Vrij Technisch Instituut
Papebrugstraat 8A
8820 Torhout

## Geïntegreerde Proef

# Baanbeschrijving

**Uitvoering**
Frank Vanwynsberghe
Duyck Leander
Stefaan Desender

**Klas** 614
**Richting** Industriële Wetenschappen
**Titularis** Karin Degryse
**Schooljaar** 1999-2000

VTI-Torhout
Papebrugstraat 8A
8820 Torhout

## Geïntegreerde proef

# Baanbeschrijving

**Eindwerk**
Frank Vanwynsberge
Leander Duyck
Stefaan Desender
**Klas** 614
**Richting** Industriële Wetenschappen
**Titularis** Karin Degryse
**Schooljaar** 1999-2000

## *Voorwoord*

*Deze geïntegreerde werd door ons geschreven in het kader van het behalen van ons diploma van het technisch secundair onderwijs.*

*In eerste instantie willen wij onze dank betuigen aan alle mensen die bijgedragen hebben aan de verwezenlijking van dit werk. Wij willen ons speciaal richten tot de begeleiders en de andere leeraars die ons holpen. Zij hebben ons alle hulp en begeleiding gegeven die we nodig hadden.*

*Wij hopen dat u als lezer dit werk met plezier leest en dat het aan uw interesses voldoet.*

# Inhoud

## *Inleiding*

Als leerlingen van het laatste jaar Industriële Wetenschappen moet men een geïntegreerde proef samenstellen. Als onderwerp hebben wij iets gekozen dat verband houdt met ruimtevaart, nl. baanbeschrijvingen.
Ten eerste zullen wij beginnen met de basis, de wiskundige omschrijving van kegelsneden zoals de ellips, de hyperbool en de parabool.
Vervolgens zullen de wetten van Kepler en de andere algemene wetten besproken worden om de invloed op een baan in de ruimte van andere hemellichamen te formuleren. Daarna worden de mogelijke interplanetaire banen en de correctie op die banen besproken en uitgediept.
Tenslotte worden deze wetten uitgewerkt in een computerprogramma. Dit programma zal aan de hand van bepaalde parameters de baan van de planeten en objecten, zoals ruimteschepen die interplanetaire banen beschrijven, kunnen berekenen.

# 1 Beginselen van de astronautiek

## 1.1 Wiskundige betekenis van kegelsneden

Elk lichaam dat zich in de ruimte bevindt, beweegt volgens de lijn van een kegeldoorsnede. Ook de lichamen die wij (de mens) de ruimte zullen insturen zullen dit doen. Daarom is het belangrijk dat we de eigenschappen van alle mogelijke kegelsneden bestuderen. Dit zullen we doen in dit hoofdstuk.

Die drie kegelsneden zijn de ellips, de hyperbool en de parabool.

De ellips is de meetkundige plaats (verzameling) van de punten, waarvan de som van de afstanden tot 2 vaste punten (brandpunten) constant is. Zijn $F_1$ en $F_2$ de brandpunten, 2a de waarde van de constante som, dan geldt voor elk punt Q van de ellips: $|F_1Q|+|F_2Q|=2a$ (Fig. 1.1.2 : de ellips)
De hyperbool is de meetkundige plaats van de punten, waarvan het verschil van de afstanden tot 2 vaste punten (brandpunten) constant is. Zijn $F_1$ en $F_2$ de brandpunten, 2a de waarde van het constante verschil, dan geldt voor elk punt Q van de hyperbool: $|F_1Q|+|F_2Q|=\pm 2a$ (Fig. 1.1.4 : de hyperbool)
De parabool is de meetkundige plaats van de punten, die op een gelijke afstand liggen van een vast punt (brandpunt: F) en van een rechte(de richtlijn genoemd) die verticaal op de symmetrieas staat. (Fig. 1.1.3 : de parabool)

Alle kegelsneden voldoen aan de vergelijking:

$$r = \frac{p}{1+\varepsilon\cos\varphi} \tag{1.1.1}$$

r = voerstraal
p = “parameter” van de kegelsnede
ε = numerieke excentriciteit
φ = pericentrumhoek

(zie Appendix)

In deze vergelijking is:

| | | |
|---|---|---|
| voor de cirkel | $p = r$ | $\varepsilon = 0$ |
| voor de ellips | $p = a(1-\varepsilon^2)$ | $0 < \varepsilon < 1$ |
| voor de parabool | $p = p$ | $\varepsilon = 1$ |
| voor de hyperbool | $p = a(\varepsilon^2-1)$ | $\varepsilon > 1$ |

Voor de cirkel vallen de twee brandpunten samen met het middelpunt, overeenkomende met $\varepsilon = 0$

| | |
|---|---|
| $F_1,F_2$ | de brandpunten |
| $PM = a$ | de halve grote as |
| $MB = b$ | de halve kleine as |
| P | het pericentrum |
| A | het apocentrum |
| $F_1Q = r$ | de voerstraal |
| $\angle PF_1Q = \varphi$ | de pericentrumhoek |
| M | het midden |

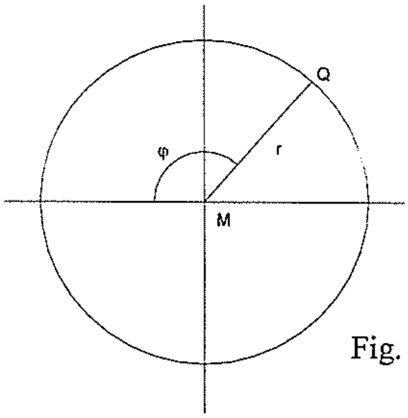


Fig. 1.1.1 : de cirkel

De ellips heeft twee brandpunten $F_1$ en $F_2$.
De afstand van een brandpunt tot het middelpunt is: $F_1M = F_2M = a\varepsilon$
De *numerieke excentriciteit* van een ellips is:

$$\varepsilon = \frac{\sqrt{a^2 - b^2}}{a}$$

Het punt P noemt men in het algemeen het *pericentrum*; wanneer de zon of de aarde in het brandpunt staat, dan noemt men dit punt respectievelijk het *perihelium* of het *perigeum*.
Het punt A noemt men het *apocentrum*, of, wanneer de zon of de aarde in het brandpunt staat, het *aphelium* of het *apogeum*.

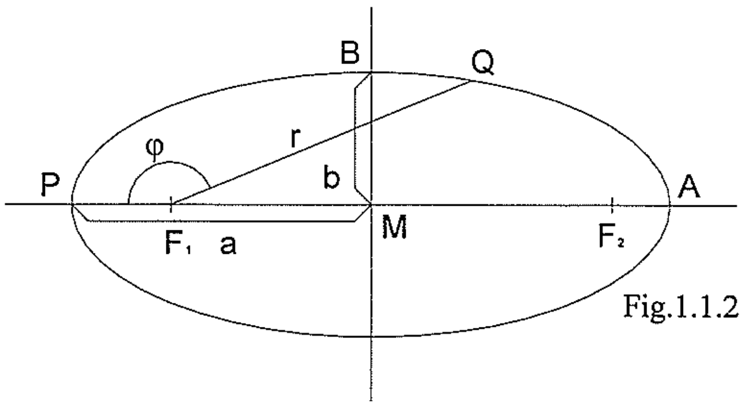


Fig.1.1.2 : de ellips

De parabool heeft maar één brandpunt, de lengte van de grote as is oneindig.

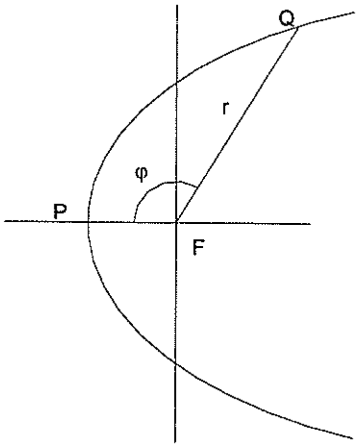



Fig.1.1.3 :de parabool

De hyperbool bestaat uit twee takken, die symmetrisch liggen ten opzichte van het middelpunt M. De excentriciteit is:

$$\varepsilon = \frac{\sqrt{a^2 + b^2}}{a}$$

De richting van een oneindig ver punt van de hyperbool wordt aangeduid door de richting van de asymptoot MC, waarvoor geldt:

$$tg\alpha = \sqrt{\varepsilon^2 - 1} = \sqrt{\frac{a^2 + b^2}{a^2} - 1} = \frac{b}{a}$$

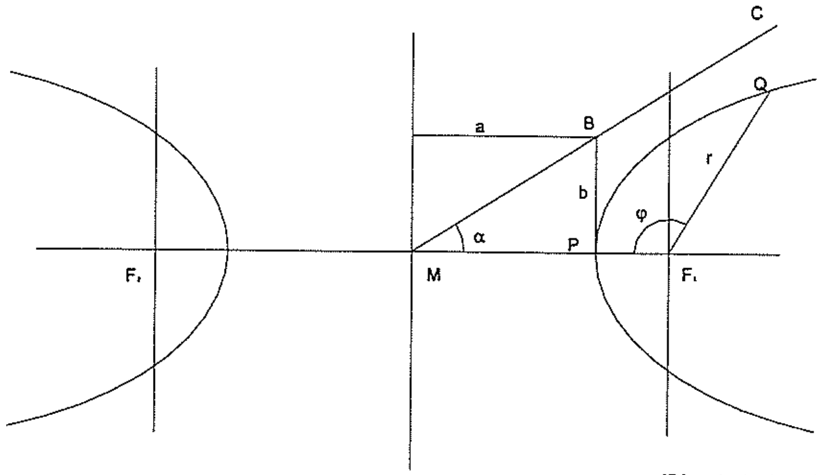



Fig.1.1.4 : de hyperbool

De parabool en de hyperbool zijn *open* krommen, die zich ook op oneindige afstand niet sluiten, in tegenstelling met de *gesloten* krommen, de cirkel en de ellips.
In de ellips is de lengte van de voerstraal $F_1P$ voor $\varphi = 0°$:

$$r_0 = \frac{a(1-\varepsilon^2)}{1+\varepsilon\cos 0°} = \frac{a(1-\varepsilon^2)}{1+\varepsilon} = a(1-\varepsilon)$$

De lengte van de voerstraal $F_1A$ voor $\varphi = 180°$:

$$r_{180} = \frac{a(1-\varepsilon^2)}{1+\varepsilon\cos 180°} = \frac{a(1-\varepsilon^2)}{1-\varepsilon} = a(1+\varepsilon)$$

De verhouding van de parameter $p(\varphi = 90°)$ tot $r_0(\varphi = 0°)$ is:

-bij de cirkel:

$$\frac{p}{r_0} = 1$$

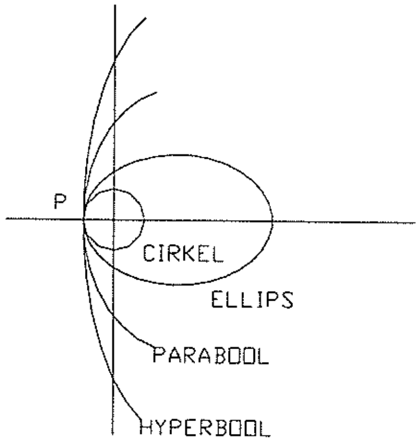



Fig.1.1.5 : de parameters van de kegelsnede

-bij de ellips:

$$\frac{p}{r_0} = \frac{a(1-\varepsilon^2)}{a(1-\varepsilon)} = 1+\varepsilon < 2$$

-bij de parabool:

$$\frac{p}{r_0} = \frac{p}{p/2} = 2$$

-bij de hyperbool:

$$\frac{p}{r_0} = \frac{a(\varepsilon^2-1)}{a(\varepsilon-1)} = 1+\varepsilon > 2$$

In aanmerking nemende, dat voor de cirkel $\varepsilon = 0$ en voor de parabool $\varepsilon = 1$, kan men dus zeggen dat voor elke kegelsnede:

$$\frac{p}{r_0} = 1+\varepsilon$$

## 1.2 De wetten van Kepler

De wetten van Kepler zijn drie wetten die elk een eigenschap geven over de baan die een lichaam in de ruimte beschrijft.
De eerste wet van Kepler zegt ons dat planeten om de zon een ellips beschrijven, met de zon in één van de brandpunten.
De tweede wet van Kepler (ook perkenwet genoemd) vertelt ons dat in een zelfde eenheid van tijd door de voerstraal een zelfde oppervlakte beschreven wordt.

