],d[:,6],lw=0.3)
plt.plot(d[0,3],d[0,6],'ro',label='start')
plt.plot(d[-1,3],d[-1,6],'ko',label='end')
#plt.axis('scaled')
plt.title(r'Z-$V_Z$')
plt.grid()
plt.xlabel('Z / kpc')
plt.ylabel(r'$V_Z$ / 10km/s')
plt.legend()
```




```python
r=np.sqrt(np.power(d[:,1],2)+np.power(d[:,2],2))
plt.plot(r,d[:,3])
plt.plot(r[0],d[0,3],'ro',label='start')
plt.plot(r[-1],d[-1,3],'ko',label='end')
#plt.axis('scaled')
plt.title(r'R-Z')
plt.grid()
plt.xlabel('R/kpc')
plt.ylabel(r'Z/kpc')
plt.legend()
```



从图中可以看出，$|\Delta z|_{per}$，$|\Delta z|_{apo}$ 与计算值接近。

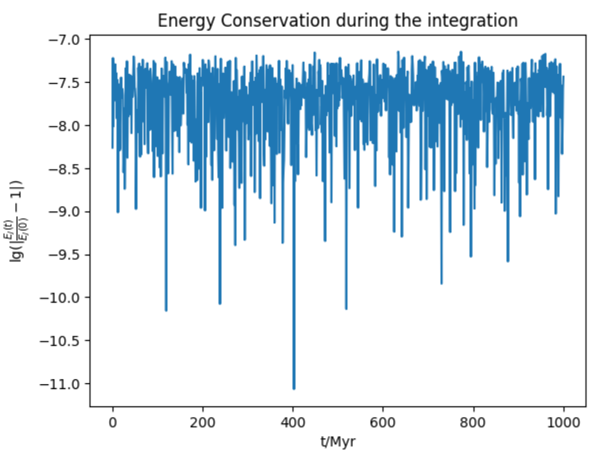
(e) rotational bar

```python
d2=np.loadtxt('pot2.txt')
d2.shape
```

```
(1000, 8)
```

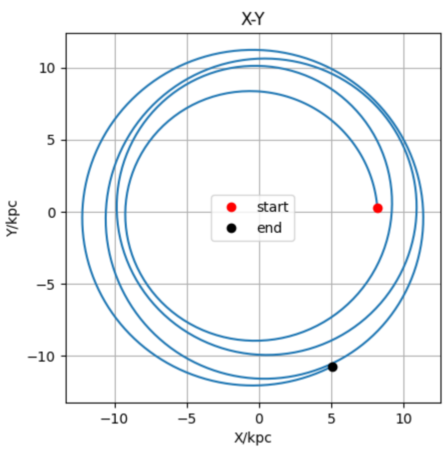
```
Hj=0.5*(np.power(d[:,4],2)+np.power(d[:,5],2)+np.power(d[:,6],2))+d[:,7]-40*
(d[:,1]*d[:,5]-d[:,2]*d[:,4])
```

