

Planetêre Beweging

Abrie Greeff
13557343

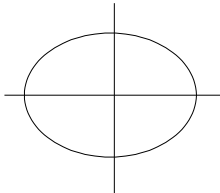
In hierdie deel van toepassings van Numeriese Metodes gaan sonnestelsels ondersoek word. Meer spesifiek gaan gekyk word na die beweging van 'n liggaam om 'n groter massa (bv. Die son) en die invloed van verskillende faktore op die akkuraatheid van die simulaties.

Eerstens is die Euler-Cromer tegniek gebruik en later vergelyk met Euler se metode. Vir die grootste deel van die toepassing is die 4e orde Runge-Kutta metode gebruik.

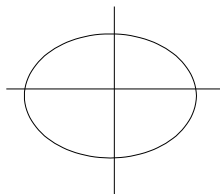
Voor die simulaties kan begin moet 'n paar fundamentele stellings en feite beskou word.

Kepler se wette van planetêre beweging

1. Alle planete beweeg in elliptiese bane, met die son by een fokuspunt.



2. Die lyn wat 'n planeet met die son verbind dek gelyke areas in gelyke tyd inkremente.



3. Vir periode = T en hoof-as lengte = a is $\frac{T^3}{a^3}$ konstant vir alle planete in dieselfde sonnestelsel.

Die periodes van planete (tyd om een omwenteling te maak) is proporsioneel aan die hoof-as lengte van hul bane tot die mag $\frac{3}{2}$

Newton se universele gravitasiewet

Vir 'n liggaam (massa m) wat beweeg om 'n liggaam (massa M), word die krag wat m na M aantrek (F_G) gegee deur