De derde wet van Kepler zegt dat de kwadraten van de omloopstijden van de planeten zich verhouden als de derde machten van de grote assen van de elliptische banen.

We beginnen met de tweede wet van Kepler omdat we deze nodig hebben in de eerste.

### 1.2.1 Tweede wet van Kepler

Stel dat een lichaam m zich om een lichaam M beweegt. Wij nemen M aan als oorsprong van een rechthoekig coördinatensysteem. De twee lichamen trekken elkaar volgens Newton aan met een kracht die gelijk is aan:

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

We gebruiken hier het – teken om aan te duiden, dat de kracht naar M gericht is.

$$F_1 = F \ \sin\varphi = -\frac{GMm}{r^2}\cdot\frac{x}{r}$$

$$F_2 = F\cos\varphi = -\frac{GMm}{r^2}\cdot\frac{y}{r}$$

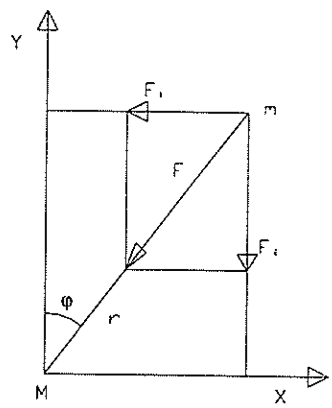



Fig.1.2.1 : aantrekkende massa's

Algemeen:

$$F = m\cdot a$$

Dus kunnen we stellen:

$$F_1 = m\cdot\frac{d^2x}{dt^2}$$

$$F_2 = m\cdot\frac{d^2y}{dt^2}$$

Daaruit volgt:

$$m\cdot\frac{d^2x}{dt^2} = -\frac{GMm}{r^2}\cdot\frac{x}{r}$$

$$m\cdot\frac{d^2y}{dt^2} = -\frac{GMm}{r^2}\cdot\frac{y}{r}$$

$$\frac{d^2x}{dt^2} = -\frac{GM}{r^2}\cdot\frac{x}{r} = -\frac{\mu}{r^3}\cdot x \qquad (1.2.1)$$

$$\frac{d^2y}{dt^2} = -\frac{GM}{r^2}\cdot\frac{y}{r} = -\frac{\mu}{r^3}\cdot y \qquad (1.2.2)$$

We vermenigvuldigen nu (1.2.1) met y en (1.2.2) met x en trekken dan 2 van 1 af.

$$y\frac{d^2x}{dt^2} - x\frac{d^2y}{dt^2} = 0$$
$$\frac{d}{dt}\left(y\frac{dx}{dt} - x\frac{dy}{dt}\right) = 0$$

Geïntegreerd geeft dit:

$$y\frac{dx}{dt} - x\frac{dy}{dt} = 2c$$

(c: constante)

We kunnen deze vorm als volgt omwerken:

$$tg\varphi = \frac{x}{y} \longrightarrow dtg\varphi = d\frac{x}{y}$$
$$\frac{d\varphi}{\cos^2\varphi} = \frac{ydx - xdy}{y^2}$$
$$\cos\varphi = \frac{y}{r} \longrightarrow \frac{1}{\cos^2\varphi} = \frac{r^2}{y^2}$$
$$d\varphi\frac{r^2}{y^2} = \frac{ydx - xdy}{y^2}$$
$$r^2\frac{d\varphi}{dt} = \frac{ydx - xdy}{dt} = 2c \qquad (1.2.3)$$

Wanneer het voorwerp m in de tijd dat zich van plaats $m_1$ naar $m_2$ beweegt, dan beschrijft de voerstraal een driehoekje $m_1m_2M$, waarvan het oppervlak is (Fig. 1.2.2 De hoeksnelheid):

$$\frac{1}{2}r \cdot rd\varphi = \frac{1}{2}r^2 d\varphi\left(+\frac{1}{2}rd\varphi dr\right)$$

waarbij we ½ $r.d\varphi dr$ verwaarlozen.
Vergelijking (1.2.3) zegt dus, dat in de eenheid van tijd door de voerstraal steeds een gelijk oppervlak beschreven wordt, of wel in gelijke tijden gelijke oppervlakken. Men noemt c de oppervlaktesnelheid.
We hebben hier de tweede wet van Kepler afgeleid, ook perkenwet genoemd.
Deze wet geldt niet alleen voor een kracht die omgekeerd evenredig is met het kwadraat van de afstand, maar voor elke centrale kracht m.a.w. een kracht die langs de lijn werkt die de 2 lichamen met elkaar verbindt.

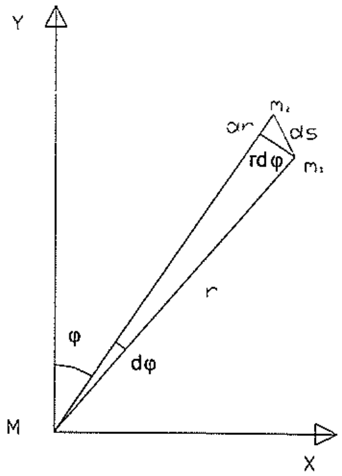



Fig. 1.2.2 : de hoeksnelheid

### 1.2.2 Eerste wet van Kepler

Wanneer we vgl. (1.2.1) met dx/dt vermenigvuldigen en vgl. (1.2.2) met dy/dt, en we tellen bij elkaar op dan krijgen we:

$$\frac{dx}{dt}\frac{d^2x}{dt^2}=\frac{1}{2}\frac{d}{dt}\left(\frac{dx}{dt}\right)^2=-\frac{\mu}{r^3}x\frac{dx}{dt}$$

$$\frac{dy}{dt}\frac{d^2y}{dt^2}=\frac{1}{2}\frac{d}{dt}\left(\frac{dy}{dt}\right)^2=-\frac{\mu}{r^3}y\frac{dy}{dt}$$

$$\frac{1}{2}\frac{d}{dt}\left[\left(\frac{dx}{dt}\right)^2+\left(\frac{dy}{dt}\right)^2\right]=-\frac{\mu}{r^3}\left(x\frac{dx}{dt}+y\frac{dy}{dt}\right) \tag{1.2.4}$$

Nu is:

$$xdx+ydy=rdr$$

En verder:

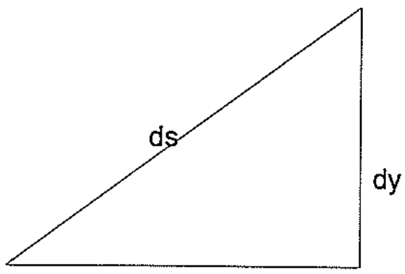



Fig.1.2.3 : de snelheidsdriehoek

$$\left(\frac{dx}{dt}\right)^2+\left(\frac{dy}{dt}\right)^2=\frac{(dx)^2+(dy)^2}{(dt)^2}=\left(\frac{ds}{dt}\right)^2=v^2$$

Vgl. (1.2.4) gaat na vermenigvuldiging met dt over in:

$$\frac{1}{2}d\left(v^2\right)=-\frac{\mu}{r^3}rdr=-\frac{\mu}{r^2}dr=\mu d\left(\frac{1}{r}\right)$$

Na integratie:

$$v^2 = \frac{2\mu}{r} + C \quad . \tag{1.2.5}$$

Uit Fig. (1.2.2) kunnen we aflezen:

$$(ds)^2 = (dr)^2 + (rd\varphi)^2$$

$$v^2 = \left(\frac{ds}{dt}\right)^2 = \left(\frac{dr}{dt}\right)^2 + r^2\left(\frac{d\varphi}{dt}\right)^2$$

$$\left(\frac{dr}{dt}\right)^2 = v^2 - r^2\left(\frac{d\varphi}{dt}\right)^2$$

$$\frac{dt}{dr} = \frac{1}{\sqrt{v^2 - r^2\left(\frac{d\varphi}{dt}\right)^2}}$$

Nu is:

$$\frac{d\varphi}{dr} = \frac{d\varphi}{dt}\frac{dt}{dr}$$

Uit vgl. (1.2.3) volgt:

$$\frac{d\varphi}{dt} = \frac{2c}{r^2} \text{ en : } r^2\left(\frac{d\varphi}{dt}\right)^2 = \frac{4c^2}{r^2}$$

$$d\varphi = \frac{\frac{2c}{r^2}dr}{\sqrt{\frac{2\mu}{r} + C - \frac{4c^2}{r^2}}}$$

Wanneer we stellen:

$$\frac{2c}{r^2}dr = d\left(\frac{\mu}{2c} - \frac{2c}{r}\right) = dz$$

$$\frac{2\mu}{r} + C - \frac{4c^2}{r^2} = \frac{\mu^2}{4c^2} + C - \left(\frac{\mu}{2c} - \frac{2c}{r}\right)^2 = a^2 - z^2$$

dan is $d\varphi = \dfrac{dz}{\sqrt{a^2 - z^2}} = \dfrac{d\left(\frac{z}{a}\right)}{\sqrt{1 - \left(\frac{z}{a}\right)^2}}$

en na integratie:

$$\varphi + C_1 = Bg\sin\frac{z}{a}$$
$$z = a\sin(\varphi + C_1)$$

na substitutie van z en a:

$$\frac{\mu}{2c} - \frac{2c}{r} = \sqrt{\frac{\mu^2}{4c^2} + C} \cdot \sin(\varphi + C_1)$$
$$r = \frac{2c}{\frac{\mu}{2c} - \sqrt{\frac{\mu^2}{4c^2} + C} \cdot \sin(\varphi + C_1)}$$
$$r = \frac{\frac{4c^2}{\mu}}{1 - \sqrt{1 + \frac{4c^2}{\mu^2}C} \cdot \sin(\varphi + C_1)} \qquad (1.2.6)$$

Wanneer we nu stellen:

$$\frac{4c^2}{\mu} = \pm a(1 - \varepsilon^2) \qquad (1.2.7)$$
$$\sqrt{1 + \frac{4c^2}{\mu^2}C} = \varepsilon \qquad (1.2.8)$$

Dan gaat vgl. (1.2.6) over in:

$$r = \frac{\pm a(1 - \varepsilon^2)}{1 - \varepsilon\sin(\varphi + C_1)} \qquad (1.2.9)$$

Deze vergelijking kan nu omgevormd worden naar $r = \frac{p}{1 + \varepsilon\cos\varphi}$, die de brandpuntsvergelijking vormt van een kegelsnede in poolcoördinaten.

