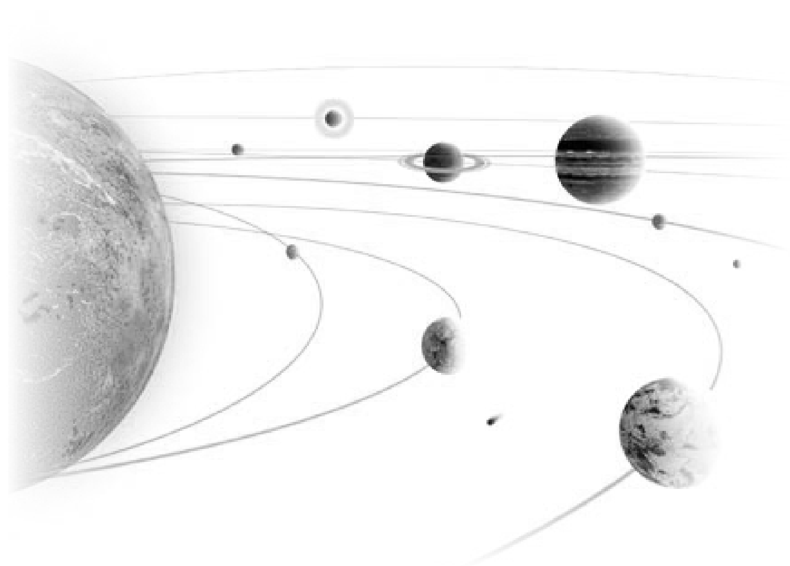
Prosjektoppgave i FYS-MEK 1110 våren 2006

Utforsking av ”tre-legeme-problemet”. Verson 0605141130



André Christoffer Andersen
Gruppe 23

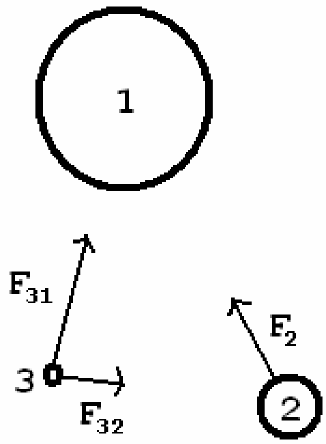
I dette prosjektet ser vi på banen til himmellegemer i solsystemet. I utganskpunktet er det fullt mulig å analytisk bestemme mye av fenomene vi ser på himmelen, men så snart vi skal ta til etterettning flere enn bare to legemer blir beregningene fort vannskelig. I dette prosjektet skal vi derfor bruke numeriske kalkulasjoner ved hjelp av programeringsspåket i MATLAB. Vi må ta utgangspunkt i noen formler før vi kan sett i gang med programering. Fra newtons gravitasjonslov og andre lov har vi at:

$$F = -\frac{GMm}{r^2}\vec{u}_r$$

$$\vec{F} = \vec{m}a$$

$$\vec{a} = \vec{F}/m = \frac{\left(-\frac{GMm}{r^2}\vec{u}_r\right)}{m} = -\frac{GM}{r^2}\vec{u}_r = -\frac{GM}{r^3}\vec{r}$$

Over finner du kraften F mellom himmellegmene, massen m av det aktuelle himmellegmet, gravitasjonskonstanten G, massen M av motstående himmellegme, avstanden r mellom dem og vektoren $\vec{r}$ mellom dem. Dette betyr i praksis at det aktuelle himmellegemets masse ikke har påvirkning på sin egen bane. Nå som vi har et utgangspunkt å arbeide fra velger jeg å bruke Euler-Cromes metode for sy sammen akselerasjon, hastighet og posisjon. Selv om den har klare svakheter viser det seg at den ofte gir overaskende bra modeller. For å ikke unødvendig komplisere koden har jeg ignorert kreftene fra mindre legemer på klart større. Legeme 1 (Solen) ignorerer kreftene fra legeme 2 og 3. Legeme 2 (Jupiter) ignorerer kraften fra legeme 3, men blir påvirket av legeme 1. Legeme 3 (astroiden) påvirker verken legemene 1 eller 2, men blir dermot påvirker av dem begge. Denne forenklingen unnskyldes med at vi i utgangspunktet vil ignorere Jupiters påvirkning på Solen og her er størrelsesforholdet bare rundt 1000, mens forholdet mellom jupiter og en vanlig asteroide kan være i hundremilloner størrelsen. Svært neglesjerbart altså. For å finne ut massen mk til en komet bruker jeg volumet V for en kule multiplisert med massetetheten p av mars:

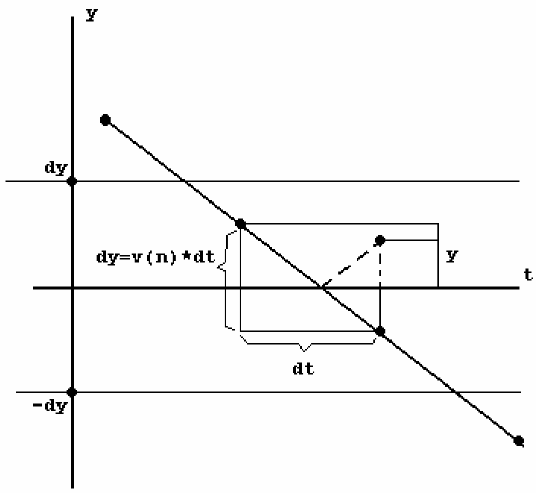


Figur 1 – Krefter

$$\frac{m_s}{m_j} \approx 1050$$

$$\frac{m_j}{m_k} = \frac{m_j}{V\rho} = \frac{m_j}{\frac{4\pi r^3}{3}\rho} \approx 195 \cdot 10^6$$

Får å studere systemet nærmere bør vi finne ut av periodetiden. Dette kan vi gjøre med å legge til en if-test som finner ut av rettning og tidspunkt for hver pasering av xz-planet, men denne if-testen har en viss unøyaktighet som vi må bli nødt til å kvitte oss med gjennom interpolasjon. If-testen ser om absoluttverdien av dy er større enn posisjonen y. Er dette sant kan det bare bety at det er like før legemet passerer eller at legemet har akkurat pasert xz-planet. Hvilken finner vi ut av ved å be om hastigheten i y-rettningen er positiv eller negativ samtidig som vi ber om y-posisjonen er positiv eller negativ. Er begge positiv eller negativ så vet vi at itereringen har kommet til første steget etter å pasere xz-planet. Nå som vi vet hva n og n+1 er så kan vi bruke bruke interpolasjon ved hjelp av:



Figur 2 – Interpolasjon

$$t_{kryssing} = t_n - \frac{y_n}{y_{n+1} - y_n} \cdot \Delta t$$

Grunnen til en så komplisert metode for å finne kryssningen er at det nå er likegyldig hvilken rettning himmellegemet kommer fra. Atlså hvis vi kommer over et legeme som går i motsatt rettning av det normale så vil vi enda kunne måle periodetiden sammtidig som vi kan måle pasering av xz-planet når den både går inn i og ut av positiv y. Kryssningstidspunktet blir lagret og kan ved slutten av programmet skrives ut og behandles.