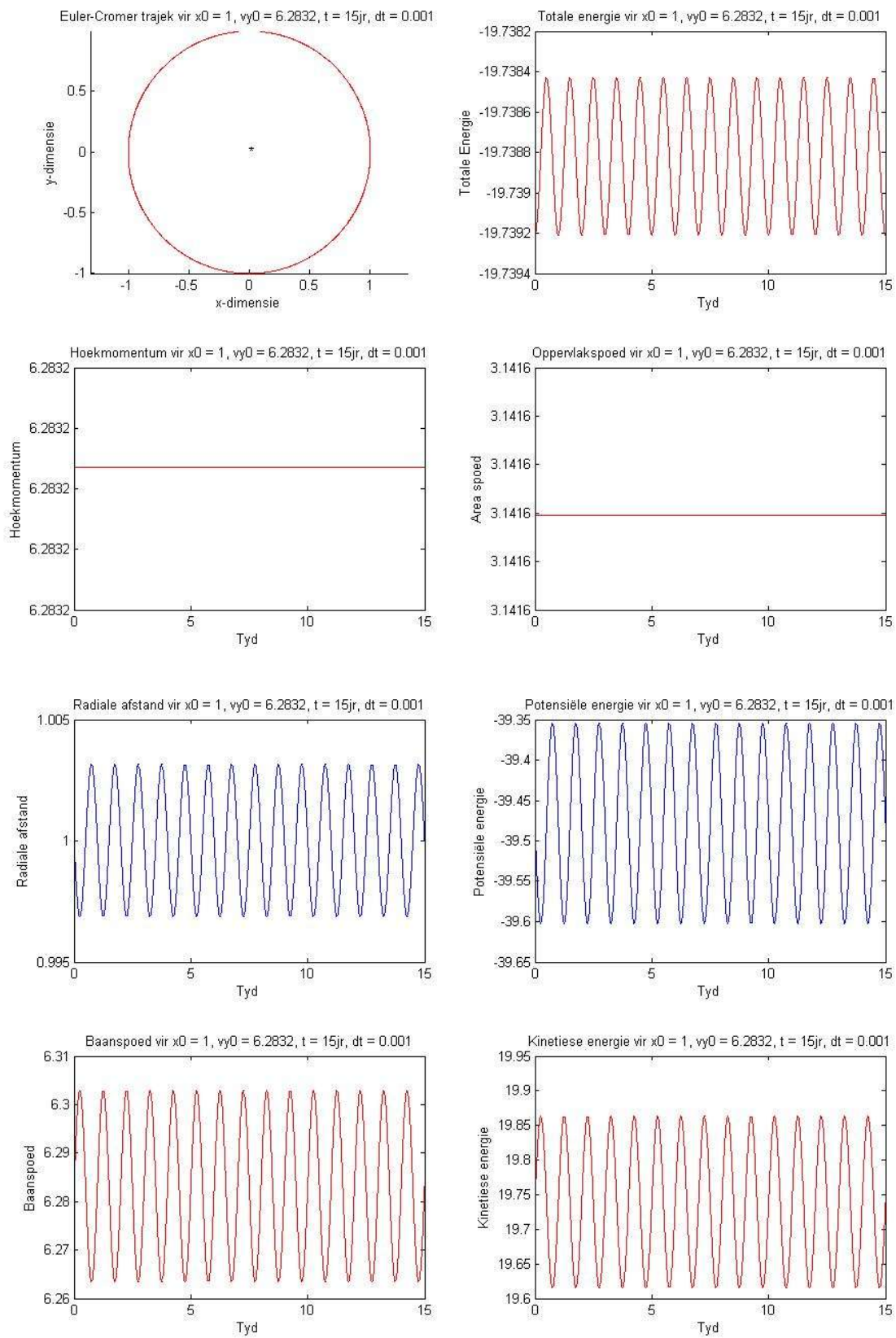
$$F_G = \frac{GMm}{r^2}$$

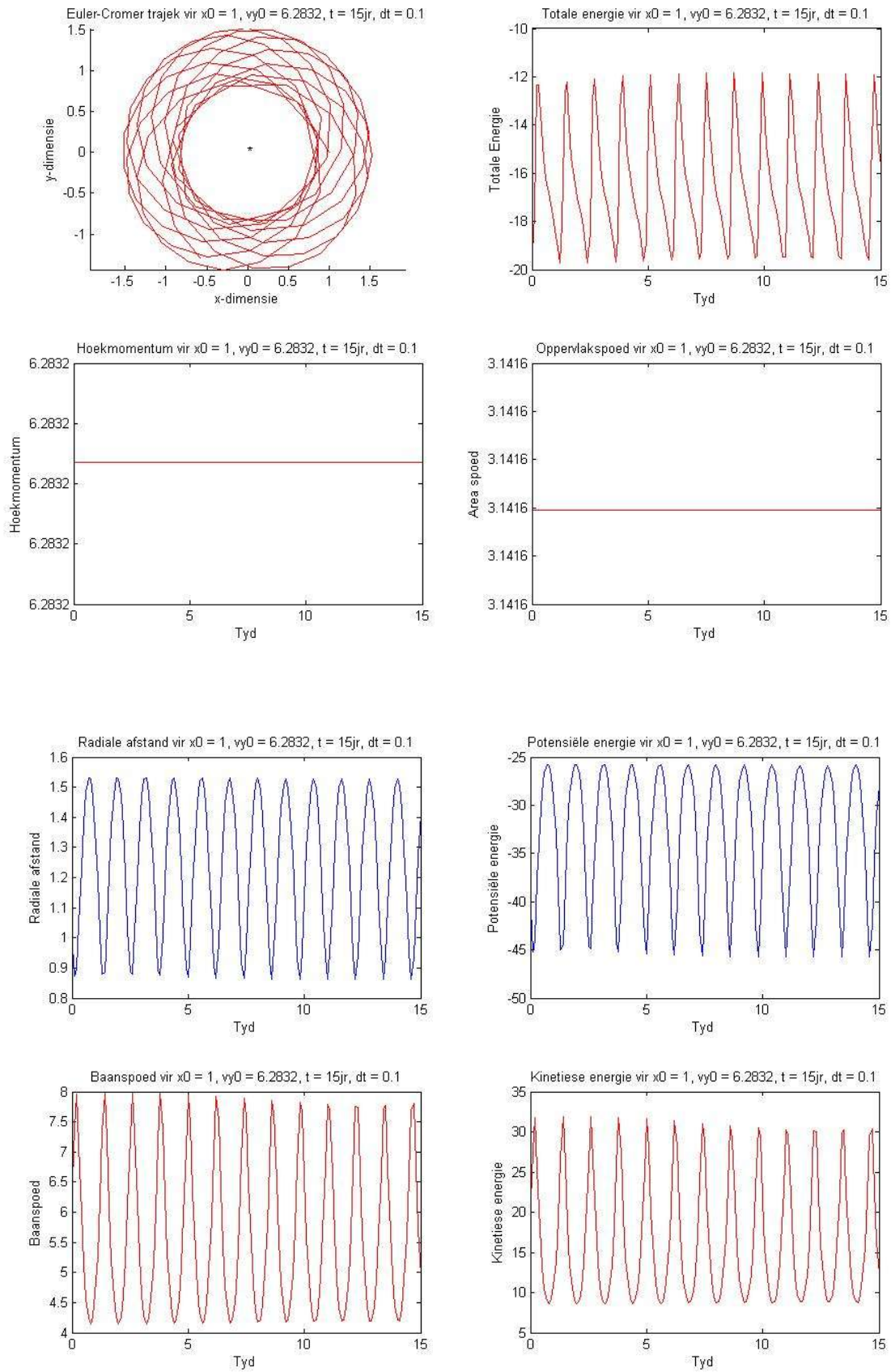
G : Gravitasiekonstante ($6.672 \times 10^{-11} \text{ m}^3 \text{ kg}^{-1} \text{ s}^{-2}$)

r : Die afstand tussen M en m .

Taak1:

Wys dat vir simulaties van die aarde se beweging om die son tydsteppe groter as ongeveer 0.01 jaar, lei tot trajekte wat nie stabiel is nie.





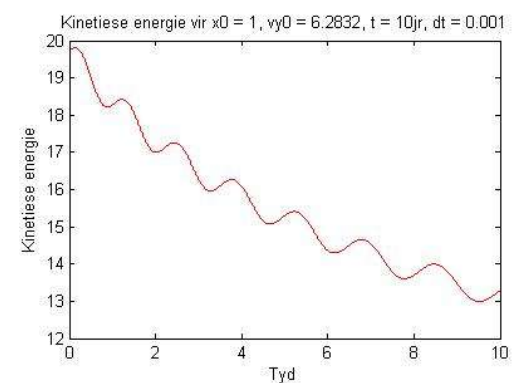
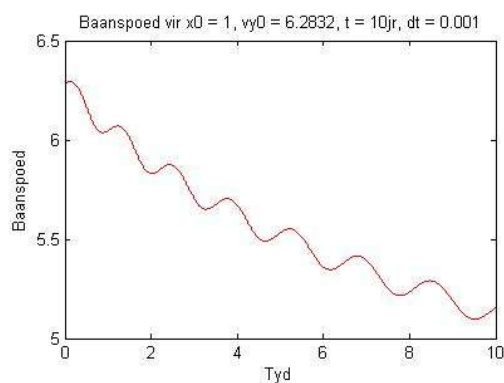
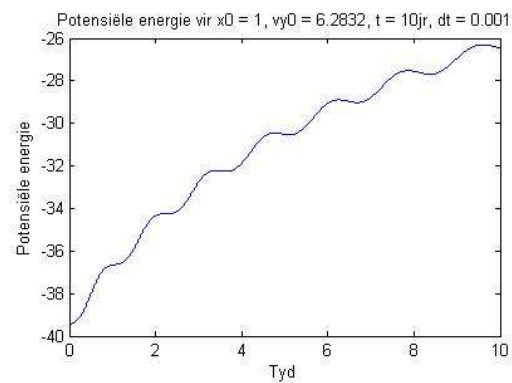
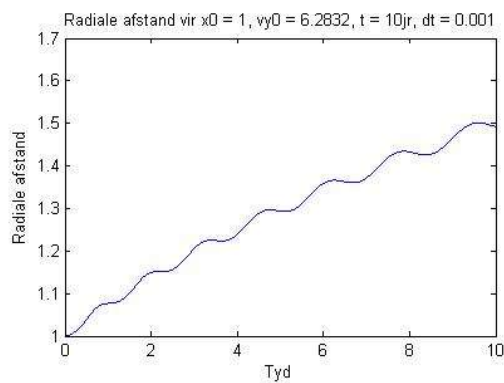
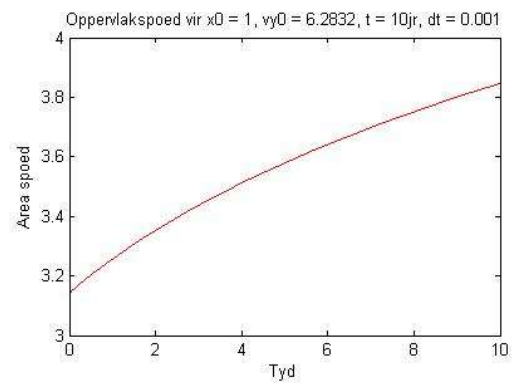
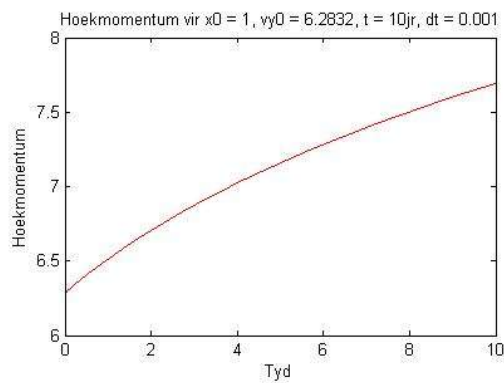
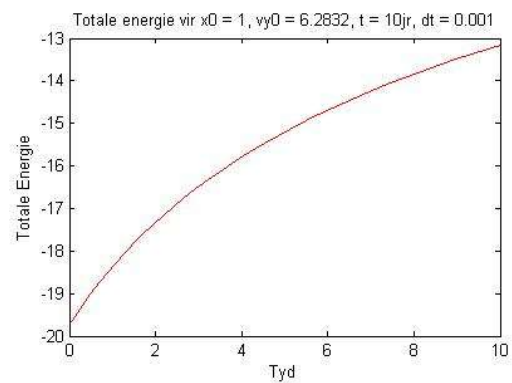
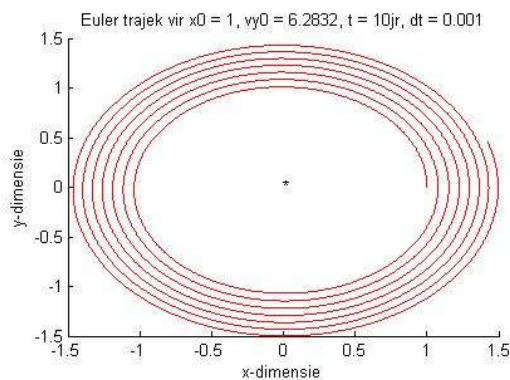
Ek het gevind soos die tyd inkrement groter word is die simulatie van die aarde se baan om die son nie meer stabiel is nie