Wij zullen $C_1$ in vgl. (1.2.9) nu zo bepalen, dat voor r = minimum, $\varphi$ nul wordt. Nu is r minimum, wanneer sin ($\varphi + C_1$) minimum is, of wel wanneer:

$$\sin(\varphi + C_1) = -1$$
$$\varphi + C_1 = \frac{3\pi}{2} \qquad \varphi = 0 \qquad \text{dus } C_1 = \frac{3\pi}{2}$$
$$\sin\left(\varphi + \frac{3\pi}{2}\right) = -\cos\varphi$$

vgl. (1.2.9) gaat dus over in:

$$r = \frac{\pm a\left(1-\varepsilon^2\right)}{1+\varepsilon\cos\varphi}$$

Hiermee hebben we de *eerste wet* van Kepler bewezen, nl. dat de planeten om de zon ellipsen beschrijven, met de zon in één van de brandpunten.

### 1.2.3 Derde wet van Kepler

De derde wet van Kepler zegt dat de kwadraten van de omloopstijden van de planeten zich verhouden als de derde machten van de grote assen van de elliptische banen. Ook dit is gemakkelijk af te leiden. Wanneer wij de oppervlakte van het elementaire driehoekje $m_1m_2M$ in fig.(1.2.2), dO noemen, en wij noemen O de oppervlakte van de gehele ellips en T de gehele omloopstijd (periode), dan volgt uit vgl. (1.2.3):

$$\frac{r^2 d\varphi}{dt} = \frac{2dO}{dt} = \frac{2O}{T} = 2c$$

Uit vgl. (1.2.8) volgt voor een ellips:

$$2c = \sqrt{\mu a(1-\varepsilon^2)}$$

De oppervlakte van een ellips is:

$$O = \pi ab = \pi a^2\sqrt{1-\varepsilon^2}$$

$$\frac{2\pi a^2\sqrt{1-\varepsilon^2}}{T} = \sqrt{\mu a(1-\varepsilon^2)}$$

$$T = a^{3/2}\mu^{-1/2}2\pi$$

$$T^2 = \frac{4\pi^2}{\mu}a^3$$

De wetten van Kepler gelden alleen als de massa van het centrale lichaam zeer veel groter is dan de massa van het lichaam dat er omheen draait.

Uit vergelijking (1.2.7)

$$\frac{4c^2}{\mu} = \pm a\left(1-\varepsilon^2\right)$$

en vergelijking (1.2.8)

$$\sqrt{1+\frac{4c^2}{\mu^2}C} = \varepsilon$$

kunnen we afleiden dat:

$$C = \frac{\mu^2}{4c^2}\left(\varepsilon^2 - 1\right) = \frac{\mu^2\left(\varepsilon^2 - 1\right)}{\pm \mu a\left(1-\varepsilon^2\right)} = \pm \frac{\mu}{a} \qquad (1.2.10)$$

We kunnen nu m.b.v. vergelijking (1.2.5)

$$v^2 = \frac{2\mu}{r} + C$$

de snelheid berekenen en we vinden dan dat:

$$v^2 = \mu\left(\frac{2}{r} \pm \frac{1}{a}\right) \qquad (1.2.11)$$

Waarin het – teken voor ellipsen geldt en het + teken voor hyperbolen.
We merken op, dat de excentriciteit van de kegelsnede in deze vergelijking niet voorkomt, deze heeft dus geen invloed op de snelheid.

## 1.3 De energiebalans bij elliptische bewegingen

In dit hoofdstuk zullen we nagaan hoe de energiebalans wordt bij de elliptische beweging.
Stel, dat we een massaeenheid hebben, die zich op een afstand r van een massa M bevind.
De zwaartekrachtconstante van M is :

$$fM = \mu$$

De aantrekkingskracht die de éénheidsmassa (m = 1) van de massa M ondervindt, is:

$$F = \frac{fMm}{r^2} = \frac{\mu}{r^2}$$

Als we de éénheidsmassa een afstand dr van de massa M verwijderen, moeten we een arbeid dW verrichten die gelijk is aan:

$$dE_p = F\,dr = \frac{\mu}{r^2}dr$$

De totale arbeid die nodig is om de éénheidsmassa van $r_1$ naar $r_2$ te brengen ($r_1 < r_2$), is dan :

$$E_p = \int_{r_1}^{r_2} \frac{\mu}{r^2}dr = -\mu\left(\frac{1}{r_2} - \frac{1}{r_1}\right) \qquad (1.3.1)$$

Wanneer $r_2$ oneindig ver van M ligt, is de arbeid:

$$E_p = -\mu\left(\frac{1}{\infty} - \frac{1}{r_1}\right) = \frac{\mu}{r_1} \qquad (1.3.2)$$

Omgekeerd is het arbeidsvermogen van de eenheidsmassa op een oneindige afstand naar een afstand $r_1$ gelijk aan:

$$E_p = -\frac{\mu}{r_1}$$

Dit arbeidsvermogen noemt men de *potentiële energie* of energie van plaats van de éénheidsmassa in het zwaartekrachtveld van M op een afstand $r_1$ van M.

Om de eenheidsmassa een snelheid v te geven, is, zoals gekend, een energie nodig die gelijk is aan:

$$E_k = \frac{v^2}{2}$$

Men noemt deze energie de *kinetische energie*, of energie van beweging van de eenheidsmassa.

Wanneer nu een eenheidsmassadeeltje een elliptische baan beschrijft om de massa M, en de voerstraal is op een zeker moment r en de snelheid is v dan is de potentiële energie van dit deeltje:

$$E_p = -\frac{\mu}{r}$$

Als nu de halve grote as van de elliptische baan a is, dan is de snelheid volgens vergelijking (1.2.11):

$$v = \sqrt{\mu\left(\frac{2}{r} - \frac{1}{a}\right)}$$

en dus de kinetische energie:

$$E_k = \frac{v^2}{2} = \frac{1}{2}\mu\left(\frac{2}{r} - \frac{1}{a}\right)$$

Daaruit volgt dat de totale energie van dit deeltje dan gelijk wordt aan:

$$E = E_p + E_k = -\frac{\mu}{r} + \frac{1}{2}\mu\left(\frac{2}{r} - \frac{1}{a}\right) = -\frac{\mu}{2a}$$

We zien, dat r uit deze vergelijking verdwenen is, wat te verwachten was. Er wordt in de baan aan het massadeeltje om de massa M geen energie toegevoerd en ook geen energie onttrokken, dus de totale energie blijft onafhankelijk van de voerstraal.
We zien dus, dat de kinetische energie en de potentiële energie elkaar in evenwicht houden, als de ene toeneemt zal de andere dalen en omgekeerd.
In het pericentrum is de potentiële energie minimaal en de kinetische energie maximaal. In het apocentrum is het juist omgekeerd.

Hieronder zie je de energiebalans voor een ellips met halve grote as a en een excentriciteit gelijk aan ε. (φ is de pericentrumhoek)

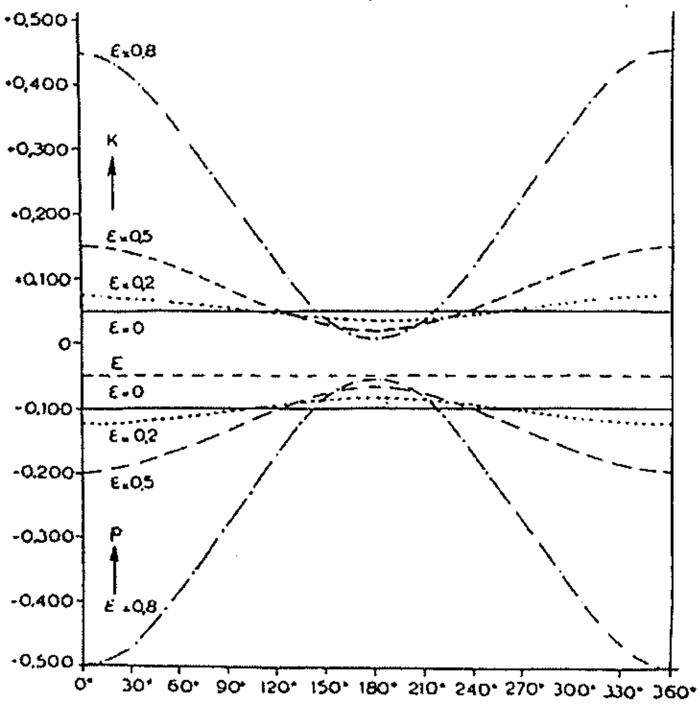



Fig. 1.3.1 Het verloop van de energie bij de elliptische beweging

In Fig. 1.3.1 is de verhouding tussen potentiële energie en kinetische energie uitgezet voor ellipsen met excentriciteit 0,8 ; 0,5 ; 0,2 en 0 en gelijke grote assen.
Een merkwaardig punt is het punt waar de kinetische energie precies de helft van de potentiële energie is. De voerstraal wordt dan gelijk aan de halve grote as. En $\varphi$ word dan gelijk aan Bgcos (-ε).
De totale energie E blijft voor alle punten van de ellips gelijk.
In Fig. 1.3.1 is ook te zien dat $E_k$ het spiegelbeeld is van $E_p$, wat nodig is omdat de totale energie E constant zou blijven.

## 1.4 Snelheden bij de elliptische beweging

We zullen de snelheden berekenen in enkele bijzondere punten van de baan.
Bij een ellips krijgen we voor r, dus $\varphi$=0°, ofwel in het pericentrum :

$$v_0^2 = \mu\left(\frac{2}{r_0} - \frac{1}{a}\right)$$

We weten dat :

$$r_0 = a(1-\varepsilon)$$

Dus:

$$v_0^2 = \mu\left(\frac{2}{a(1-\varepsilon)} - \frac{1-\varepsilon}{a(1-\varepsilon)}\right)$$

$$v_0^2 = \frac{\mu}{a}\frac{1+\varepsilon}{1-\varepsilon} \qquad (1.4.1)$$

Voor $r_{180}$, dus voor $\varphi = 180°$, of anders gezegd in het apocentrum geldt:

$$v_{180}^2 = \mu\left(\frac{2}{r_{180}} - \frac{1}{a}\right)$$

We weten dat :

$$r_{180} = a(1+\varepsilon)$$

Dus:

$$v_{180}^2 = \mu\left(\frac{2}{a(1+\varepsilon)} - \frac{1+\varepsilon}{a(1+\varepsilon)}\right)$$

$$v_{180}^2 = \frac{\mu}{a}\frac{1-\varepsilon}{1+\varepsilon} \qquad (1.4.2)$$

Uit vergelijking (1.4.1) en (1.4.2) halen we nu dat:

$$\frac{v_0^2}{v_{180}^2} = \frac{\mu}{a}\frac{1+\varepsilon}{1-\varepsilon} \Big/ \frac{\mu}{a}\frac{1-\varepsilon}{1+\varepsilon} = \frac{(1+\varepsilon)^2}{(1-\varepsilon)^2}$$

$$\frac{v_0}{v_{180}} = \frac{1+\varepsilon}{1-\varepsilon} \qquad (1.4.3)$$

En vervolgens, aangezien ook:

$$\frac{r_{180}}{r_0} = \frac{a(1+\varepsilon)}{a(1-\varepsilon)} = \frac{1+\varepsilon}{1-\varepsilon} \qquad (1.4.4)$$

Uit vergelijking (1.4.3) en (1.4.4) halen we dat :

$$\frac{v_0}{v_{180}} = \frac{r_{180}}{r_0}$$

$$v_0 r_0 = v_{180} r_{180} \qquad (1.4.5)$$

Uit vergelijking (1.4.4) kunnen we de excentriciteit afleiden.

$$\varepsilon = \frac{r_{180} - r_0}{r_{180} + r_0} \qquad (1.4.6)$$

We kunnen dus met deze vergelijking de excentriciteit van een baan berekenen als we $r_0$ en $r_{180}$ kennen.