```
plt.plot(d[:,0],np.log10(np.abs(Hj/Hj[0]-1)))
plt.ylim()
plt.xlabel('t/Myr')
plt.ylabel(r'$\lg(|\frac{E_J(t)}{E_J(0)}-1|)$')
plt.title('Energy Conservation during the integration')
plt.show()
```

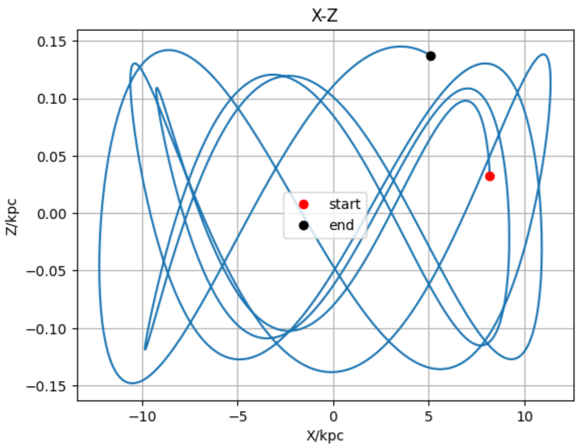


能量波动控制在了10^{-7}之内

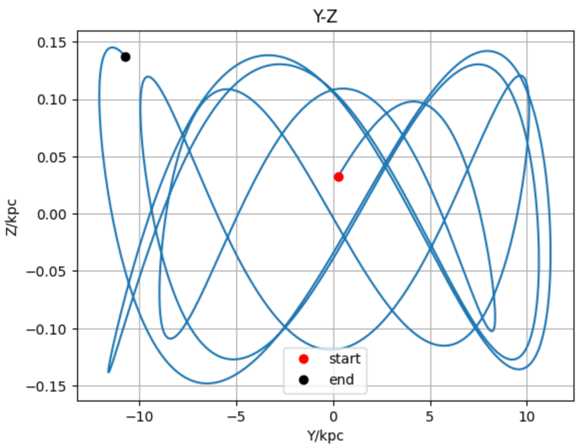
```
plt.plot(d2[:,1],d2[:,2])
plt.plot(d2[0,1],d2[0,2],'ro',label='start')
plt.plot(d2[-1,1],d2[-1,2],'ko',label='end')
plt.axis('scaled')
plt.title('X-Y')
plt.grid()
plt.xlabel('X/kpc')
plt.ylabel('Y/kpc')
plt.legend()
```



```python
plt.plot(d2[:,1],d2[:,3])
plt.plot(d2[0,1],d2[0,3],'ro',label='start')
plt.plot(d2[-1,1],d2[-1,3],'ko',label='end')
#plt.axis('scaled')
plt.title('X-Z')
plt.grid()
plt.xlabel('X/kpc')
plt.ylabel('Z/kpc')
plt.legend()
```




```python
plt.plot(d2[:,2],d2[:,3])
plt.plot(d2[0,2],d2[0,3],'ro',label='start')
plt.plot(d2[-1,2],d2[-1,3],'ko',label='end')
#plt.axis('scaled')
plt.title('Y-Z')
plt.grid()
plt.xlabel('Y/kpc')
plt.ylabel('Z/kpc')
plt.legend()
```



此时的轨迹是原本的变种，振动幅度增加，同时轨迹变得更加没有规律。

Appendix: source code

```c
#include<stdio.h>
#include<math.h>
#include<memory.h>

```

```
#define Mb 409.0  //M_gal=2.235*10^7
#define Md 2856.0
#define Mh 1018.0
#define bb 0.23 //kpc
#define ad 4.22
#define bd 0.292
#define ah 2.562
#define L  200.0
#define gamma 2.0

#define vf 0.0102202 //将速度从 10 km/s -> kpc/Myr
#define step 1000//写入间隔步数

void bar(double &potb,double &fxb,double &fyb,double &fzb,double x,double y,double z){
    potb=-Mb/sqrt(pow(x,2.0)+pow(y,2.0)+pow(z,2.0)+pow(bb,2.0));
    double com=-Mb/pow(pow(x,2.0)+pow(y,2.0)+pow(z,2.0)+pow(bb,2.0),1.5);
    fxb=x*com;
    fyb=y*com;
    fzb=z*com;
}

void disk(double &potd,double &fxd,double &fyd,double &fzd,double x,double y,double z){
    potd=-Md/sqrt(pow(x,2.0)+pow(y,2.0)+pow(ad+sqrt(pow(z,2.0)+pow(bd,2.0)),2.0));
    double com=-Md/pow(pow(x,2.0)+pow(y,2.0)+pow(ad+sqrt(pow(z,2.0)+pow(bd,2.0)),2.0),1.5);
    fxd=x*com;
    fyd=y*com;
    fzd=z*com*(ad+sqrt(pow(bd,2.0)+pow(z,2.0)))/sqrt(pow(bd,2.0)+pow(z,2.0));
}