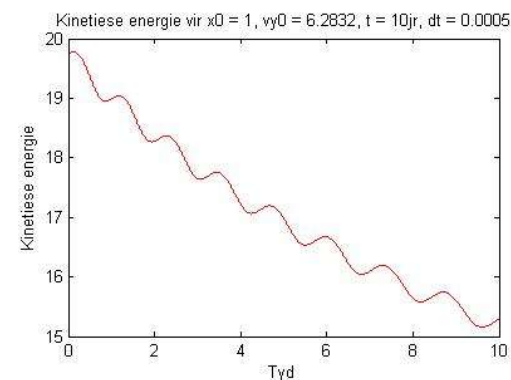
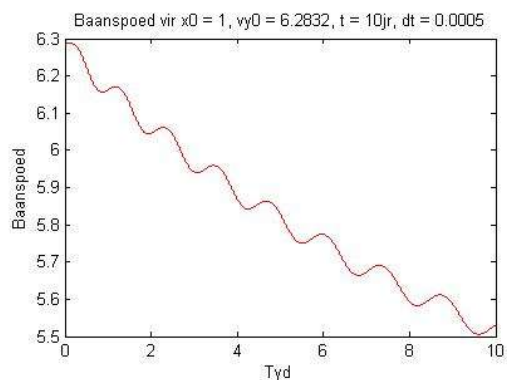
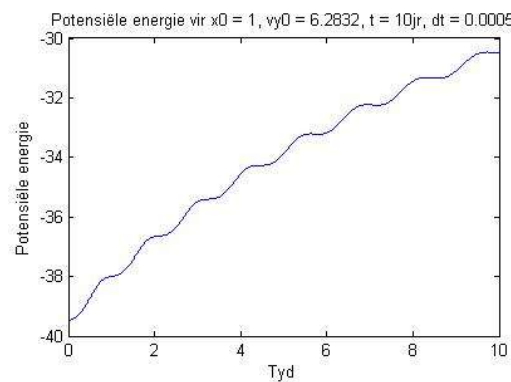
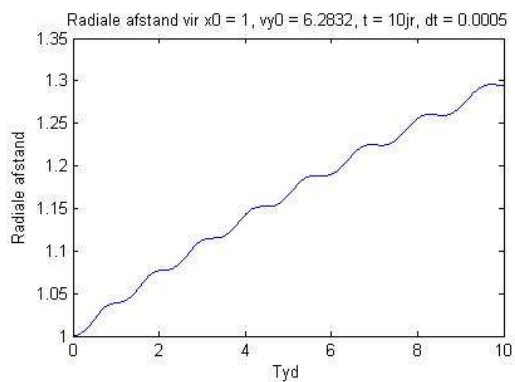
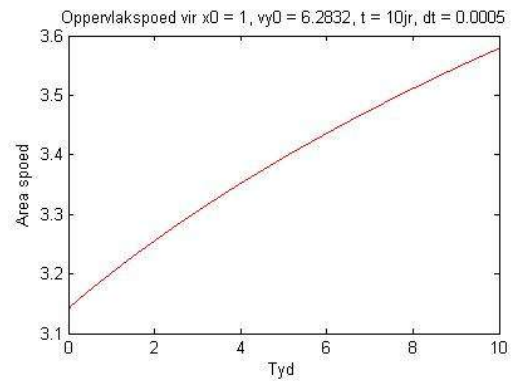
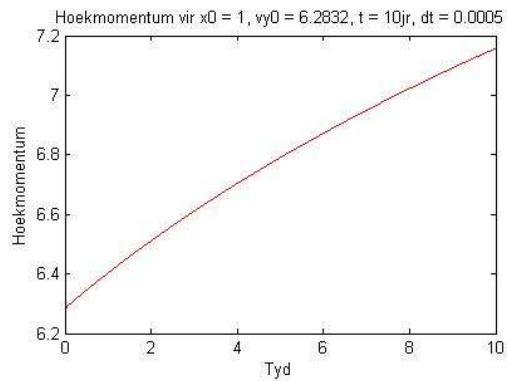
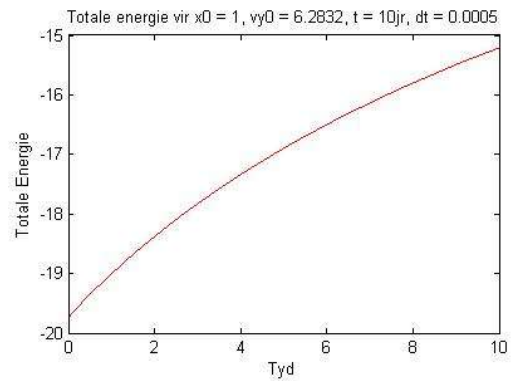
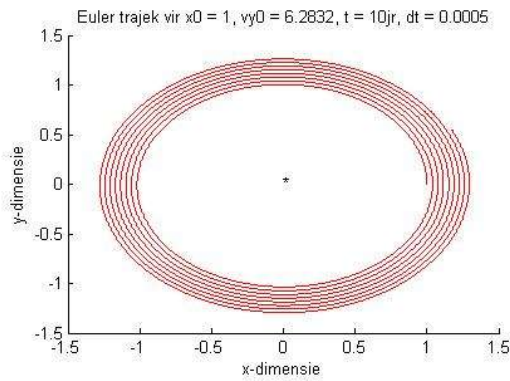
Die grafieke van die kinetiese-, potensiele- en totale meganiese energie toon nie-tipiese gedrag nie. Om die fout maklik te sien moet net na die grafiek van die totale meganiese energie gekyk te word. As die Euler-Cromer tegniek 'n eksakte oplossing gelever het sou dit 'n konstante waarde gelever het. In hierdie geval sien ons dat fluktuasies voorkom met die groter tydsteppe. Ons kan wel sê dat vir $\Delta t = 0.001$ kan ons benaderings maak van die aarde se baan. Oosilering kom nog steeds voor agv die Euler-Cromer metode, maar dis nie meer so grof nie.

Taak2:

Wys dat die Euler benadering altyd lei tot 'n weg-spiralende trajek hoe klein die tydinterval Δt ook al gekies word.

Ek het die beweging van die Aarde om die son gesimuleer met behulp van die Euler tegniek.





Vir $dt = 0.0005$ vind ek dat die Aarde uitwaarts beweeg vanaf die Son. Die radiale afstand van die Aarde word groter met tydsverloop, agv die spiraal vorm van die Aarde se baan.

Die x en y komponente van die Aarde se snelheid ossileer beide met konstante amplitudes soos verwag word. Die snelheid (v), egter, word kleiner met tydsverloop. Die enigste verduideliking hiervoor is dat die afstand wat deur die Aarde afgelê word, groter word. Dit is inderdaad wat hier

gebeur.

Die kinetiese energie daal ook heelwat (agv daling in snelheid). Die potensiele energie styg egter met verloop van tyd (agv styging in die waarde van die radiale afstand). Die laaste fout wat teen Euler tel is dat die totale meganiese energie styg met tyd. Maw hoe langer die Aarde om die Son wentel hoe meer energie kry hy. Daar is geen eksterne kragte wat inwerk teen die son nie en dus is dit onmoontlik vir die energie om te vermeerder. Die rede hiervoor is weer Euler se tegniek wat 'n fout met elke berekening in bring.