## 1.5 De circulaire en de parabolische snelheid

Uit vergelijking (1.2.11) luidde:

$$v^2 = \mu\left(\frac{2}{r} - \frac{1}{a}\right)$$

Voor de cirkel is: a = r

Dus:

$$v_c^2 = \frac{\mu}{r} \qquad (1.5.1)$$

voor de parabool is: a = ∞, dus 1/a = 0

dus:

$$v_p^2 = \frac{2\mu}{r} \quad \text{(parabolische snelheid)} \qquad (1.5.2)$$

De circulaire snelheid is de snelheid waarmee een lichaam een cirkelvormige baan om een centraal aantrekkend lichaam beschrijft. (bv. een satelliet om de aarde) De circulaire

snelheid is ook de snelheid waarbij de aantrekkingskracht tussen de twee lichamen en de middelpuntvliedende kracht in evenwicht worden gehouden opdat het lichaam in zijn baan zou blijven.
Stel nu dat een lichaam met massa m een cirkelvormige baan beschrijft om een lichaam met massa M.
Dan is de aantrekkingskracht:

$$F_1 = G\frac{mM}{r^2} \rightarrow F_1 = \frac{m\mu}{r^2}$$

En de middelpuntvliedende kracht:

$$F_2 = \frac{mv^2}{r}$$

Beide moeten dus gelijk aan elkaar zijn opdat het lichaam m in zijn baan om lichaam M zou blijven. Hieruit volgt:

$$\frac{m\mu}{r^2} = \frac{mv^2}{r}$$

$$v_c^2 = \frac{\mu}{r}$$

Dit stemt overeen met de circulaire snelheid.
De parabolische snelheid noemt men ook wel de *ontsnappingssnelheid,* aangezien de parabool zich oneindig ver uitstrekt, zal een lichaam dat zich met de parabolische snelheid voortbeweegt zich oneindig ver verwijderen van een centraal lichaam, als er geen andere aantrekkende lichamen aanwezig zijn. Het zal dus ontsnappen aan dat centraal lichaam.

*We zagen in een vorig stuk dat de arbeid*, nodig om een éénheidsmassa van een afstand $r_1$ van een massa M naar oneindig te brengen gelijk is aan:

$$W = G\frac{M}{r_1} = \frac{\mu}{r_1}$$

Wanneer we deze arbeid willen verrichten door het deeltje een snelheid te geven, dan moet de kinetische energie gelijk zijn aan de te verrichten arbeid.

$$\frac{1}{2}v^2 = \frac{\mu}{r_1}$$

$$v = \sqrt{\frac{2\mu}{r_1}}$$

Dit is in overeenstemming met de vergelijking van de parabolische snelheid.
Wanneer we nu vanop het oppervlak van een planeet willen vertrekken, dus wanneer $r_1$ gelijk wordt aan de straal van de planeet R dan wordt de formule:

$$v = v_{po} = \sqrt{\frac{2\mu}{R}}$$

We spreken nu over de ontsnappingssnelheid vanop het oppervlak van de planeet. Voortaan aangeduid met $v_{po}$.

Uit vergelijking (1.5.1) en (1.5.2) volgt nog:

$$\frac{v_p^2}{v_c^2} = \frac{2\mu}{r} \bigg/ \frac{\mu}{r} = 2$$

$$\frac{v_p}{v_c} = \sqrt{2}$$

Dit wil dus zeggen dat op eenzelfde afstand van het centrale punt de ontsnappingssnelheid $\sqrt{2}$ maal groter is dan de circulaire snelheid.

### 1.5.1 De snelheid voor een circulaire baan

De arbeid die men moet leveren om een lichaam de circulaire snelheid te geven in een baan met straal r bestaat uit:

a) de arbeid die men moet verrichten om het lichaam van afstand R (gerekend vanaf middelpunt aarde) naar afstand r op te heffen,(potentiële energie).
   Deze arbeid is volgens vergelijking (1.3.1) per massa-eenheid:
   $$E_p = \mu\left(\frac{1}{R} - \frac{1}{r}\right)$$

b) de arbeid die men moet verrichten om het lichaam de snelheid te geven.(kinetische energie)
   $$E_k = \frac{1}{2} m v_c^2$$

en $\quad v_c^2 = \frac{\mu}{r}$

dus $\quad E_k = \frac{1}{2}\frac{\mu m}{r}$

De totale arbeid die men moet verrichten is dus

$$E = E_p + E_k = \mu m\left[\left(\frac{1}{R} - \frac{1}{r}\right) + \frac{1}{2r}\right] = \mu m\left(\frac{1}{R} - \frac{1}{2r}\right) \qquad (1.5.3)$$

Om het lichaam in een circulaire baan te brengen, moeten we het een snelheid v geven, zodat:

$$\frac{1}{2} m v^2 = \mu m\left(\frac{1}{R} - \frac{1}{2r}\right)$$

dus $\quad v^2 = 2\mu\left(\frac{1}{R} - \frac{1}{2r}\right) = \frac{2\mu}{R}\left(1 - \frac{R}{2r}\right)$

$$v^2 = v_{po}^2\left(1 - \frac{R}{2r}\right) \qquad (1.5.4)$$

Voor r = ∞ vinden we:

$v^2 = v_{po}^2$ (ontsnappingssnelheid op aardoppervlak)

en voor r = R vinden we:

$v^2 = \frac{v_{po}^2}{2} = v_c^2$ (circulaire snelheid op aardoppervlak)

In onderstaande tabel zijn voor verschillende hoogten boven de aarde h,(h = r − R), de circulaire snelheid $v_c$ en de snelheid v, die nodig is om de circulaire snelheid te bereiken, berekend. Tevens is daarbij de omloopstijd T aangegeven, die bij die hoogte behoort.

| h km | $v_c$ km/sec | v km/sec | T |
|---|---|---|---|
| 0 | 7.91 | 7.91 | 1h 24min 15sec =5055sec = $v_{co}$ |
| 100 | 7.85 | 7.98 | 1h 26min 12sec =5172sec |
| 200 | 7.79 | 8.03 | 1h 28min 14sec =5294sec |
| 500 | 7.62 | 8.20 | 1h 34min 21sec =5661sec |
| 1000 | 7.36 | 8.44 | 1h 44min 52sec =6292sec |
| 2000 | 6.91 | 8.81 | 2h 6min 54sec =7614sec |
| ∞ | 0 | 11.19 | ∞ = $v_{co}$ |

### 1.5.2 De ontsnappingssnelheid en de zwaartekrachtversnelling

De versnelling door de zwaartekracht aan de oppervlakte van een planeet is:

$$g = \frac{\mu}{r^2}$$

wanneer r de straal van de planeet is.
Zoals reeds gekend is:

$$\mu = fM$$

en voor M kunnen we schrijven:

$$M = \gamma\, r^3$$

M.a.w. de massa is evenredig met de derde macht van de straal. De evenredigheidsconstante γ hangt voor een bolvormig lichaam alleen van het soortelijk gewicht af.
Voor planeten met gelijk soortelijk gewicht kunnen wij dus schrijven:

$$g = \frac{f\gamma\, r^3}{r^2} = c_1 r$$

Dus de versnellingen door de zwaartekracht aan de oppervlakte van *twee planeten* zijn evenredig met hun stralen. Voor de ontsnappingssnelheid aan de oppervlakte van een planeet geldt:

$$v_{po}^2 = \frac{2\mu}{r} = \frac{2f\gamma\, r^3}{r} = c_2 r^2$$

De ontsnappingssnelheid en ook de circulaire snelheid aan het oppervlak zijn dus, bij gelijk soortelijk gewicht, evenredig met de straal van de planeet.

### 1.5.3 Voorbeeld snelheid

In de eerste plaats zullen wij de ontsnappingssnelheid aan de oppervlakte van de aarde berekenen met de volgende formule:

$$v_{po}^2 = \frac{2\mu}{R}$$ Hierbij is: R = 6368 km en $\mu_A = 3{,}989.\ 10^{20}\ cm^3/sec^2$

zodat de ontsnappingssnelheid moet zijn:

$$v_{po} = \sqrt{\frac{2.3{,}989.10^{20}}{6{,}368.10^{8}}} = 1{,}119.10^{6}\ cm/_{sec} = 11{,}19\ km/_{sec}$$

Indien de aarde het enige aantrekkende lichaam in het heelal zou zijn, dan zou een voorwerp dat met deze snelheid de oppervlakte van de aarde verlaat zich oneindig ver verwijderen. Maar we weten dat dit niet zo is en dat we dus rekening moeten houden met de aantrekkings – kracht van de zon. Tengevolge daarvan zal een lichaam dat de aarde met de ontsnappingssnelheid verlaat, zich niet oneindig ver verwijderen, maar uiteindelijk een elliptische baan om de zon gaan beschrijven.

Volgens vergelijking 1.5.1 is de circulaire snelheid:

$$v_c^2 = \frac{\mu}{r}$$

Hierbij is: r = 6368 km + 500 km = 6868 km

$\mu_A = 3{,}989\ .\ 10^{20}\ cm^3/_{sec^2}$

Dus is de circulaire snelheid gelijk aan:

$$v_c = \sqrt{\frac{3{,}989.10^{20}}{6{,}868.10^{8}}} = 7{,}62.10^{5}\ cm/_{sec} = 7{,}62\ km/_{sec}$$

Van Spoetnik 3 is bekend, dat ten tijde van het lanceren de grootste afstand H tot de aarde 1880 km bedroeg en de kleinste afstand h = 230 km.
De Spoetnik beschreef dus geen cirkel maar een ellips.
Hoe groot was de grote as en de excentriciteit van deze ellips?