Om ikke annet er sakt så er alle posisjonsgrafer y-posisjon vs x-posisjon og i meter.

MATLAB er spesielt egnet for vektorer, arrayer og matriser, derfor vil datastrukturen kunne settes opp som i tabell 1. Vi kan også gjøre vanlige matematiske opperasjoner direkte på vektorene. Vi trenger altså ikke å "manuelt" iterer gjennom alle verdiene i vektorene og arrayene for å endre på dem, noe som gjør arbeidet svært lettere og mer oversiktlig.

Tabell 1: Banedata

tid	x	y	z
t1	x1	y1	z1
t2	x2	y2	z2
t3	x3	y3	z3
t4	x4	y4	z4
....	...	...	...
tn	xn	yn	zn

MATLAB:

```
%--------------------------------------------------------------------------------
%Tar inn en brøk som parameter. Den skal brukes til å skalere periodetiden
%til et av himmellegemene i forhold til det andre.
function grv = grv(Tbrok);

%Gravitasjonskonstanten for det metriske systemet
G = 6.67259E-11;

%Massene for legemene
m1 = 1.99E30;
m2 = 1898.6E24;
m3 = 1E17;

%Sekunder i en dag
s_dag = 86400;

%Dager i året
d_aar = 365.25636;

%Resolusjon
delta_t = s_dag;

%Tiden vi skal følge bevegelsene
max_t = s_dag * d_aar * 150 * 11;

%Antall steg vi skal ta i forhold til resolusjon og maksimal tid
steg = round(max_t/delta_t);

%Array for posisjonsvektorene til legeme 2 og 3
R2 = zeros(3, steg);
R3 = zeros(3, steg);

%Array for tiden til alle legemene, perioden for legeme 3 og avstanden mellom legeme 3 og 2
t = zeros(1,steg);
T = zeros(1,steg);
R32abs = zeros(1,steg);

%Startposisjon til legeme 2
r2 = [  -5.758295190842140E+11, -5.704511997951533E+11, 1.525358540575477E+10];
%Starthastighet til legeme 2
v2 = [   9.038714838773942E+03, -8.674265884570330E+03, -1.662886596994482E+02];

%Startposisjon til legeme 3
r3 = [3.557989834508110E+11, 3.557989834508110E+11, -2.063566923033582E+11];
%Starthastighet til legeme 3
v3 = [  1.083148303916873E+04, 5.276386924919434E+03, 3.576615538180552E+03];

%Hvis det blir oppgitt en brøk så overskriver vi startverdiene for legeme 3
if Tbrok > 0
    r3 = (r2.*1.00) .* Tbrok^(2/3);
    v3 = v2 .* Tbrok^(-1/3);
end

%Initaliserer to variabeler for å lagre passeringene av xz-planet slik at
%vi kan beregne periodetidene.
t_inn=0;
t_ut=0;

for i = 1:steg
%Tar inn en brøk som parameter. Den skal brukes til å skalere periodetiden
%til et av himmellegemene i forhold til det andre.
function grv = grv(Tbrok);

%Gravitasjonskonstanten for det metriske systemet
G = 6.67259E-11;

%Massene for legemene
m1 = 1.99E30;
m2 = 1898.6E24;
m3 = 1E17;

%Sekunder i en dag
s_dag = 86400;

%Dager i året
```

```
d_aar = 365.25636;

%Resolusjon
delta_t = s_dag;

%Tiden vi skal følge bevegelsene
max_t = s_dag * d_aar *20* 11.86;

%Antall steg vi skal ta i forhold til resolusjon og maksimal tid
steg = round(max_t/delta_t);

%Array for posisjonsvektorene til legeme 2 og 3
R2 = zeros(3, steg);
R3 = zeros(3, steg);

%Array for tiden til alle legemene, perioden for legeme 3 og avstanden mellom legeme 3 og 2
t = zeros(1,steg);
T = zeros(1,steg);
R32abs = zeros(1,steg);

%Startposisjon til legeme 2 (Jupiter)
r2 = [  -5.758295190842140E+11, -5.704511997951533E+11, 1.525358540575477E+10];
%Starthastighet til legeme 2
v2 = [   9.038714838773942E+03, -8.674265884570330E+03, -1.662886596994482E+02];

%Startposisjon til legeme 3 (Achilles)
r3 = [2.846769934538636E+11 -8.318966217997770E+11 -7.444700009655958E+10];
%Starthastighet til legeme 3
v3 = [1.047688935274932E+04  4.100248482776472E+03  1.853655907948878E+03];

%Hvis det blir oppgitt en brøk så overskriver vi startverdiene for legeme 3
if Tbrok > 0
    r3 = (r2.*1.00) .* Tbrok^(2/3);
    v3 = v2 .* Tbrok^(-1/3);
end

%Initaliserer to variabeler for å lagre passeringene av xz-planet slik at
%vi kan beregne periodetidene.
t_inn=0;
t_ut=0;

for i = 1:steg
    %Oppdaterer tiden
    t(i) = (i-1)*delta_t;

    %Lagrer andre legemes posisjonen
    R2(1,i) = r2(1); R2(2,i) = r2(2); R2(3,i) = r2(3);

    %Lagrer tredje legemes posisjonen
    R3(1,i) = r3(1); R3(2,i) = r3(2);  R3(3,i) = r3(3);

    %Avstanden mellom legeme 2 og 1
    r2abs = sqrt( r2(1)*r2(1) + r2(2)*r2(2) + r2(3)*r2(3) );

    %Avstanden mellom legeme 3 og 1
    r3abs = sqrt( r3(1)*r3(1) + r3(2)*r3(2) + r3(3)*r3(3) );

    %Vektoren mellom legeme 3 og 2
    r32 = r2-r3;

    %Avstanden mellom legeme 3 og 2
    r32abs = sqrt( r32(1).*r32(1) + r32(2).*r32(2) + r32(3).*r32(3) );
    R32abs(i) = r32abs;

    %Faraten til hastighetsvektoren til legeme 3
    v3abs = sqrt(v3(1).*v3(1) + v3(2).*v3(2) + v3(3)*v3(3));

    %Momentan perioden for legeme 3
    T(i) = (2*pi*r3abs)/v3abs;

    %Akselerasjonen til legeme 2 fra legeme 1
    a2 = -( G*m1 .* r2 ) / (r2abs^3);

    %Akselerasjonen til legeme 3 fra legeme 1
    a3 = -( G*m1 .* r3 ) / (r3abs^3);

    %Akselerasjonen til legeme 3 fra legeme 2
```

```
    a32 = -( G*m2 .* r32 ) / (r32abs^3);

    %Hastigheten til legeme 2
    v2 = v2 + a2 .* delta_t;

    %Hastigheten til legeme 3
    v3 = v3 + (a3 + a32) .* delta_t;

    %Posisjon til legeme 2
    r2 = r2 + v2 * delta_t;

    %Posisjon til legeme 2
    r3 = r3 + v3 * delta_t;

    %Er absoluttverdien av avstanden til xz-planet mindre enn sist delta r
    %så skal vi kan vi starte å finne krysningspunktet
    if abs( r3(2) ) < abs( v3(2) * delta_t)
        %Er y komponentene til farten og posisjonen positiv har vi akkurat
        %passert xz-planet i possitiv rettning.
        if v3(2) > 0 && r3(2) > 0
            %Utvider en array for lagring av tidspunktet
            t_inn = [t_inn; t(i) - ( R3(2,i) * delta_t ) / ( r3(2) - R3(2,i))];
        end

        %Er y komponentene til farten og posisjonen negativ har vi akkurat
        %passert xz-planet i negatic rettning.
        if v2(3) < 0 && r3(2) < 0
            %Utvider en array for lagring av tidspunktet. Her represnterer
            %R3(2,i) = y(1), r3(2) = y(n+1) og t(i) = t(n)...
            t_ut = [ t_ut; t(i) - ( R3(2,i) * delta_t ) / ( r3(2) - R3(2,i) ) ];
        end
    end


end
    %Fjerner første null i arrayene. Kan evt. skrive dem ut også

    t_inn=t_inn(2:length(t_inn));
    t_ut=t_ut(2:length(t_ut));

%Plotter tid vs periode og kraftforhold
figure(1)
F32=4E31./(R32abs.*R32abs);
plot( t(:),F32(:),'-r', 'LineWidth', 2  );
hold on;
plot( t(:), T(:) );
hold off;
grid on;

%Plotter x-posisjon vs y-posisjon
figure(2)
plot( R3(1, :), R3(2, :));
hold on
plot( R2(1, :), R2(2, :), '-r', 'LineWidth', 3);
plot(0,0, '.', 'MarkerSize', 30);
hold off
axis('equal');

%figure(4)
%plot3( R3(1, 1: steg), R3(2, 1: steg), R3(3, 1: steg) );
%hold on
%plot3( R2(1, 1: steg), R2(2, 1: steg), R2(3, 1: steg), '-r', 'LineWidth', 2 );
%plot3(0,0,0, '.', 'MarkerSize', 30);
%hold off
%axis('equal');
%grid on;
%grid on;
%------------------------------------------------------------------------------------
```