Taak3:

Verifieer Kepler se tweede wet vir 'n elliptiese baan: die verbindingslyn tussen die son en planeet vee oor gelyke grootte areas in gelyke tydintervalle, m.a.w. die oppervlakspoed is konstant. Toets ook kwantitatief dat die baan inderdaad ellipties is.

Kepler 2: Die lyn wat die planeet en sy son verbind, dek gelyke areas in gelyke tydinkremente.

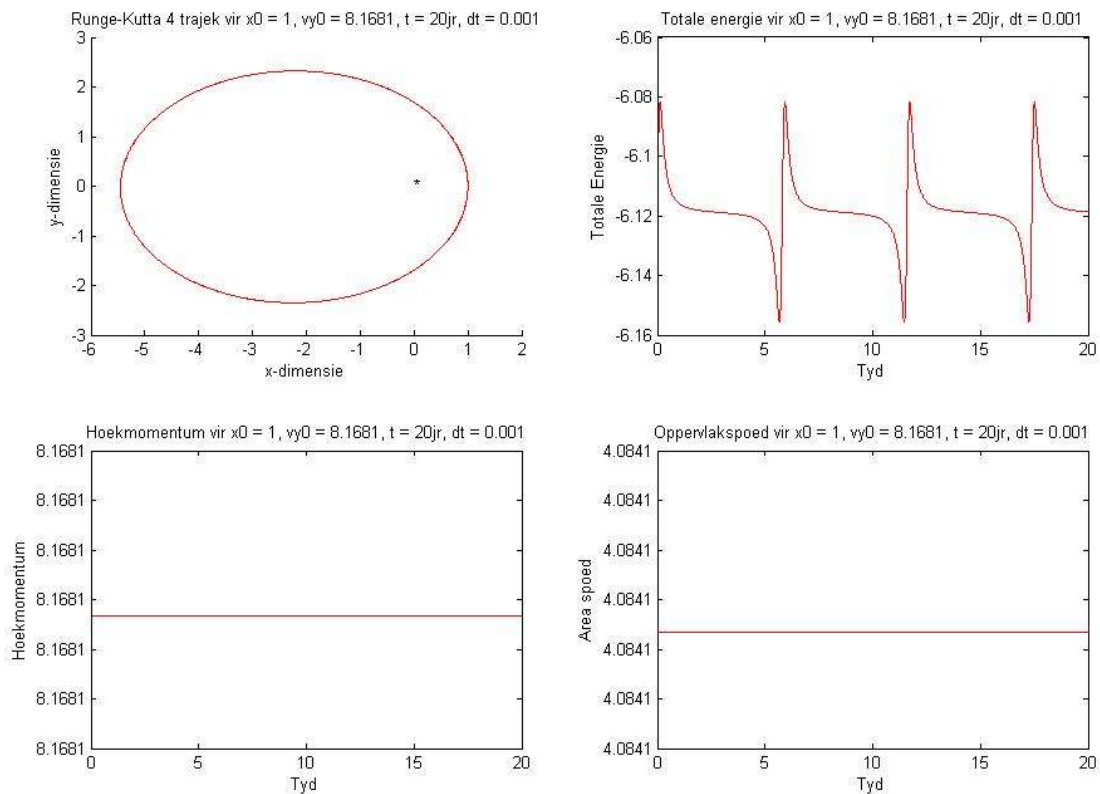
Ek het die area wat die lyn r dek per dt vir elke tydstep uitgerek deur die vergelyking

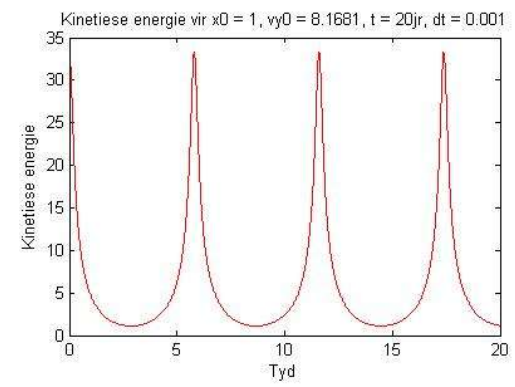
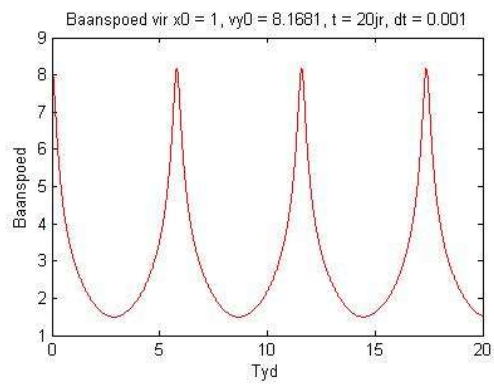
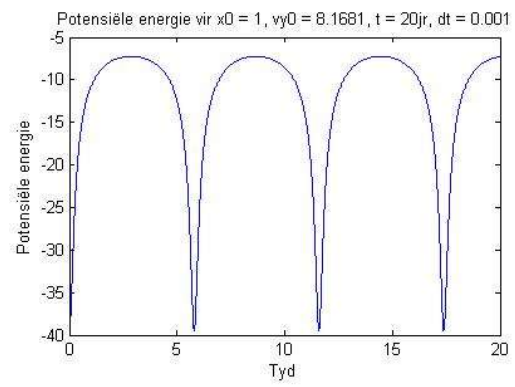
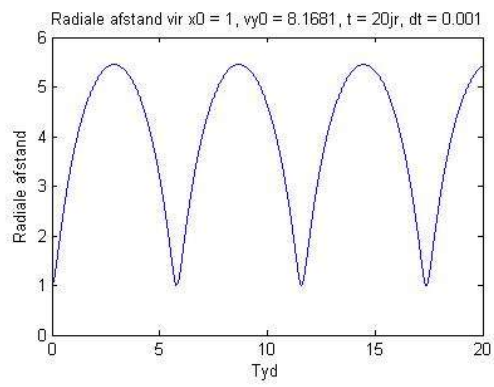
$$A_i = \frac{|x_i(y_{i+1} - y_i) - y_i(x_{i+1} - x_i)|}{2}$$

Dit word dan uitgerek vir elke twee punte op die ellips: (x_i, y_i) en

$$(x_{i+1}, y_{i+1}) \forall i \in (0, \frac{\text{tydperk}}{dt})$$

Volgens Kepler moet A dan konstant bly vir elke waarde van i, dws oor die hele tydperk. Soos die grafiek toon bly A (die area wat die lyn r dek in een tydstep) konstant behalwe vir 'n baie klein afwyking by $t=0$ tot $t \approx 1.1$ en $t \approx 4.7$ tot $t \approx 7.2$. Hierdie afwykings kan toegeskryf word aan die fout wat Runge-Kutta maak. Dus geld Kepler se tweede wet wel vir hierdie simulatie.





Taak4:

Ondersoek die trajek van Halley se komeet. Dit het 'n elliptiese baan met 'n periode van 76 JR en 'n perihelion van 0.59 AE. Gebruik probeer-en-tref tegnieke om sy maksimum baanspoed en aphelion afstand te bepaal. Hoe vergelyk dit met die baan van Pluto?

Die volgende parameters is ingevoer om die stelsel te simuleer.

Massa van die komeet = 1

$$x_0 = 0.59 \text{ AE}, \quad y_0 = 0 \text{ AE}$$

tydperk = 76jr.