Uit de gegevens kan men afleiden dat:
2a = 2R + H + h

of anders geschreven:

$$a = R + \frac{H+h}{2}$$

Waaruit men de halve grote as kan afleiden:

$$a = 6370 + \frac{1880+230}{2} = 7425km = 7{,}425.10^{8}cm$$

Volgens vergelijking 1.4.6 is:

$$\varepsilon = \frac{r_{180} - r_0}{r_{180} + r_0}$$

Hierbij is:

$$r_{180} = R + H = 6370 + 1880km = 8250km$$
$$r_0 = R + h = 6370 + 230km = 6600km$$

Dus kan men afleiden dat:

$$\varepsilon = \frac{8250-6600}{8250+6600} = 0{,}111$$

Hoeveel is de snelheid van Spoetnik 3 in zijn apogeum en hoeveel in het perigeum?
Uit de vergelijking 1.4.2 weten we dat:

$$v_{180} = \sqrt{\frac{\mu_A}{a}\frac{1-\varepsilon}{1+\varepsilon}} = \sqrt{\frac{3{,}989.10^{20}}{7{,}425.10^{8}}\frac{1-0{,}111}{1+0{,}111}} = 6{,}56\, km/_{sec}$$

Uit de vergelijking 1.4.3 kunnen we $v_0$ berekenen:

$$\frac{v_0}{v_{180}} = \frac{1+\varepsilon}{1-\varepsilon}$$

dus:

$$v_0 = v_{180}\frac{1+\varepsilon}{1-\varepsilon} = 6{,}56\frac{1{,}111}{0{,}889}\, km/_{sec} = 6{,}56.1{,}25\, km/_{sec} = 8{,}19\, km/_{sec}$$

## 1.6 Het drie-lichamenprobleem

We hebben tot nu toe steeds aangenomen dat een voorwerp slechts onder de aantrekking valt van één lichaam, b.v. een satelliet viel slechts onder de aantrekking van de aarde.
In werkelijkheid zal een voorwerp in de ruimte aan de aantrekking van vele lichamen onderworpen zijn. Beweegt men nu vanop de aarde naar de maan, dan zal de aantrekking van de aarde verminderen, die van de zon zal ongeveer gelijk blijven en die van de maan zal toenemen.
In het begin is de aantrekkingskracht van de aarde overheersend en voor niet al te nauwkeurige berekeningen kunnen we de aantrekkingskracht van de zon en de maan verwaarlozen.
In de reis naar de maan krijgen we achtereenvolgens in volgorde van aantrekking:

De aarde
De aarde en de zon
De zon, de aarde en de maan
De zon en de maan
De maan

In vorige hoofdstukken hebben we wetten gezien die bedoeld waren voor twee lichamen. Die vergelijkingen waren niet al te ingewikkeld.
De vergelijkingen die we zouden krijgen voor meer dan twee lichamen (drie-lichamenprobleem) zouden te ingewikkeld zijn.
De vergelijkingen die men zou krijgen zoude slechts op te lossen zijn door trapsgewijze integratie. Dit is een buiten omslachtig werk en is een werk voor supersnelle moderne machines.

Dit drie-lichamenprobleem kunnen we mits vereenvoudigingen, praktischer benaderen. B.v. men heeft drie lichamen $M_1$, $M_2$ en m, en m is zo klein dat ze geen aantrekking uitoefent op $M_1$ en $M_2$. Dit maakt de toestand al een stuk eenvoudiger.
Gelukkig komt deze toestand zéér veel voor in de ruimte. Meestal is het dat $M_2 >> m$ maar ook dat $M_1 >> M_2$. (b.v. zon, planeet, ruimteschip)
Bij de behandeling van circulaire bewegingen zagen we dat de aantrekkingskracht precies gelijk moet zijn aan de middelpuntvliedende kracht. Bij een drie-lichamenprobleem met het derde lichaam zéér klein kunnen we dit ook aantreffen.
Bij een centraal lichaam is de meetkundige plaats van de punten waar dit het geval is een cirkel. Bij het drie-lichamenprobleem kan dit slechts in enkele gevallen. In deze punten heffen de aantrekkingskrachten en de middelpuntvliedende krachten elkaar op.

We zullen nu nagaan hoeveel dergelijke punten er zijn en waar ze gelegen zijn.

In fig.1.6.1 zijn de aantrekkende lichamen M en m. Die lichamen hebben een gemeenschappelijk zwaartepunt Z en die lichamen beschrijven om dit zwaartepunt een cirkelbaan.
We stellen de afstand $\overline{Mm} = 1$.
De afstand $\overline{MZ}$ noemen we r en de afstand $\overline{mZ}$ noemen we R. Dus is:

R + r = 1

Volgens de hefboomregel:

$$\frac{m}{M} = \frac{r}{R} \qquad (1.6.1)$$

Daaruit volgt:

$$r = \frac{m}{M+m} \qquad R = \frac{M}{M+m} \qquad (1.6.2)$$

We beschouwen nu in het zwaartekrachtveld van M en m een willekeurig punt P dat met de lijn $\overline{Mm}$ om het zwaartepunt Z meeroteert. Op P werkt de kracht $F_1$ veroorzaakt door M, de kracht $F_2$ veroorzaakt door m en de middelpuntvliedende kracht C, veroorzaakt door Z

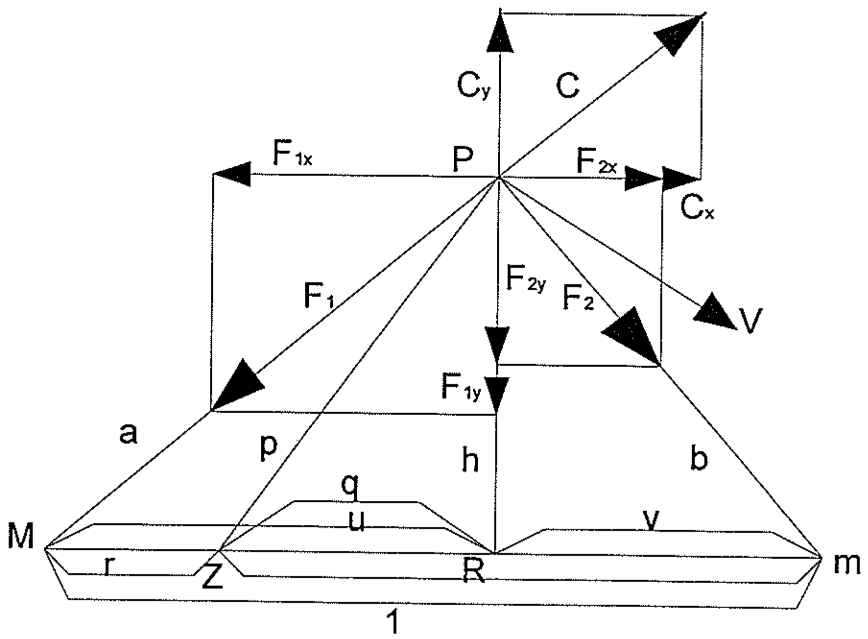



Fig.1.6.1 : de krachten bij een drie-lichamensysteem

De afstand van P tot M noemen wij a, de afstand P tot m noemen wij b. De lengte van de loodlijn uit P op de lijn Mm is h. De afstand van het voetpunt van die loodlijn tot M is u, en tot m is v.
Dus:

$u + v = 1$

Tenslotte is de afstand van P tot het zwaartepunt Z gelijk aan p.
We kunnen nu stellen:

$$F_1 = \frac{fM}{a^2} \qquad F_2 = \frac{fm}{b^2} \qquad C = \frac{v^2}{p}$$

V is hier de omwentelingssnelheid van het punt p om het zwaartepunt Z.
Aangezien P met M en m mederoteert, is de omwentelijngstijd T gelijk aan de omwentelingstijd van M en m om Z. (zie eindwerk Arenens Nick en Debever Pieter, klas: 614 , IW, '98-'99, hoofdstuk 8)

$$T = 2\pi(fM + fm)^{-\frac{1}{2}} (\overline{mM})^{\frac{3}{2}}$$

Aangezien de afstand $\overline{mM}$ op 1 gesteld is, krijgen we:

$$T = 2\pi(fM + fm)^{-\frac{1}{2}}$$

De snelheid V is nu gelijk aan:

$$V = \frac{2\pi p}{T} = p(fM + fm)^{\frac{1}{2}}$$

Dus:

$$C = \frac{V^2}{p} = pf(M + m)$$

We ontbinden nu de drie krachten en de richtingen loodrecht op en evenwijdig aan de lijn $\overline{Mm}$.
We krijgen dan:

$$F_{1x} = \frac{fM}{a^2}\frac{u}{a} \quad F_{2x} = \frac{fm}{b^2}\frac{v}{b} \quad C_x = qf(M + m)$$
$$F_{1y} = \frac{fM}{a^2}\frac{h}{a} \quad F_{2y} = \frac{fm}{b^2}\frac{h}{b} \quad C_y = hf(M + m)$$

De evenwichtsvoorwaarden zijn nu:

$$-F_{1x} + F_{2x} + C_x = 0$$
$$-F_{1y} - F_{2y} + C_y = 0$$

Ofwel

$$-\frac{fM}{a^2}\frac{u}{a} + \frac{fm}{b^2}\frac{v}{b} + qf(M + m) = 0$$

$$-\frac{fM}{a^2}\frac{h}{a}-\frac{fm}{b^2}\frac{h}{b}+hf(M+m)=0$$

Aangezien: $f \neq 0$,kunnen we delen door $f$

$$-M\frac{u}{a^3}+m\frac{v}{b^3}+q(M+m)=0 \qquad (1.6.3)$$

$$-M\frac{h}{a^3}-m\frac{h}{b^3}+h(M+m)=0 \qquad (1.6.4)$$

Er zijn nu twee gevallen mogelijk:

$h=0$

$h\neq 0$

Het geval h = 0 betekent dat het punt op de lijn $\overline{Mm}$ of het verlengde ervan ligt.
We inderscheiden drie gevallen:

a: $P_1$ ligt tussen M en m
b: $P_2$ ligt rechts van m
c: $P_3$ ligt links M

In geval a (Fig. 1.6.2) is:

$$a+b=1$$
$$u=a$$
$$v=b$$
$$q=u-r=a-\frac{m}{M+m}$$

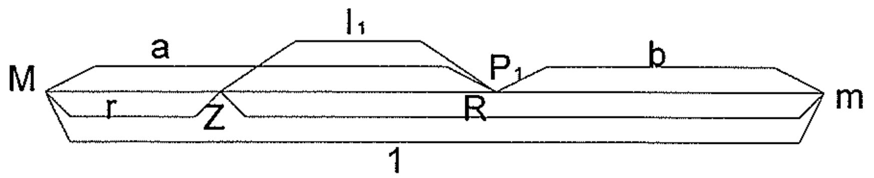



Fig.1.6.2 : het eerste libratiepunt

Vergelijking 1.6.3 gaat dus over in:

$$-\frac{M}{a^2}+\frac{m}{b^2}+\left(a-\frac{m}{M+m}\right)(M+m)=0$$

Door toepassing van vergelijkingen 1.6.1 en 1.6.2 kunnen we deze vergelijking omwerken naar:

$$-\frac{mR}{ra^2}+\frac{mr}{rb^2}+(a-r)\left(\frac{mR}{r}+m\right)=0$$
$$-\frac{R}{a^2}+\frac{r}{b^2}+(a-r)(R+r)=0$$
$$-\frac{R}{a^2}+\frac{r}{b^2}+a-r=0 \tag{1.6.5}$$

We noemen nu de afstand $\overline{P_1Z}=l_1$.
Daaruit volgt dan:

$$(r+l_1)-r-\frac{R}{(r+l_1)^2}+\frac{r}{(R-l_1)^2}=0$$
$$l_1-\frac{R}{(r+l_1)^2}+\frac{r}{(R-l_1)^2}=0 \tag{1.6.6}$$