La oss først studere hvordan forholdet mellom Jupiter og Solen utarter seg. Over ca 50 julianske år passerer Jupiter xz-planet, inn i positiv y, ved følgende tidspunkter:

Tabell 2: xz-passeringer

t1	1.3914e+008
t2	5.1354e+008
t3	8.8793e+008
t4	1.2623e+009

Gjennomsnittet av periodene som ble kalkulert gis ved:

$$T_j = \frac{(t_4 - t_3) + (t_3 - t_2) + (t_2 - t_1)}{3} = \frac{t_4 - t_1}{3}$$
$$= 3.7438 \cdot 10^8 s \cdot \left(\frac{1\text{dag}}{86400\text{s}}\right)\left(\frac{1\text{år}}{365.25\text{dag}}\right) = 11.86\text{år}$$

Det viser seg altså at Jupiter bruker 11.86 ganger så lang tid på å bane seg vei rundt solen enn hva jorden gjør. Dette kan umulig være bra for avlinga :p. Vi vet fra tidligere at massen til himmelegemet vi har fokus på ikke skal invirke på dets egen bane, dette bekrefter også programmet – altså ingen endringer i periodetid eller plot. Det skal sies at om vi hadde laget en fullstendig modell så ville Jupiter dra på Solen og således påvirke periodetiden til seg selv, men siden vi ignorer dette så er det ingen forskjell som sagt. Derimot vil endring i hastighet bety svært mye for banen som blir laget. Halv hastighet gir en periodetid på:

$$T_j = \frac{t_9 - t_1}{8} = \frac{1.4455 \cdot 10^9 \text{ - } 8.2256 \cdot 10^7}{8}$$
$$= 1.7040 \cdot 10^8 s \cdot \left(\frac{1\text{dag}}{86400\text{s}}\right)\left(\frac{1\text{år}}{365.25\text{dag}}\right) = 5.40\text{år}$$

Ved bruk av dobbel hastighet løsriver Jupiter seg fra solens gravitasjonsfelt og driver ut mot universet. Dette bekreftes analytisk ved:

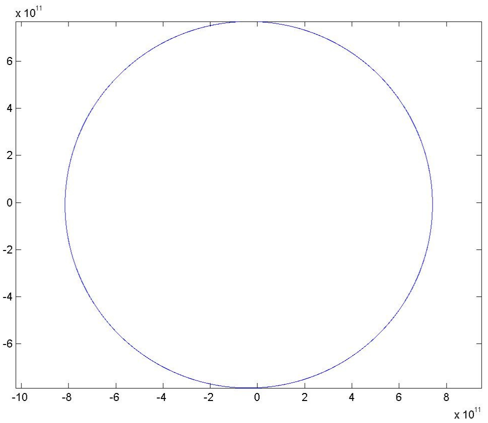
$$v_e = \sqrt{\frac{2GM}{R}} = \left(\sqrt{\frac{2 \cdot (6.7 \cdot 10^{-11}) \cdot (2.0 \cdot 10^{30})}{7.8 \cdot 10^{11}}}\right) m/s =$$
$$v_j = 2\sqrt{(x)^2 + (y)^2 + (z)^2} = 2\left(\sqrt{(9.0 \cdot 10^3)^2 + (-8.8 \cdot 10^3)^2 + (-1.7 \cdot 10^2)^2}\right) m/s$$
$$v_e = 1.9 \cdot 10^5 m/s < 2.5 \cdot 10^5 m/s = v_j$$

Her har vi x, y og z komponentene til jupiters initielle hastighet vj, gravitasjonskonstanten G, Solas masse M, og avstanden R til sola og løsrivingshastighetn ve. Bare ca tall er brukt for å få et raskt overslag. Altså det er klart at

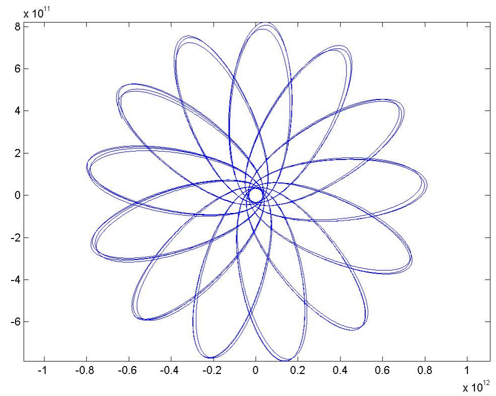
Jupiter med så stor fart vil rive seg løs fra solens gravitasjonsfelt.

La oss prøve å manipulere litt på de fysiske premissene. Hva om vi sier at kraften ikke er proposjonal med r^2, men snarere $r^{2.1}$ eller kanskje $r^{1.9}$. Et vanlig plott av Jupiters bane er helt sirkulært slik som i figur 2, der r^2 brukes. Ser vi på $r^{1.9}$ får vi plottet i figur 4. Det er klart at ved lavere eksponent vil kreftene øke og dermed tvinge Jupiter nærmere ved selv større avstander. Ved $r^{2.1}$ divergerer og forlater Jupiter solsystemet. Det er fasinerende å se hvor lite som skal til for at naturen ikke er i balanse. Man blir satt i et philosofisk lys når man tenker over hvor matematisk perfekt verden oppfører seg. Hvorfor alt systemet?

Figur 3 – Jupiter bane



Figur 4 – tøyse fys: $r^{1.9}$



En slik sirkelformet bane må være bunnet i mattematisk symetri. På tide å skjære oss i gjennom og finne hva som ligger bak og utlede et forhold som gjør at vi kan lage sirkulære baner på komando.

Vi har kraften F, massen M til det påvirkende himmellegemet, avstanden r mellom objektet og subjektet og tyngdeakselerasjonen a. Bruker vi da Newtons gravitasjonslov og andre lov som vi utledet i begynnelsen kan vi ved bruk av sentripental akselerasjonen med periodetiden T finne Keplers 3. lov. Dette er ikke et forsøk på noe bevis, da det krever en del mer, men for sirkulære tilfeller av planetbevegelser har vi at:

$$a = \frac{GM}{r^2} = \frac{4\pi^2 r}{T^2}$$

$$T^2 = \frac{4\pi^2}{GM} r^3 \left(\Rightarrow T = \sqrt{\frac{4\pi^2}{GM}} r^{3/2} = k r^{3/2} \text{ Keplers 3. lov} \right)$$

Ved bruk av Keplers 3. lov kan man finne forholdet mellom en ønsket ny bane og Jupiters kjente bane. Da faller mange av variablene bort:

$$\frac{T_x^2}{T_j^2} = \frac{\left(\frac{4\pi^2}{GM}\right) r_x^3}{\left(\frac{4\pi^2}{GM}\right) r_j^3}$$

$$r_x = \left(\frac{T_x}{T_j}\right)^{2/3} r_j$$

Denne formelen kan vi bruke for å transformere Jupiters baneradius, hastighet og periodetid, men med samme form. Ønsker vi å se på i forhold til hastighet kan vi bare bruke omkretsen O og hastigheten v.

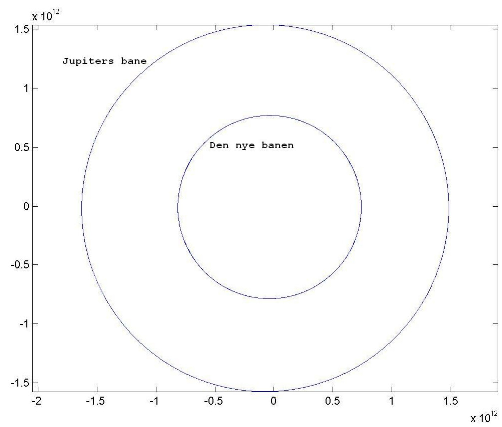
$$T = \frac{O}{v} = \frac{2\pi r}{v}$$

$$r_x = \left(\frac{T_x}{T_j}\right)^{2/3} r_j = \left(\frac{\frac{2\pi r_x}{v_x}}{\frac{2\pi r_j}{v_j}}\right)^{2/3} r_j = \left(\frac{r_x v_j}{r_j v_x}\right)^{2/3} r_j$$

$$\left(v_x^{\,2/3}\right)^{3/2} = \left(\frac{r_j}{r_x}\left(\frac{r_x}{r_j}\right)^{2/3} v_j^{\,2/3}\right)^{3/2}$$

$$v_x = \left(\frac{|r_j|}{|r_x|}\right)^{1/2} v_j$$

Figur 5 - Skalering av bane



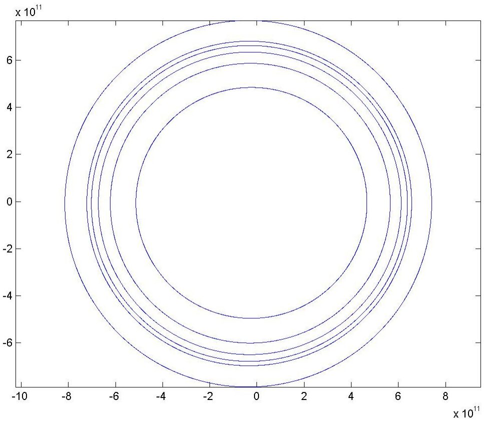
Disse formlene vil bli yppelig for å testes i programet. Prøver nye initalverdiene for et jupiter, men med halve avstandene fra sola og samme form:

$$r_x = 2r_j$$

$$v_x = \left(\frac{|r_j|}{|r_x|}\right)^{1/2} v_j = \left(\frac{|2r_j|}{|r_j|}\right)^{1/2} v_j$$

$$v_x = \sqrt{2} v_j$$

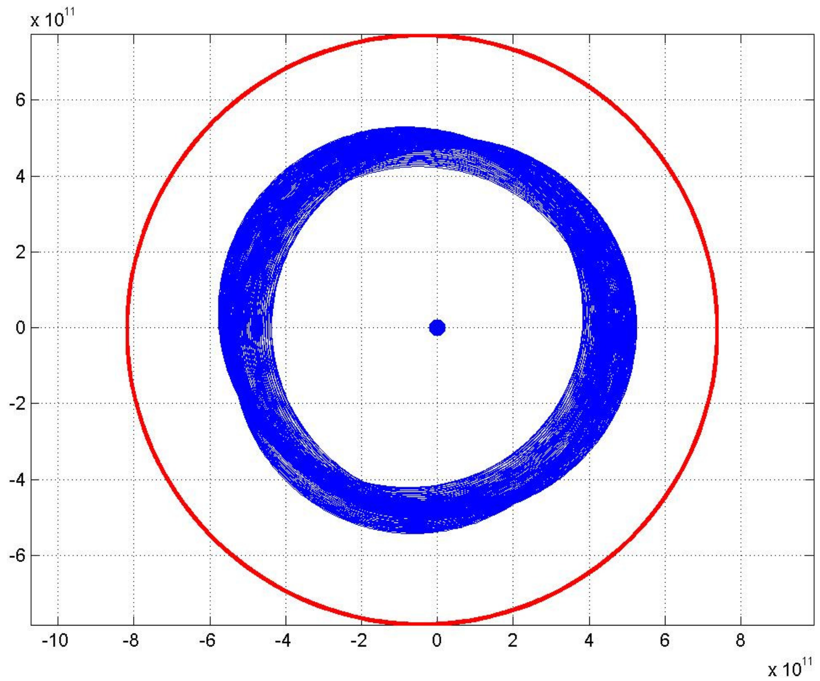
Ved plotting virker det som om at likningen er riktig. Denne formelen kan vi implimentere i koden fullstendig. På den måten kan vi se litt nærmere på fenomenet "gravitasjonell resonans", der vi ved forskjellige brøker av jupiters omløpstid finner at himmellegemer har vanseligheter med å holde seg i sirkulær bane. Disse områdene kalles "Kirswood-gapene". Fra innerst til ytterst bane i Figur 5 har vi brøkdelene av Jupiters perioder gitt med: 1/2, 2/3, 3/4, 4/5 og 5/6 (samt Jupiter selv)



Figur 6 – Brøk skalering

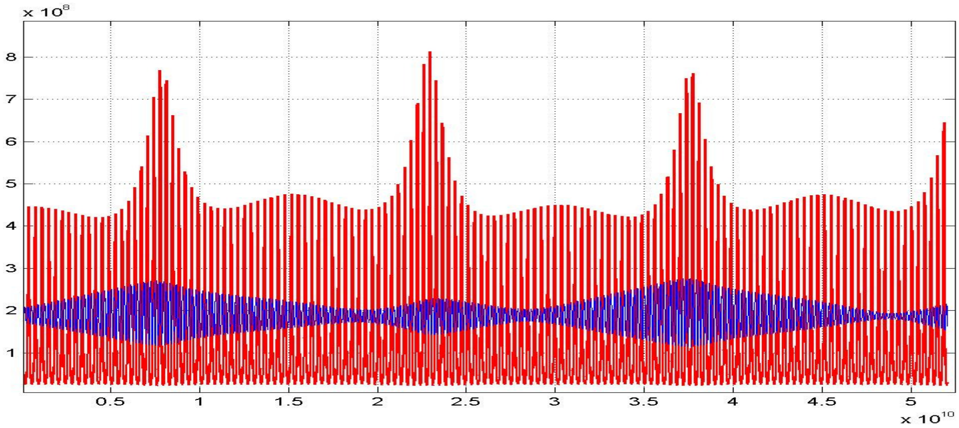
Til nå har vi bare sett på to himmellegemer om gangen. Hvordan vil systemet reagere om vi plasser en ny aktør inn i bildet. Det er derfor viktig å sette inn riktig initialbetingelser for rikig størrelsesorden. Alle variablene i programmet med lavest indeks er størst og hvis det er to så er det det første indekstall i forhold til neste indekstall. For eksempel er v2 hastigheten til legeme nummer to mens r32 er posisjonsvektoren mellom legeme 3 og 2 (asteroide og jupiter). Setter så inn en astroide på 1.0e17 kg (har egentlig ingen ting å si) med halve Jupiters periodetid.

Figur 6 – Resonerende omløpsbane



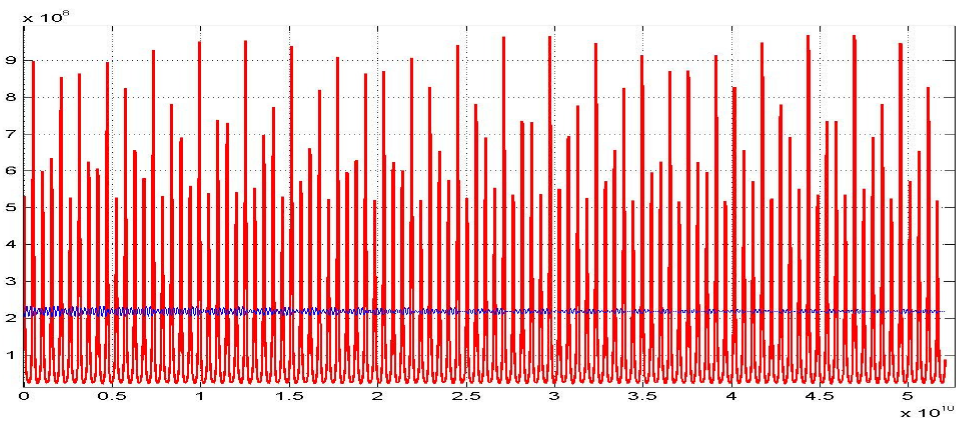
Den røde sirklen viser Jupiters bane, den blå er astroiden og punktet er sola. Vi ser klart at astroiden går i en uperturbert bane og virker som om den ligger i et spesielt område mellom Sola og Jupiter.