Ek het begin met beginsnelheid $v_{x0} = 0$ en $v_{y0} = 2 \times \pi$. Die komeet se beginsnelheid was te klein en het 'n paar keer om die son beweeg in 76 jaar, maar die baan was heelwat kleiner as wat dit moet wees.

Ek het v_{y0} se waarde aanhoudend vermeerder tot by $v_{y0} = 3.65 \pi$ waar die verskil tussen 'n volledige en onvolledige baan haarfyn geraak het. Ek het toe die aantal desimale syfers vermeerder, deur steeds die probeer-en-tref tegniek te gebruik. Op die ou end het ek tot stilstand gekom by $v_{y0} = 3.65191027 \pi \approx 11.472814$.

Om verder die korrektheid van hierdie benadering te toets het ek v_{y0} gaan bereken vanaf die analitiese uitdrukkings:

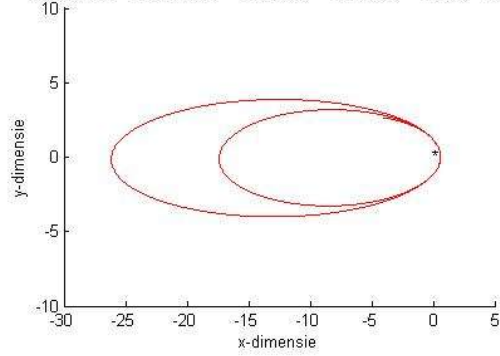
$$\text{Periode: } P = \frac{2 \times \pi}{\sqrt{GM}} a^{\frac{3}{2}}$$

$$\text{Ellips snelheid: } v_y = \sqrt{GM \times \left(\frac{2}{r_{\min}} - \frac{1}{a} \right)}$$

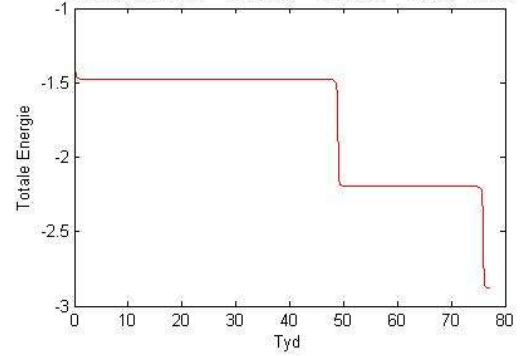
Die vergelyking van die periode is herskryf om a te bereken omdat ons $P = 76$ en $GM = 4 \pi^2$ reeds het. Uit hierdie vergelykings kry ons dat $v_{y0} = 11.47278665942302$ wat ooreenstem met die waarde wat ek met die probeer-en-tref tegniek gekry het.

Vir hierdie beginsnelheid het die ellips nog steeds nie 'n volledige elliptiese baan nie, maar dit kan toegeskryf word aan die tydinkrement wat te groot is vir die Runge-Kutta benadering. 'n Manier om dit te verbeter is die gebruik van 'n 'adaptive' Runge-Kutta metode wat die tydinkremente sal aanpas by berekeninge waar die punte baie verander en 'n groter kans vir 'n fout sal voorkom.

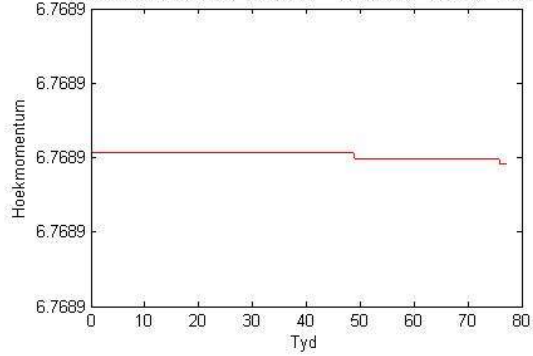
Runge-Kutta 4 trajek vir $x_0 = 0.59$, $y_0 = 11.4728$, $t = 77jr$, $dt = 0.001$



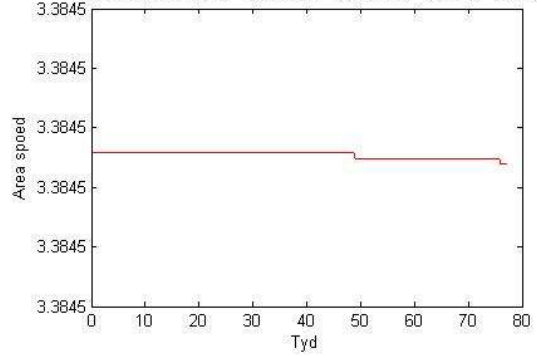
Totale energie vir $x_0 = 0.59$, $y_0 = 11.4728$, $t = 77jr$, $dt = 0.001$



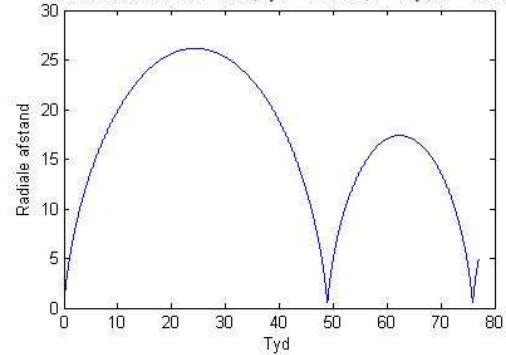
Hoekmomentum vir $x_0 = 0.59$, $y_0 = 11.4728$, $t = 77jr$, $dt = 0.001$



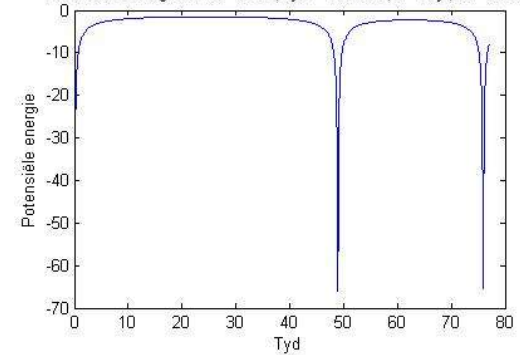
Oppervlakspeed vir $x_0 = 0.59$, $y_0 = 11.4728$, $t = 77jr$, $dt = 0.001$



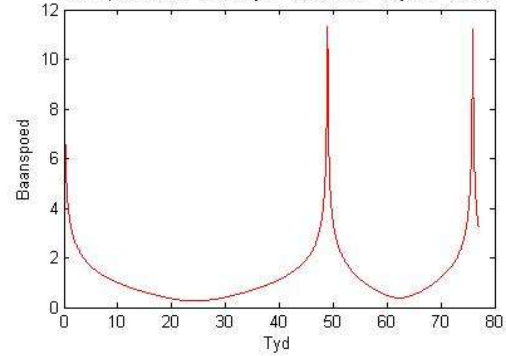
Radiale afstand vir $x_0 = 0.59$, $y_0 = 11.4728$, $t = 77jr$, $dt = 0.001$



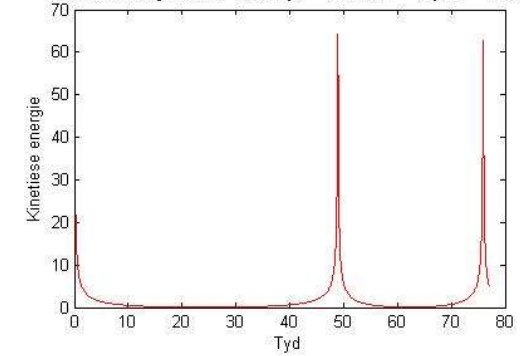
Potensiële energie vir $x_0 = 0.59$, $y_0 = 11.4728$, $t = 77jr$, $dt = 0.001$



Baanspoed vir $x_0 = 0.59$, $y_0 = 11.4728$, $t = 77jr$, $dt = 0.001$



Kinetiese energie vir $x_0 = 0.59$, $y_0 = 11.4728$, $t = 77jr$, $dt = 0.001$



Taak5:

Bepaal die kinetiese -, potensiële -, en totale meganiese- energie van die sisteem, asook die hoekmomentum. Toon dat in die geval van 'n sirkelvormige baan bly al hierdie groothede konstant, terwyl die kinetiese en potensiële energie vir 'n elliptiese baan komplementêr ossilleer, sodat hul som, die totale meganiese energie steeds tydkonstant bly. Bewys ook uit die analitiese model dat dit verwag word vir 'n konserwatiewe kragveld of enige sentraalkrag.