void halo(double &poth,double &fxh,double &fyh,double &fzh,double x,double y,double z){
    poth=Mh/ah*(1.0/(gamma-1.0)*log((1.0+pow(sqrt(pow(x,2.0)+pow(y,2.0)+pow(z,2.0))/ah,gamma-1.0))/(1.0+pow(L/ah,gamma-1.0)))-pow(L/ah,gamma-1.0)/(1.0+pow(L/ah,gamma-1)));
    double com=-Mh/ah*pow(sqrt(pow(x,2.0)+pow(y,2.0)+pow(z,2.0))/ah,gamma-1.0)/(pow(x,2.0)+pow(y,2.0)+pow(z,2.0))/(1.0+pow(sqrt(pow(x,2.0)+pow(y,2.0)+pow(z,2.0))/ah,gamma-1.0));
    fxh=com*x;
    fyh=com*y;
    fzh=com*z;
}

void integrator(double h,int n,double d[][8]){//Leapfrog integrator
    int i,j,k;
    FILE *fp;
    double potb,potd,poth,potf;
    double fxb,fxd,fxh,fxf;
    double fyb,fyd,fyh,fyf;
    double fzb,fzd,fzh,fzf;
    //首先计算初始势
    bar(potb,fxb,fyb,fzb,d[0][1],d[0][2],d[0][3]);
```

```
    disk(potd,fxd,fyd,fzd,d[0][1],d[0][2],d[0][3]);
    halo(poth,fxh,fyh,fzh,d[0][1],d[0][2],d[0][3]);
    d[0][7]=potb+potd+poth;
    //写入第一行数据
    fp=fopen("pot.txt","w");
    for(i=0;i<8;i++){
        fprintf(fp,"%lf ",d[0][i]);
    }
    fprintf(fp,"\n");
    fclose(fp);
    //开始演化
    for(j=0;j<n/10000;j++){
        for(i=1;i<10001;i++){
            d[i][0]=d[i-1][0]+h;
            d[i][1]=d[i-1][1]+d[i-1][4]*h*vf+0.5*pow(h*vf,2.0)*
(fxb+fxd+fxh);
            d[i][2]=d[i-1][2]+d[i-1][5]*h*vf+0.5*pow(h*vf,2.0)*
(fyb+fyd+fyh);
            d[i][3]=d[i-1][3]+d[i-1][6]*h*vf+0.5*pow(h*vf,2.0)*
(fzb+fzd+fzh);
            d[i][4]=d[i-1][4]+0.5*h*vf*(fxb+fxd+fxh);
            d[i][5]=d[i-1][5]+0.5*h*vf*(fyb+fyd+fyh);
            d[i][6]=d[i-1][6]+0.5*h*vf*(fzb+fzd+fzh);

            bar(potb,fxb,fyb,fzb,d[i][1],d[i][2],d[i][3]);
            disk(potd,fxd,fyd,fzd,d[i][1],d[i][2],d[i][3]);
            halo(poth,fxh,fyh,fzh,d[i][1],d[i][2],d[i][3]);
            d[i][4]=d[i][4]+0.5*h*vf*(fxb+fxd+fxh);
            d[i][5]=d[i][5]+0.5*h*vf*(fyb+fyd+fyh);
            d[i][6]=d[i][6]+0.5*h*vf*(fzb+fzd+fzh);
            d[i][7]=potb+potd+poth;
            if(i%1000==999){
                printf("step=%d\n",i+10000*j+1);
            }
        }
        //每10000行写入一次数据
        fp=fopen("pot.txt","a+");
        for(i=step;i<10001;i=i+step){
            for(k=0;k<8;k++){
                fprintf(fp,"%lf ",d[i][k]);
            }
            fprintf(fp,"\n");
        }
        fclose(fp);
        //写完后将数组复原
        for(i=0;i<8;i++){
            d[0][i]=d[10000][i];
        }
    }
}

int main(){
    double h;
    int n;

    h=0.001;//时间间隔，以Myr为单位
    n=(int)1000/h;
    double d[10001][8];//t,x,y,z,vx,vy,vz,pot
```

```
    memset(d,0.0,sizeof(d));
    //初始化
    d[0][0]=0.0;d[0][1]=8.2;d[0][2]=0.0;d[0][3]=0.025;
    d[0][4]=-1.11;d[0][5]=25.224;d[0][6]=0.725;

    integrator(h,n,d);
    return 0;
}
```