Voor geval b (Fig. 1.6.3) is:

$$u=a$$
$$v=-b$$
$$q=l_2$$

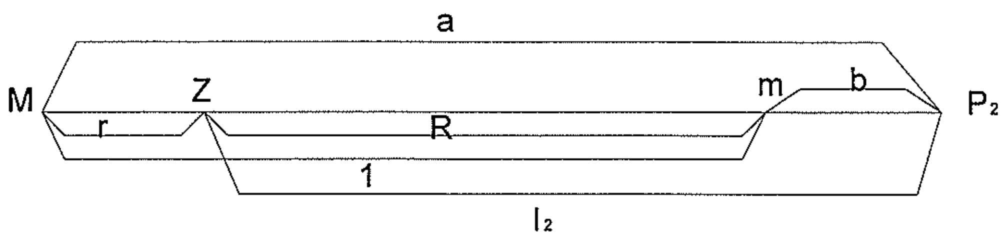



Fig.1.6.3 : het tweede libratiepunt

We leiden dan weer op dezelfde manier af uit vergelijking 1.6.5:

$$l_2-\frac{R}{(l_2+r)^2}-\frac{r}{(l_2-R)^2}=0 \tag{1.6.7}$$

Voor geval c (Fig. 1.6.4) is:

$$b-a=1$$
$$u=-a$$
$$v=b$$
$$q=-l_3$$

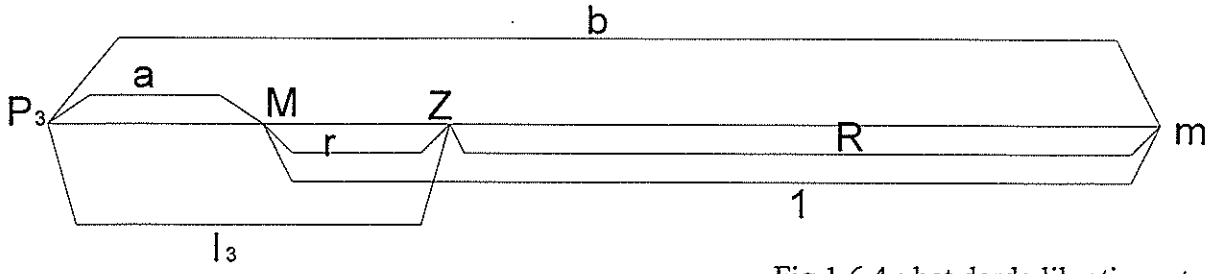



Fig.1.6.4 : het derde libratiepunt

We leiden dan weer op dezelfde manier af uit vergelijking 1.6.5:

$$l_3 - \frac{R}{(l_3 - r)^2} - \frac{r}{(l_3 + R)^2} = 0 \tag{1.6.8}$$

We krijgen nu voor het geval: $h \neq 0$.
We mogen dan vergelijking 1.6.4 delen door h.
We krijgen dan:

$$M\left(1 - \frac{1}{a^3}\right) + m\left(1 - \frac{1}{b^3}\right) = 0$$

m.b.v. vergelijking 1.6.1:

$$\frac{m}{M} = \frac{r}{R}$$

Krijgen we :

$$R\left(1 - \frac{1}{a^3}\right) + r\left(1 - \frac{1}{b^3}\right) = 0 \tag{1.6.9}$$

Uit vergelijking 1.6.3 kunnen we afleiden door eliminatie van M en m:

$$-R\frac{u}{a^3} + r\frac{v}{b^3} + q = 0 \tag{1.6.10}$$

q kunnen we anders schrijven als:

$$\begin{aligned} q &= u - r \\ &= u - ur + ur - r \\ &= u(1-r) - r(1-u) \\ &= uR - vr \end{aligned}$$

Ingevuld in vergelijking 1.6.10 geeft dit:

$$-R\frac{u}{a^3}+r\frac{v}{b^3}+uR-vr=0$$

$$uR\left(1-\frac{1}{a^3}\right)-vr\left(1-\frac{1}{b^3}\right)=0 \qquad (1.6.11)$$

Wanneer we vergelijking 1.6.9 met u vermenigvuldigen, krijgen we:

$$uR\left(1-\frac{1}{a^3}\right)+ur\left(1-\frac{1}{b^3}\right)=0 \qquad (1.6.12)$$

Als we vergelijking 1.6.11 van vergelijking 1.6.12 aftrekken krijgen we:

$$ur\left(1-\frac{1}{b^3}\right)+vr\left(1-\frac{1}{b^3}\right)=0$$

Aangezien $r \neq 0$: (anders zoude we slechts 2 lichamen hebben)

$$(u+v)\left(1-\frac{1}{b^3}\right)=0$$

Daar ook:

$$u+v\neq 0$$

Dan is:

$$1-\frac{1}{b^3}=0$$

dus:

$$b=1$$

Verder volgt hieruit ook dat :

$$a=1$$

Het punt P is dus de tophoek van een gelijkzijdigedriehoek, waarvan de basis de lijn $\overline{Mm}$ Is. Aangezien er twee van zulke driehoeken zijn, één boven en één onder deze lijn, zijn er ook twee punten $P_4$ en $P_5$ die aan de voorwaarde voldoen.

In totaal bestaan er dus vijf punten en niet meer, waar de drie krachten elkaar in evenwicht houden. Men noemt deze punten de *libratiepunten*.
Fig.1.6.5 geeft de vijf libratiepunten met de juiste afstanden voor het stelsel aarde-maan, waarvoor:

$$r = \frac{1}{82}$$

Alle libratiepunten draaien met het stelsel aarde-maan mede, wat betekent dat kleine stoffelijke massa's die in de punten $L_1$, $L_2$ en $L_3$ geplaatst worden, dus niet de bij hun afstand van de aarde passende circulaire snelheden hebben.

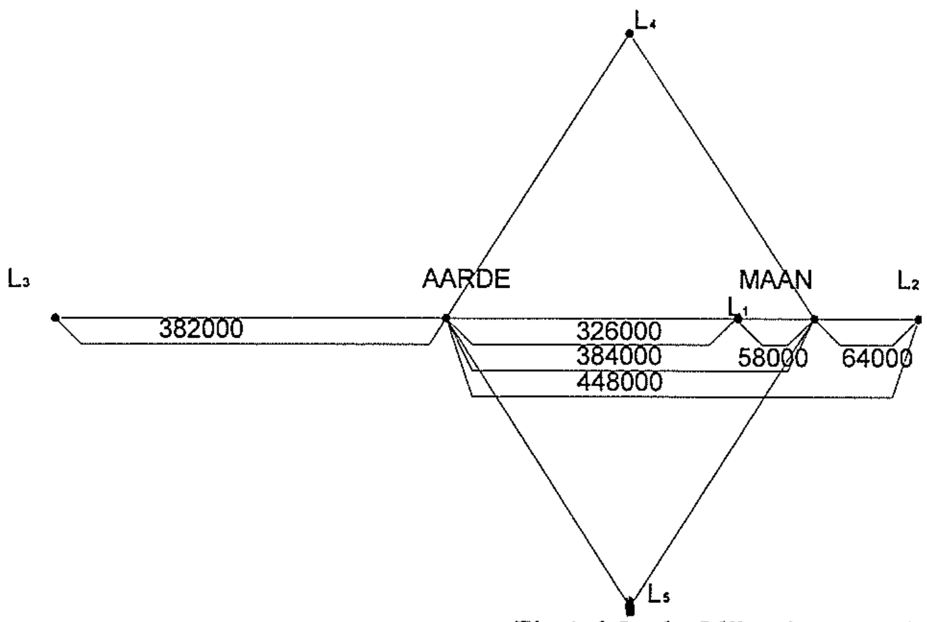



Fig.1.6.5 : de 5 libratiepunten in het stelsel aarde-maan

We merken op dat $L_1$ niet het neutrale punt is, de zon heeft nog een grote invloed op dat punt.

Het feit dat in de libratiepunten de aantrekkingskrachten en de centrifugale krachten elkaar opheffen, zegt nog niet dat een lichaam dat in één van die punten geplaatst wordt daar in stabiel evenwicht verkeert. Dit is echter wel zo in de punten $L_4$ en $L_5$ als er niet te grote krachten uit geoefend worden op een lichaam dat zich in één van die punten bevindt, dan zal dit lichaam een kleine ellips om het libratiepunt beschrijven, toch zal het steeds in de buurt van dat punt blijven.

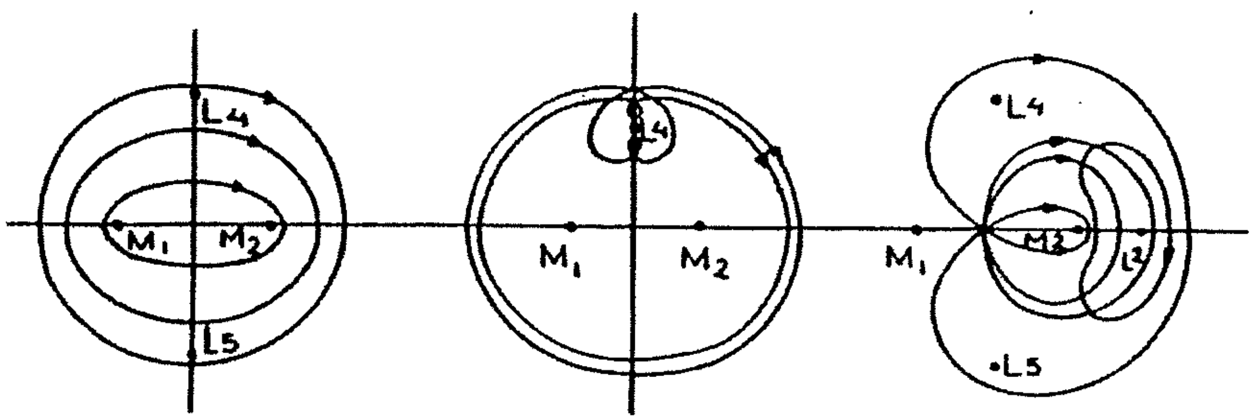



Fig.1.6.6 : relatieve banen bij een drie-lichamensysteem

# 2 Interplanetaire banen en plaatsbepaling in de ruimte

## 2.1 Het berekenen van de baan + voorbeeld

### 2.1.1 Het berekenen van de baan

Hoe kunnen we de elliptische baan van een object bepalen aan de hand van drie plaatsbepalingen? Dit zullen we zien in dit hoofdstuk.

De coördinaten van de 3 plaatsen zijn:

| Plaats | Afstand van de zon | Hoek |
|---|---|---|
| $R_1$ | $r_1$ | $\lambda_1$ |
| $R_2$ | $r_2$ | $\lambda_2$ |
| $R_3$ | $r_3$ | $\lambda_3$ |

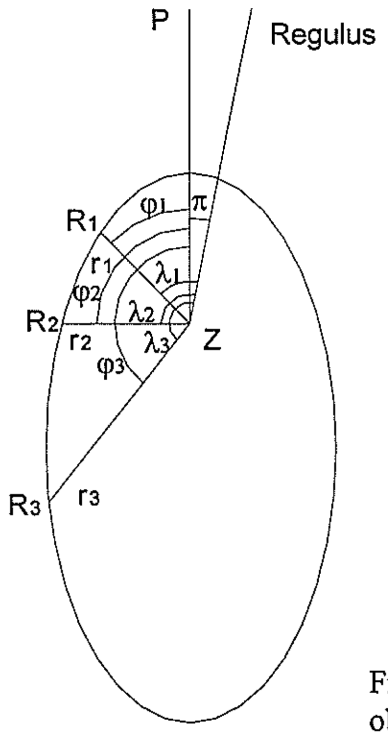



Fig. 2.1.1 : drie plaatsen van een object op verschillende tijdstippen

Op elk van de waarnemingen kunnen we de volgende vergelijking toepassen:

$$r = \frac{p}{1+\varepsilon\cos\varphi}$$

Voor het eerste punt $R_1$ geldt dus: $r_1 = \dfrac{p}{1+\varepsilon\cos\varphi_1}$ (2.1.1)

We veronderstellen nu dat de grote as van de ellips een hoek π maakt met de lijn Zon-Regulus. Daaruit volgt dan dat: $\varphi_1 = \lambda_1 - \pi$