Ek het die stelsel met $dt = 0.001$ eers gesimuleer met die parameters

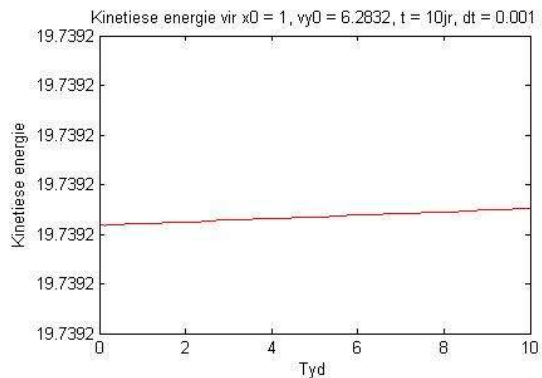
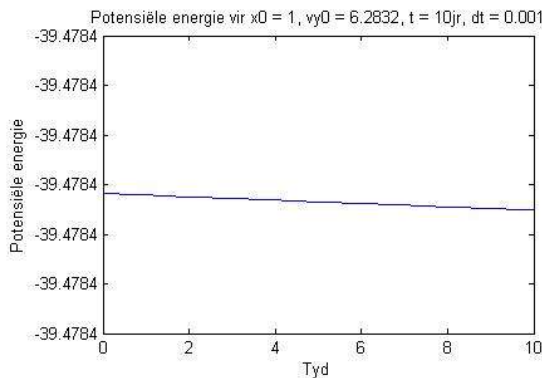
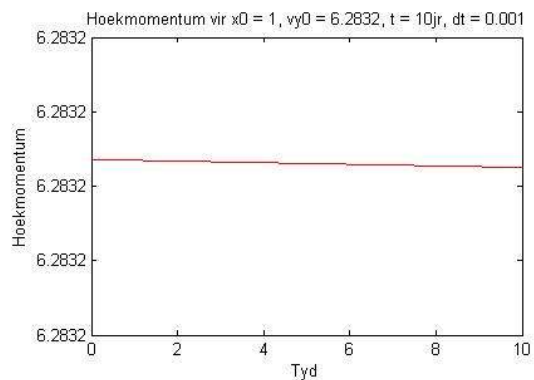
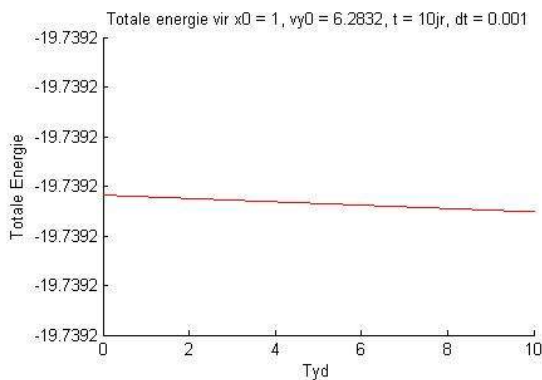
$$x_0 = 1, y_0 = 0, v_{x_0} = 0, v_{y_0} = 2\pi$$

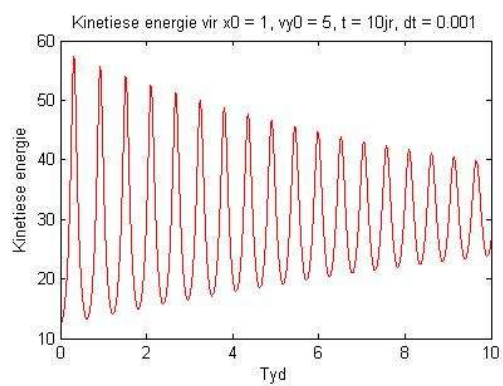
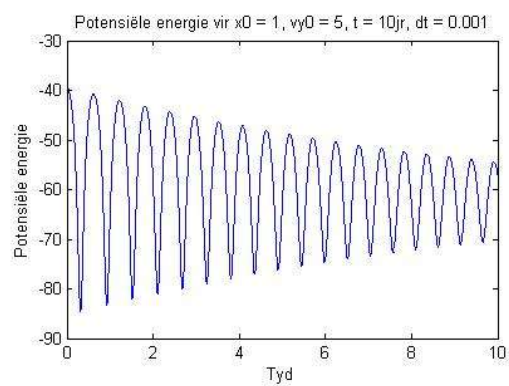
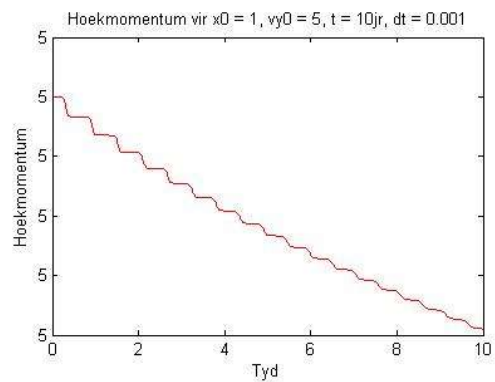
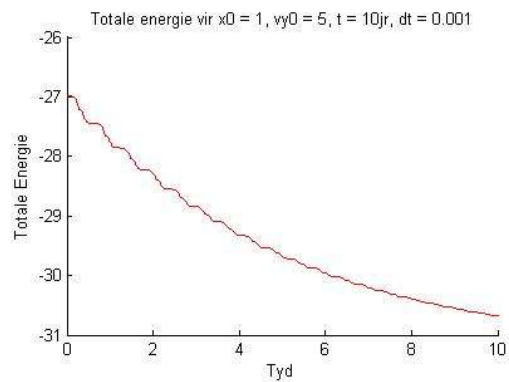
vir 'n tydperk van 10 jaar.

Die kinetiese-, potensiële- en meganiese energie bly almal konstant. Die hoekmomentum

$$L_x = m(x \cdot v_y - y \cdot v_x) \text{ bly ook konstant vir die hele tydperk.}$$

Vir 'n elliptiese baan met $v_{y_0} = 5$ en $x_0 = 1$, ossilleer die potensiële- en kinetiese energie, maar die totale energie bly steeds konstant. Die hoekmomentum bly ook weer konstant.