Figur 7 – 1/2 Tj blå: periode vs tid
rød: kraftforhold

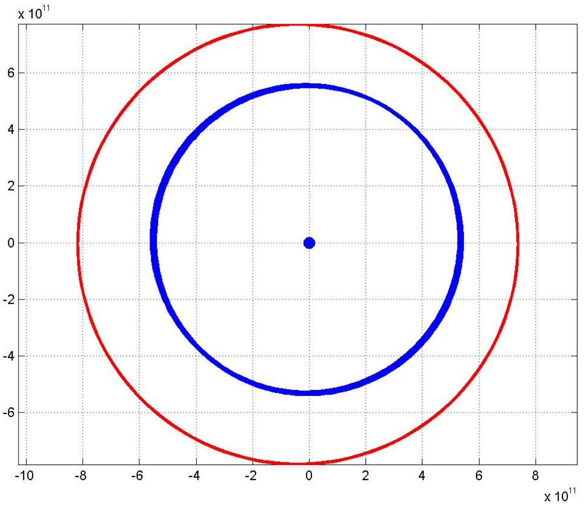


Den spesiell tilstanden vi ser ved brøken 1/2 viser at det er en slags harmoni – et samarbeid mellom kraftkildene som gjør en symetrisk påvirkning. I figur 7 er det blå feltet selveste perioden mens, det røde representerer kraftforholdet som kommer fra jupiter. Altså ser vi at perioden blir påvirket i harmoniske svigninger alt etter avstanden til jupiter. Siden kraften er k/r^2 der k er alle konstantene i ei "suppe" så får vi at jo høyere (den røde) grafen er jo mer påvirkes periodetiden (blå). Denne harmoniske virkningen ser ut til å fortsette videre. Som vi vet når det kommer til resonerende systemer så vil de virke stabiliserende eller tvert om. Selv en så liten endring på 5% av radiusen vil bety at resoneringen at denne spesielle stillingen går over til en spesielt stabil curve.

Figur 8 – 5% endring i r blå: periode vs tid
rød: kraftforhold



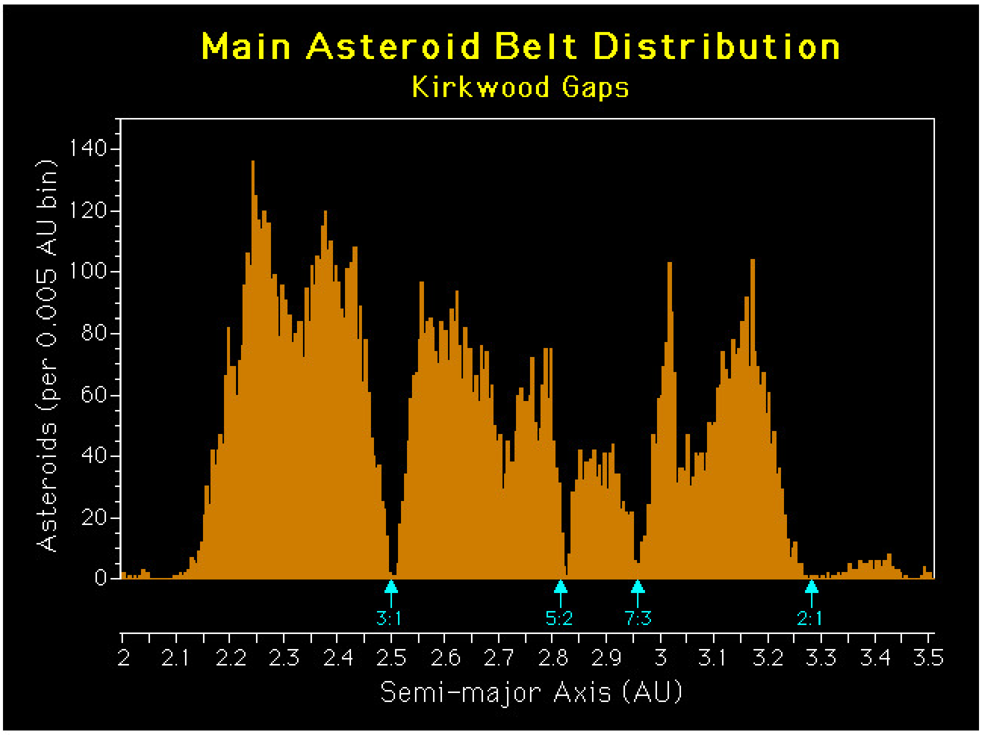
Figur 9 - 5% endring av r



I figur 8 ser vi at kreftene fra Jupiter nå er spredd likt. En jevn fordeling som her betyr som periodeplottet tilsier en stabil og sirkulær bane. Det er tydeligvis svært lite som skal til for at et legeme bli skjøvet ut ved dette grenspunktet. I figur 9 ser vi klart hvor "fin" bane som blir dannet med en så slik distrubusjon av krefter over tid. Finnes det noen fenomener viser disse forholden i vårt virkelige solsysteme?

Vi har asteriodebeltene som starter ved 2.1 AU fra solen til ca 3.3 AU. Hvor ligger så astroiden for 1/2 av jupiters omløpstid? Nettopp ved 3.3 AU. Ser vi på fordelingen av asteroidebeltet i figur 10 ser vi at dette punktet faktisk er et grensområde for asteroider.