Dus:

$$r_1 = \frac{p}{1+\varepsilon\cos(\lambda_1-\pi)} \qquad (2.1.2)$$

$$r_2 = \frac{p}{1+\varepsilon\cos(\lambda_2-\pi)} \qquad (2.1.3)$$

$$r_3 = \frac{p}{1+\varepsilon\cos(\lambda_3-\pi)} \qquad (2.1.4)$$

Uit deze vergelijkingen kunnen we gemakkelijk p, ε en π halen.
Uit (2.1.2) en (2.1.3) halen we:

$$p = r_1 + \varepsilon r_1\cos(\lambda_1-\pi) = r_2 + \varepsilon r_2\cos(\lambda_2-\pi)$$

Waaruit volgt:

$$\varepsilon = \frac{\mathrm{r}_2 - r_1}{r_1\cos(\lambda_1-\pi) - r_2\cos(\lambda_2-\pi)} \qquad (2.1.5)$$

Uit (2.1.3) en (2.1.4) halen we:

$$\mathrm{p} = r_3 + \varepsilon r_3\cos(\lambda_3-\pi) = r_2 + \varepsilon r_2\cos(\lambda_2-\pi)$$

Waaruit volgt:

$$\varepsilon = \frac{\mathrm{r}_3 - \mathrm{r}_2}{\mathrm{r}_2\cos(\lambda_2-\pi) - \mathrm{r}_3\cos(\lambda_3-\pi)} \qquad (2.1.6)$$

Uit (2.1.5) en (2.1.6) halen we dan:

$$\frac{\mathrm{r}_2 - \mathrm{r}_1}{\mathrm{r}_1\cos(\lambda_1-\pi) - \mathrm{r}_2\cos(\lambda_2-\pi)} = \frac{\mathrm{r}_3 - \mathrm{r}_2}{\mathrm{r}_2\cos(\lambda_2-\pi) - \mathrm{r}_3\cos(\lambda_3-\pi)}$$

Verder uitgewerkt geeft dit:

$$(r_3 - r_1)r_2\cos(\lambda_2-\pi) = (r_3 - r_2)r_1\cos(\lambda_1-\pi) + (r_2 - r_1)r_3\cos(\lambda_3-\pi) \qquad (2.1.7)$$

We weten dat:

$$\cos(\alpha-\beta)=\cos\alpha\cos\beta+\sin\alpha\sin\beta$$

We passen dit toe op vergelijking (2.1.7) en noemen voor de eenvoudigheid.

$$(r_3-r_2)r_1=A$$
$$(r_3-r_1)r_2=B$$
$$(r_2-r_1)r_3=C$$

We krijgen dan:

$$\mathrm{B}(\cos\lambda_2\cos\pi+\sin\lambda_2\sin\pi)=\mathrm{A}(\cos\lambda_1\cos\pi+\sin\lambda_1\sin\pi)+\mathrm{C}(\cos\lambda_3\cos\pi+\sin\lambda_3\sin\pi)$$

Als we nu alles delen door cos π dan krijgen we:

$$\mathrm{B}(\cos\lambda_2+\sin\lambda_2 tg\pi)=\mathrm{A}(\cos\lambda_1+\sin\lambda_1 tg\pi)+\mathrm{C}(\cos\lambda_3+\sin\lambda_3 tg\pi)$$

Hieruit halen we dan de waarde voor π:

$$\pi=tg^{-1}\left(-\frac{A\cos\lambda_1-B\cos\lambda_2+C\cos\lambda_3}{A\sin\lambda_1-B\sin\lambda_2+C\sin\lambda_3}\right)$$

Wanneer wij π kennen is uit vergelijking (2.1.5) of (2.1.6) de grootte van ε te berekenen en daarna door een kleine omvorming van vergelijking (2.1.1), (2.1.2), (2.1.3) of (2.1.4) de grootte van p.
Aangezien zoals bekend $p=a(1-\varepsilon^2)$ kunnen we hieruit de halve grote as (a) halen.

### 2.1.2 Voorbeeld baanberekening

We nemen aan dat het ruimteschip in een elliptisch vlak beweegt. We nemen als voorbeeld 10/10/2056 om 10 u. Aan boord van het ruimteschip worden er twee foto's gemaakt, de eerste van de Zon en de tweede van de aarde. Opmetingen van het resultaat geven het volgende:

- Schijnbare Regulus lengte van de zon $\lambda'_1$ 92°
- Schijnbare Regulus lengte van de aarde $\lambda'_2$ 136°

Schijnbare lengte wil zeggen: de lengte vanuit het ruimteschip gezien en Regulus lengte betekent dat de meridiaan door Regulus als nulpunt op de ecliptica wordt genomen.
In de Astronautische almanak van het jaar 2056 de 10$^{de}$ oktober om 10 u. vinden we voor de coördinaten van de aarde:
Regulus lengte $\lambda_2$ 226°.
De afstand van de zon tot de aarde is $r_2$ = 1,49 . 10$^8$ km.
Uit de volgende vergelijking vinden we nu dat de afstand van de Zon tot het ruimteschip:

$$r=r_2\frac{\sin\lambda'_2-\lambda_2}{\sin\lambda'_1-\lambda_2}=1{,}49.10^8\frac{\sin(136°-226°)}{\sin(92°-136°)}=2{,}15.10^8 km$$

Verder is:

$\lambda = 180° + \lambda''_1 = 180° + 92° = 272°$

Nu weten we al waar het ruimteschip zich bevindt in het zonnestelsel, omdat we ervan uitgingen dat de beweging zich in het vlak van de ecliptica bevond. Indien dit niet het geval is komt er een coördinaat bij nl. de breedtegraad van het ruimteschip.
Op de volgende data krijgen we de volgende gegevens:

| Datum: | Afstand van de zon r: | Regulus lengte: |
|---|---|---|
| 10/10/2056 | 2,15 . $10^8$ km | 272° |
| 29/11/2056 | 1,86 . $10^8$ km | 289° |
| 28/12/2056 | 1,47 . $10^8$ km | 303,5° |

Nu berekenen we eerst:

$$(r_3 - r_2)r_1 = A = (1{,}47 - 1{,}86)2{,}15 = -0{,}8385.10^{16}$$

$$(r_3 - r_1)r_2 = B = (1{,}47 - 2{,}15)1{,}86 = -1{,}2648.10^{16}$$

$$(r_2 - r_1)r_3 = C = (1{,}86 - 2{,}15)1{,}47 = -0{,}4263.10^{16}$$

Uit de vergelijking zie berekening baan: volgt dat,

$$tg\pi = \frac{-0{,}8385.\cos 272° + 1{,}2648.\cos 289° - 0{,}4263.\cos 203{,}5°}{-0{,}8385.\sin 272° + 1{,}2648.\sin 289° - 0{,}4263.\sin 203{,}5°}$$

$$= -\frac{0{,}029 + 0{,}412 - 0{,}235}{0{,}838 - 1{,}197 + 0{,}335} = 37$$

Waaruit volgt dat: $\pi = 88{,}5°$

Uit vergelijking 2.1.5 volgt dat:

$$\varepsilon = \frac{r_2 - r_1}{r_1 \cos(\lambda_1 - \pi) - r_2 \cos(\lambda_2 - \pi)} = \frac{1{,}86 - 2{,}15}{2{,}15 \cos 183{,}5° - 1{,}86 \cos 200{,}5°} = 0{,}71°$$

Volgens vergelijking 2.1.2 is:

$$p = r[1 + \varepsilon \cos(\lambda - \pi)]$$

$$= 2{,}15.10^8 (1 - 0{,}71 \cos 183{,}5°)$$

$$= 0{,}63.10^8$$

Waaruit volgt dat:

$$a = \frac{p}{1 - \varepsilon^2} = \frac{0{,}63.10^8}{1 - 0{,}71^2} = 1{,}27.10^8 km$$

### 2.1.3 De nauwkeurigheid van de resultaten

Het is duidelijk dat onze resultaten fouten zullen vertonen. In het volgende stuk tekst en redenering zullen wij u dat aantonen.
Wanneer de fotografische platen die 10° in het vierkant van de hemel beslaan 30 x 30 cm groot zijn en we nemen aan dat de nauwkeurigheid van onze opmeting 1/20 mm is, dan betekend dit een nauwkeurigheid in de plaatsbepaling van de vaste sterren van 6 boog –

seconden komt overeen met 3/100.000 radialen op een afstand van de zon tot de aarde betekend dit een fout in de positie van 3. $10^{-5}$ . 1,5. $10^{8}$ km = 4500 km.
We geven dit getal enkel weer om de orde van de grootte te bepalen. Het zou zeer ingewikkelde berekeningen vergen om uit de opgegeven fout in de opmeting van de fotografische plaat de fout in de plaatsbepaling van het ruimteschip te berekenen.
We kunnen aannemen dat de maximale fout zeker driemaal zo groot is als daarnet berekend. Dan zou dit ongeveer 13500 km zijn. Nu zou je denken dat dit een grote fout is maar in het zonnestelsel is dit nog maar een kleine fout. Pas als we grote reizen, van honderd duizenden kilometers, zouden maken; zou de fout ontoelaatbaar worden waardoor we andere methoden zouden moeten zoeken. En ook op zo'n afstand mogen we niet meer aannemen dat ons ruimteschip nog een correcte Kepler baan zal volgen. Dit omdat de aantrekking van de planeet in het spel komt en dan krijgen we te maken met een drie lichaam probleem.
In de nabijheid van de planeten zijn radiografische methoden zoals radar of radiopeiling de aangewezen methoden.

## 2.2 Mogelijke banen + baancorrectie

Bij het behandelen van de mogelijke banen die kunnen optreden zullen we zien dat er drie soorten zijn, nl.: de directe, de indirecte en de Hohmannbanen.
Bij deze bespreking zullen we ook moeten aannemen dat:
1. Het grootste gedeelte van de reis is in vrije vlucht, zodanig dat we de periode van versnelling van de motor kunnen verwaarlozen.
2. Er wordt enkel rekening gehouden met de aantrekkingskracht van de zon waardoor we de baan van het ruimteschip als een zuivere Keplerbaan mogen beschouwen.
3. De aarde is normaal de planeet van vertrek.

### 2.2.1 De Directe baan

Bij de berekening van de baan naar een binnenplaneet moet de periheliumstraal van de ellips kleiner of hoogstens gelijk zijn aan de straal van de baan van de planeet. De apheliumstraal moet groter of minstens gelijk zijn aan de straal van de aardbaan.