Taak6:

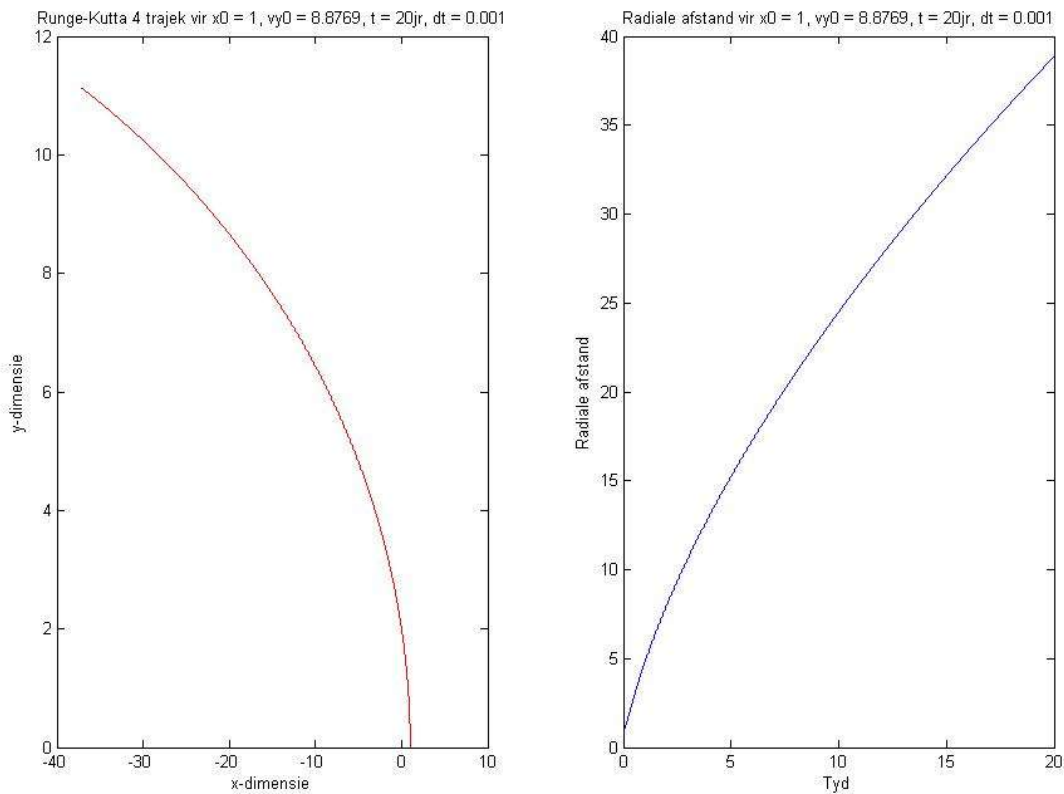
Beskou 'n planeet 1 AE vanaf die son. Gebruik 'n probeer-en-tref-tegniek met wissellende aanvangspoed om die nodige minimum om vanaf die son te kan ontsnap, te bepaal, d.w.s. 'n nie-geslote baan te volg. Vergelyk die simulatie se resultaat met die analities berekende waarde.

Die probleem by hierdie vraag van presies waneer die planeet nie meer in 'n geslote baan om die son beweeg nie, is dat mens nie altyd van 'n grafiek kan aflees of 'n baan geslote is of nie.

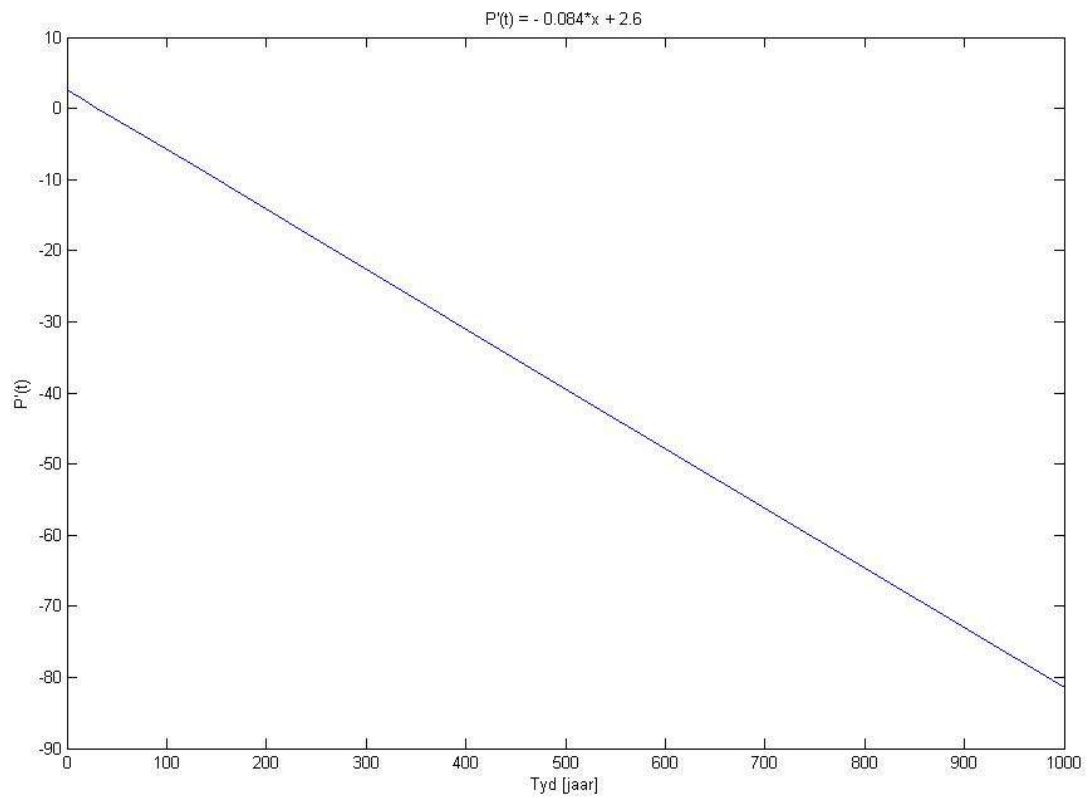
'n Manier om dit te ondersoek is om te kyk wat met die radiale afstand gebeur ver die toekoms in.

Ek het eerstens die radiale afstand teenoor tyd 'geplot'. Met behulp van Matlab se 'Basic fitting' funksie het ek 'n vergelyking vir die grafiek gekry (kwadratiese funksie). Ek het 'n kwadratiese funksie passing gedoen omdat die radiale afstand slegs een draaipunt hoef te hê om 'n omwenteling te doen. Dit was dus nie nodig vir hoër orde polinome nie.

Beskou nou die passingsfunksie $P(t)$ – 'n benadering van die radiale afstand teenoor tyd, en sy afgeleide $P'(t)$. Indien $P'(t) > 0 \forall t > 0$ sal $P'(t)$, en dus die radiale afstand, altyd styg. Die planeet se omwentelings baan is dus nie geslote nie.



Ek het wel die probeer-en-tref tegniek gebruik, maar hoe nader ek aan $v_{y_0} = 2.8284\pi$ gekom het, hoe moeiliker was dit om te besluit of die omwentelingsbaan geslote was al dan nie.



Die passingsfunksie word gegee as $P(t) = -0.042x^2 + 2.6x + 2.7$

$$P'(t) = P'(t) = -0.0840x + 2.6$$

$P'(t)$ is net een keer gelyk aan 0 dus voltooi die planeet sy baan slegs een keer.