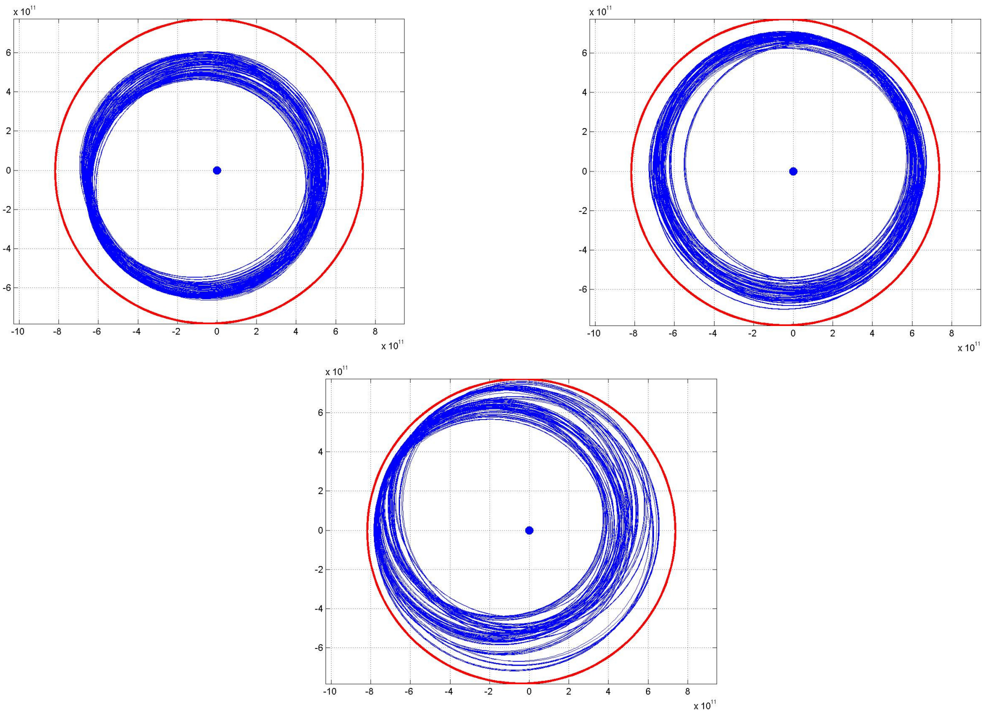
Figur 10



This graph was created in January 1997 using all asteroids with "well-determined" orbits.
Hentet fra http://ssd.jpl.nasa.gov/?histo_a_ast 12. mai 2006

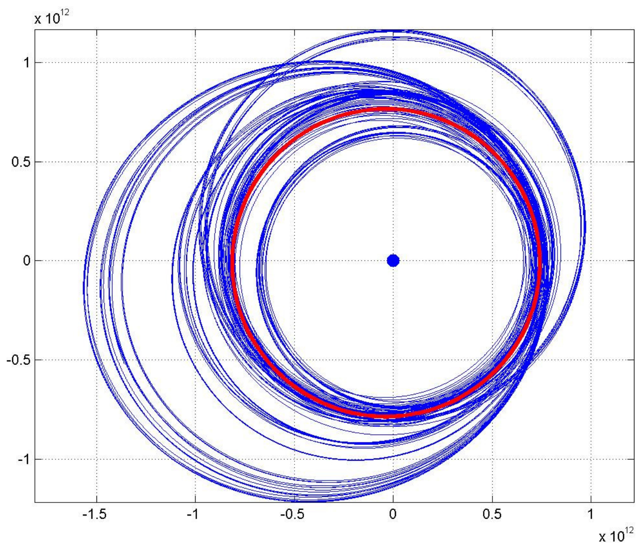
Kirkwood gapene det er her snakk om kommer av spesielle tilfeller av heltalls brøker av perioden til Jupiter. Utviklingen vi ser ved figur 8, 11 og 12 er at banen blir stadig mer og mer ustabil. Noe som understøttes av virkelige observasjoner. Etter 3.3 AU og utover er det få asteorider.

Figur 11 - astroide med T 2/3, 3/4 og 4/5



I figur 11 ser vi tre plott med henholdsvis periodetider og skalering av baner som tilsvarer brøkene 2/3, 3/4 og 4/5. Legg merke til den stadig mer ustabile tilstanden etter som vi nærmer oss samme omløpstid og radius som Jupiter. Dette er selvsagt naturlig da jo nærmere astoriden kommer jupiter jo mer blir den påvirket.

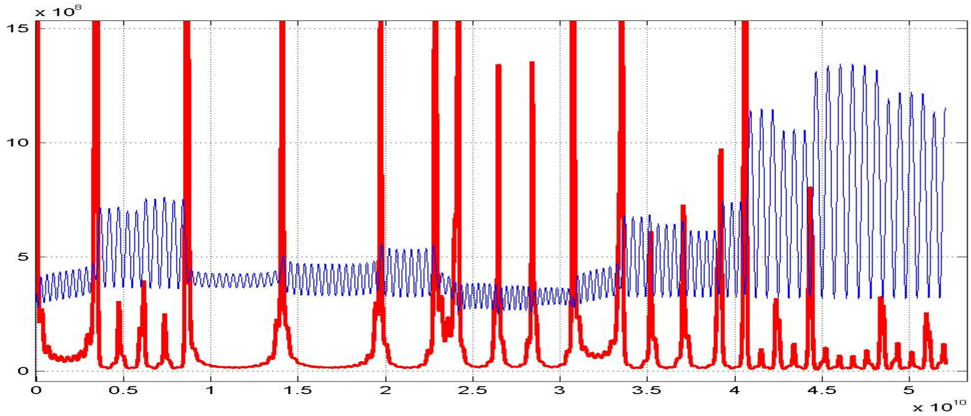
Figur 11 - 5/6 T



Ved 5/6 av Jupiters periodetid kan vi klart se tilfeller der asteoriden kommer i nærkontakt med Jupiter. Det skjer fenomnele endringer i banen over forholdsvis kort tid.

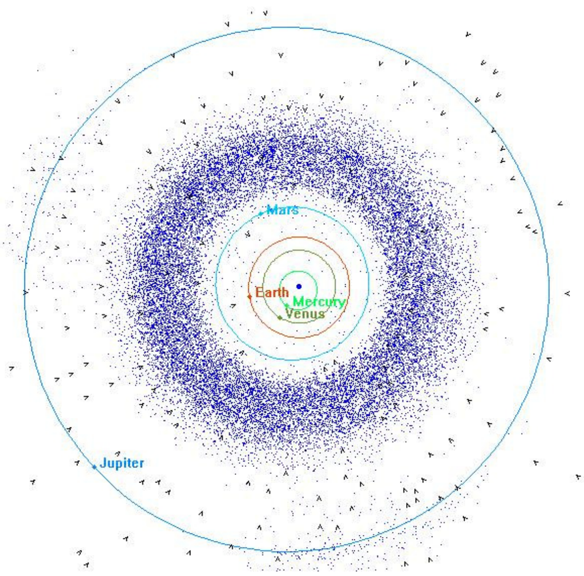
Figur 12 – 5/6 T

blå: periode vs tid
rød: kraftforhold



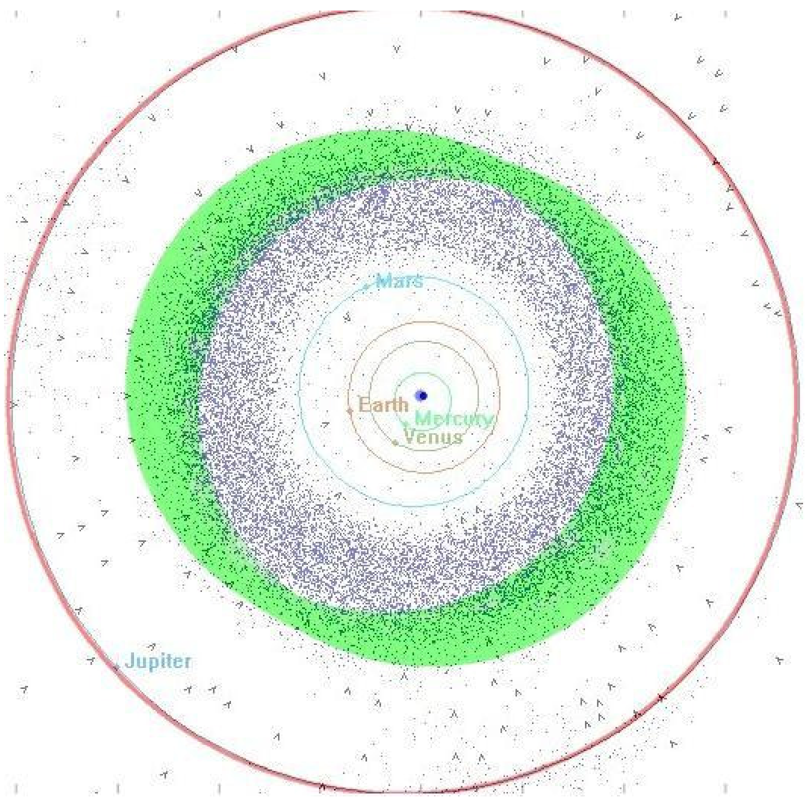
Det som karakteriserer bevegelsen er at astroiden og Jupiter stort sett ikke er nær hverandre, men først når de kommer nær så er det snakk om svært nært og astroiden reagerer enormt. Den blir på en måte slynget omkring. På Figur 12 ser det ut som forskjellige nivåer med baner. For hver gang vi har et klimaks av krefter så skjer store endringer i perioden. Alle store endringer i perioden er avskilt med et kraftklimaks. Disse platåene som dannes av periodetiden ser vi også igjen i figur 11. Hver tilnærmet sirkel er et intervall der perioden svinger konstant. Og jo smalere periodesvingien er jo mer sirkulært går banen. For eksempel kan det tenke seg at den store sirkelen vi kan kjenne igjenn nede tilvenstre i figur 11 er den samme som siste platået i figur 12. La oss se tilbake på distribusjonen til asteoridene.

Figur 13 – Asteoride distribusjon



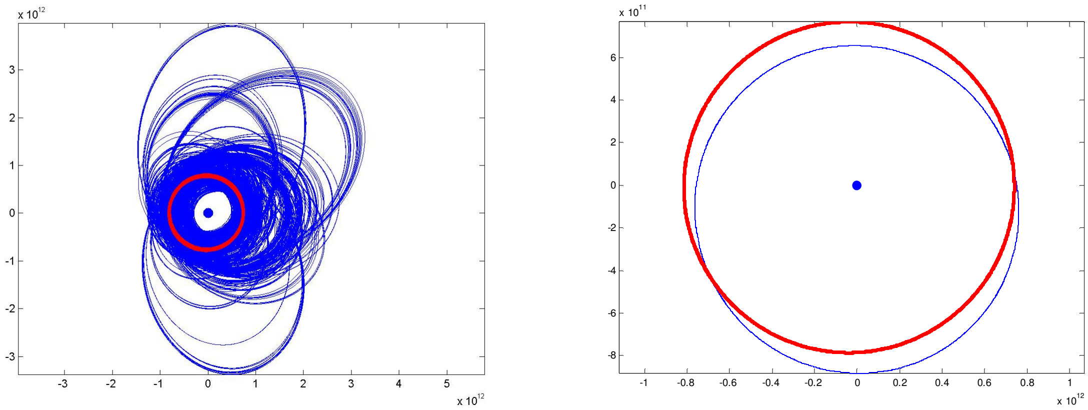
Altså hvorfor finnes dette beltet av asteorider som kommer rett etter mars? Som vi nevnt er de et produkt av forholdet mellom Jupiter og Solen. Det spesielle karekteristikkene til dette belte beskytter og verner de indre planetene mot himmellegemer som vil smaddre dem. Dette har vært esensielt for at liv på jorda kunne utvikle seg. Jupiter stabiliserer solsystemet på en slik måte at den fanger opp legemer som ville ellers hatt svært kaotiske baner, med muligheten for langt mer nedslag. Utviklingen av liv ville aldri kommet til det nivået vi har i dag hvis Jorda har fått nedlag oftere enn nå. Faktisk er vi allerde på overtid før neste kataklysmiske nedlaget.

Figur 14 – Grøn: Bane til 1/2 T
Blå: Asteoridefordeling
Rød: Jupiterbanen



I figur 14 ser vi hvordan vår asteoride med halve perioden til Jupiter legger seg i forhold til de andre kjente asteroider i solsystemet. Den ligger som vi vet fra figur 10 helt ved ytterkanten av asteoridebeltet. Alle asteorider med avstander fra Solen større enn beltet gir ustabile baner, noe som vi gjenkjenner fra tidligere kalkulasjoner og plottinger.

Figur 15 Blå: Achilles, Rød: (egentlig) jupiterbane



Til slutt kan vi se på forskjellen mellom banen til Asteroiden Achilles. Til venstre i figur 15 er Jupiter tatt med i beregningene til høyre ser vi hvordan banen er om jupiter ikke var til stede. Enorm forskjell, altså.