Matlab kode

Hierdie is al die hoof programme wat gebruik is in die simulaties. Parameters en funksie roepe moet in calculate en initial verander word om aan te pas by die vraag waarmee gewerk word.

```
function kepler()  
[x,v_x,y,v_y,dt] = initial();  
calculate(x,v_x,y,v_y,dt);
```

```
function [x,v_x,y,v_y,dt] = initial()  
  
rmin = 0.59;  
GM = 4*pi^2;  
P=76;  
a = ((P*sqrt(GM)/(2*pi))^(2/3);  
x=rmin;  
y=0;  
v_x=0;  
v_y=sqrt(GM*(2/rmin - 1/a));  
dt=0.001;  
  
r = 1.4*sqrt(x^2+y^2);  
  
figure('doublebuffer','on');  
subplot(2,2,1);  
axis([-30, 10, -30 30])  
hold all;  
text(0,0,'*');
```

```
function calculate(x0,v_x0,y0,v_y0,dt)  
jr=77;  
t=0:dt:jr;  
[x, y, v_x, v_y, r,KE, PE, E] = RK4(x0, y0, v_x0, v_y0, t, dt);  
  
subplot(2,2,1);  
plot(x,y,'r');  
title(['Runge-Kutta 4 trajek vir',vals(x0, v_y0, jr, dt)]);  
xlabel('x-dimensie');  
ylabel('y-dimensie');  
  
subplot(2,2,2);  
plot(t,E,'r');  
title(['Totale energie vir',vals(x0, v_y0, jr, dt)]);  
xlabel('Tyd');  
ylabel('Totale Energie');  
  
subplot(2,2,3);  
plot(t,x.*v_y - y.*v_x,'r');  
title(['Hoekmomentum vir',vals(x0, v_y0, jr, dt)]);  
xlabel('Tyd');  
ylabel('Hoekmomentum');  
  
subplot(2,2,4);  
plot(t,(x.*v_y - y.*v_x)./2,'r');  
title(['Oppervlakspeed vir',vals(x0, v_y0, jr, dt)]);  
xlabel('Tyd');  
ylabel('Area speed');  
  
figure;  
subplot(2,2,1);
```



```

plot(t,r,'b');
title(['Radiale afstand vir',vals(x0, v_y0, jr, dt)]);
xlabel('Tyd');
ylabel('Radiale afstand');

subplot(2,2,2);
plot(t,PE,'b');
title(['Potensiële energie vir',vals(x0, v_y0, jr, dt)]);
xlabel('Tyd');
ylabel('Potensiële energie');

v = sqrt(v_x.^2 + v_y.^2);
subplot(2,2,3);
plot(t,v,'r');
title(['Baanspoed vir',vals(x0, v_y0, jr, dt)]);
xlabel('Tyd');
ylabel('Baanspoed');

subplot(2,2,4);
plot(t,KE,'r');
title(['Kinetiese energie vir',vals(x0, v_y0, jr, dt)]);
xlabel('Tyd');
ylabel('Kinetiese energie');

```

```

function [KE, PE, E]=energie(v_x,v_y,r);
GM = 4*pi^2;
v = sqrt(v_x.^2+v_y.^2);
KE = 1/2*v.^2;
PE = -GM ./ r;
E = KE + PE;

```

```

function [ret]=vals(x0, v_y0, t, dt)
ret = [' x0 = ', num2str(x0)...
      ', vy0 = ', num2str(v_y0)...
      ', t = ', num2str(t),'jr'...
      ', dt = ', num2str(dt)...
      ];

```

```

function [x,y,v_x,v_y,r, KE, PE, E] = Euler(x0, y0, v_x0, v_y0, t, dt);

x(1)=x0;
y(1)=y0;
v_x(1)=v_x0;
v_y(1)=v_y0;

r(1) = sqrt(x0^2+y0^2);

for i=2:length(t),
    v_x(i)=v_x(i-1) - (4*pi^2 * x(i-1) * dt) / r(i-1)^3;
    v_y(i)=v_y(i-1) - (4*pi^2 * y(i-1) * dt) / r(i-1)^3;

    x(i)=x(i-1) + v_x(i-1)*dt;
    y(i)=y(i-1) + v_y(i-1)*dt;

    r(i) = sqrt(x(i)^2+y(i)^2);
end;

[KE, PE, E]=energie(v_x,v_y,r);

```

```

function [x,y,v_x,v_y,r, KE, PE, E] = Euler_Cromer(x0, y0, v_x0, v_y0, t, dt);

x(1)=x0;
y(1)=y0;
v_x(1)=v_x0;
v_y(1)=v_y0;

r(1) = sqrt(x0^2+y0^2);

for i=2:length(t)
    v_x(i) = v_x(i-1) - (4*pi^2 * x(i-1) * dt) / r(i-1)^3;
    v_y(i) = v_y(i-1) - (4*pi^2 * y(i-1) * dt) / r(i-1)^3;

    x(i)=x(i-1) + v_x(i)*dt;
    y(i)=y(i-1) + v_y(i)*dt;

    r(i) = sqrt(x(i)^2+y(i)^2);

end;
[KE, PE, E]=energie(v_x,v_y,r);

```

```

function [x,y,v_x,v_y,r, KE, PE, E] = RK4(x0, y0, v_x0, v_y0, t, dt);

x(1)=x0;
y(1)=y0;
v_x(1)=v_x0;
v_y(1)=v_y0;

r(1) = sqrt(x0^2+y0^2);

for i=2:length(t)
    faktor(i-1) = - (4*pi^2 * dt) ;

    dx1 = v_x(i-1) * dt;
    dy1 = v_y(i-1) * dt;

    r(i) = sqrt((x(i-1)+0.5*dx1)^2+(y(i-1)+0.5*dy1)^2);

    dvx1 = faktor(i-1)/ r(i-1)^3 * x(i-1);
    dvy1 = faktor(i-1)/ r(i-1)^3 * y(i-1);

    dx2 = (v_x(i-1) + dvx1/2) * dt;
    dy2 = (v_y(i-1) + dvy1/2) * dt;

    r(i) = sqrt((x(i-1)+0.5*dx2)^2+(y(i-1)+0.5*dy2)^2);

    dvx2 = faktor(i-1)/ r(i-1)^3 * (x(i-1) + dx1/2);
    dvy2 = faktor(i-1)/ r(i-1)^3 * (y(i-1) + dy1/2);

    dx3 = (v_x(i-1) + dvx2/2) * dt;
    dy3 = (v_y(i-1) + dvy2/2) * dt;

    r(i) = sqrt((x(i-1)+0.5*dx3)^2+(y(i-1)+0.5*dy3)^2);

    dvx3 = faktor(i-1)/ r(i-1)^3 * (x(i-1) + dx2/2);
    dvy3 = faktor(i-1)/ r(i-1)^3 * (y(i-1) + dy2/2);

    dx4 = (v_x(i-1) + dvx3) * dt;
    dy4 = (v_y(i-1) + dvy3) * dt;

    r(i) = sqrt((x(i-1)+0.5*dx4)^2+(y(i-1)+0.5*dy4)^2);

    dvx4 = faktor(i-1)/ r(i-1)^3 * (x(i-1) + dx3);
    dvy4 = faktor(i-1)/ r(i-1)^3 * (y(i-1) + dy3);

    x(i) = x(i-1) + dx1/6 + dx2/3 + dx3/3 + dx4/6;

```

```
y(i) = y(i-1) + dy1/6 + dy2/3 + dy3/3 + dy4/6;  
  
v_x(i) = v_x(i-1) + dvx1/6 + dvx2/3 + dvx3/3 + dvx4/6;  
v_y(i) = v_y(i-1) + dvy1/6 + dvy2/3 + dvy3/3 + dvy4/6;  
  
r(i) = sqrt(x(i)^2+y(i)^2);  
end;  
[KE, PE, E]=energie(v_x,v_y,r);
```