$r_0 \leq r_2$ en $r_{180} \geq r_1$

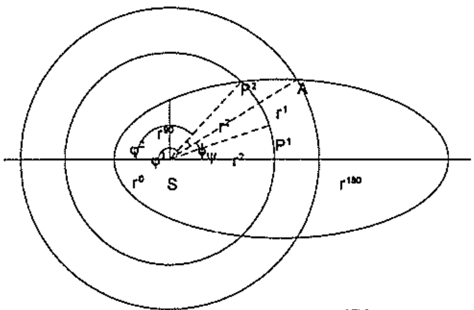



Fig. 2.2.1: een binnenplaneet

Eerder hebben we bewezen dat de kegelsneden voldoen aan vergelijking 1.1.1:

$$r = \frac{P}{1 + \varepsilon \cos \varphi}$$

Als we nu aannemen bij de ellips dat:

$$P = a(1-\varepsilon^2) \quad \text{en} \quad 0 < \varepsilon < 1$$

Uit deze vergelijkingen kunnen we nu besluiten dat:

1. $r_0 = \dfrac{a(1-\varepsilon^2)}{1+\varepsilon\cos\varphi}$ met $\varphi = 0°$,

   $r_0 = a(1-\varepsilon)$

2. $r_{180} = \dfrac{a(1-\varepsilon^2)}{1+\varepsilon\cos\varphi}$ met $\varphi = 180°$,

   $r_{180} = a(1+\varepsilon)$

Als we deze twee waarden nu invullen, vinden we:

$$a(1-\varepsilon) \leq r_2 \quad \text{en} \quad a(1+\varepsilon) \geq r_1$$

Na het vermenigvuldigen van de eerste vergelijking met ( 1 + ε ) en de tweede vergelijking met ( 1 - ε ), vinden we:

$$a(1-\varepsilon^2) \leq r_2(1+\varepsilon) \quad \text{en} \quad a(1-\varepsilon^2) \geq r_1(1-\varepsilon)$$

Wanneer we dit alles nu delen door $r_1$, m.a.w. dit uitdrukken in astronomische eenheden, bekomt men:

$$\frac{a(1-\varepsilon^2)}{r_1} \leq \frac{r_2(1+\varepsilon)}{r_1} \quad \text{en} \quad \frac{a(1-\varepsilon^2)}{r_1} \geq \frac{r_1(1-\varepsilon)}{r_1}$$

Om de vergelijking te vereenvoudigen, noemen we:

$$\frac{r_2}{r_1} = n \quad \text{en} \quad P = \frac{a(1-\varepsilon^2)}{r_1} \text{, zodat: } P \leq n(1+\varepsilon) \quad \text{en} \quad P \geq (1-\varepsilon) \qquad (2.2.1)$$

Bij het zoeken van een vergelijking voor een hyperbolische baan, zien we niet veel verschillen met het voorgaande. Hierbij is de periheliumafstand kleiner of hoogstens gelijk aan de straal van de baan van de planeet.

$$r_0 \leq r_2$$

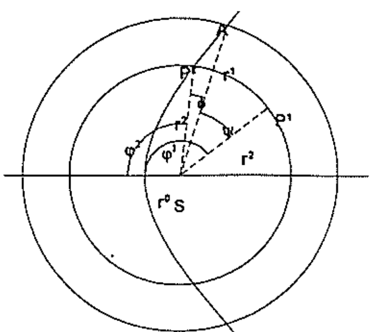



Fig. 2.2.2: hyperbolische baan

Met $P = a(\varepsilon^2 - 1)$ en $\varepsilon > 1$, krijgen we:

$$r_0 = \frac{a(\varepsilon^2 - 1)}{1 + \varepsilon \cos \varphi} \quad \text{met } \varphi = 0°$$

$$r_0 = a(\varepsilon - 1)$$

Na het invullen van deze vergelijking vinden we:

$$a(\varepsilon - 1) \leq r_2$$

Wanneer we deze vergelijking vermenigvuldigen met ( ε + 1 ), krijg je:

$$a(\varepsilon^2 - 1) \leq r_2(\varepsilon + 1)$$

Als we deze vergelijking nu ook omzetten in astronomische eenheden, dan wordt dit:

$$P = n(\varepsilon + 1) \tag{2.2.2}$$

De manier om de vergelijking voor een buitenplaneet te vinden is gelijkaardig aan de binnenplaneten. Hierbij is de periheliumstraal van de ellips kleiner of hoogstens gelijk aan de straal van de aardbaan. De apheliumstraal van de ellips moet groter of minstens gelijk aan de straal van de baan van de planeet.

$$r_0 \leq r_1 \quad \text{en} \quad r_{180} \geq r_2$$

Als we de vergelijking van de kegelsneden toepassen, dan kunnen we schrijven dat:

1. voor de parabool:

$$a(1 - \varepsilon) \leq r_1 \quad \text{en} \quad a(1 + \varepsilon) \geq r_2$$

Bij het vermenigvuldigen van het eerste lid met ( 1 + ε ) en het tweede lid met ( 1 - ε ), dan krijg je:

$$a(1 + \varepsilon^2) \leq r_1(1 + \varepsilon) \quad \text{en} \quad a(1 - \varepsilon^2) \geq r_2(1 - \varepsilon)$$

Wanneer we dit terug omzetten in astronomische eenheden, dan bekom je:

$$P \leq (1 + \varepsilon) \quad \text{en} \quad P \geq n(1 - \varepsilon) \tag{2.2.3}$$

2. voor de hyperbolische baan:

$$a(1 - \varepsilon) \leq r_1$$

Bij het vermenigvuldigen van deze vergelijking met ( 1 + ε ), dan krijg je:

$$a(1 - \varepsilon^2) \leq r_1(1 + \varepsilon)$$

Wanneer we dit omzetten in astronomische eenheden, dan bekom je:

$$P \leq (1+\varepsilon) \qquad (2.2.4)$$

Als we nu van de binnen- en buitenplaneten dit alles in een grafiek zetten, kunnen we bemerken dat de kegelsneden elkaar zullen snijden in een gemeenschappelijk punt. Voor de beide gevallen is de parameter van dit punt gelijk aan:

$$P = \frac{2n}{n+1} \qquad (2.2.5)$$

En is de excentriciteit gelijk aan:
voor de binnenplaneten:

$$\varepsilon = \frac{n-1}{n+1} \qquad (2.2.6)$$

voor de buitenplaneten:

$$\varepsilon = -\frac{n-1}{n+1} \qquad (2.2.7)$$

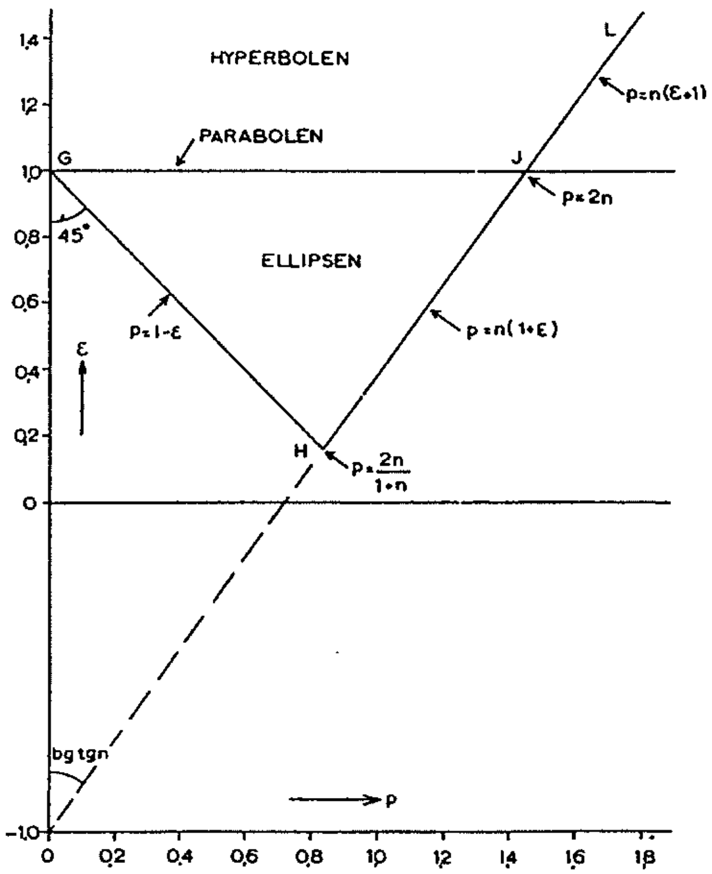



Fig. 2.2.3: de binnenplaneten

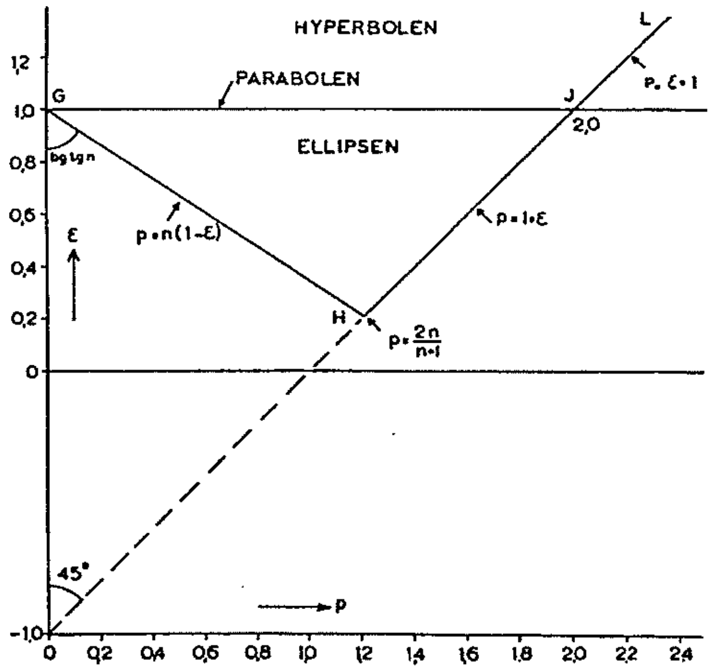



Fig. 2.2.4: de buitenplaneten

## 2.2.2 De Hohmannbaan

De Hohmannbaan heeft de vorm van een halve ellips die de banen van de twee planeten zal raken. De Hohmannbaan wordt gekoppeld aan enkele voordelen en nadelen:

Voordelen:

- Men heeft genoeg met een kleine snelheid.
- Het is het meest economisch op basis van brandstofgebruik.

Nadelen:

- Het is de langste reistijd.
- Moeilijke navigatie.
- Het is niet de kortste baan.

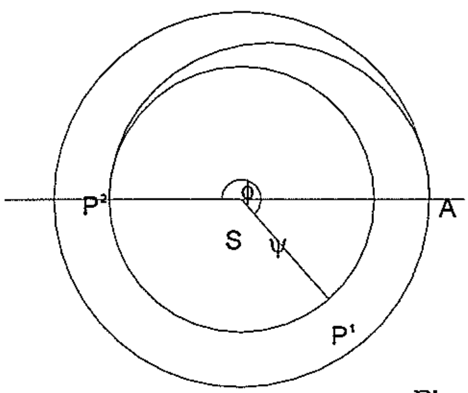



Fig. 2.2.5: de Hohmannbaan

Hierbij zal: $r_0 = r_2$ en $r_1 = r_{180}$

Op deze vergelijkingen zullen we dezelfde bewerkingen uitvoeren als bij de directe banen en bekomen we de volgende uitdrukkingen:

Voor binnenplaneten:

$$P = (1 + \varepsilon) \quad \text{en} \quad P = n(1 - \varepsilon)$$

Hieruit volgt dat:

$$1 + \varepsilon = n(1 - \varepsilon) \tag{2.2.8}$$

Uit deze vergelijkingen kunnen we opnieuw het volgende bepalen:

$$P = \frac{2n}{n+1} \quad \text{en} \quad \varepsilon = -\frac{n-1}{n+1} \tag{2.2.9}$$

Voor de buitenplaneten zal enkel de excentriciteit van teken veranderen:

$$\varepsilon = \frac{n-1}{n+1} \tag{2.2.10}$$

### 2.2.3 De Indirecte banen

Als je de onderstaande figuur bekijkt, zul je zien dat de ellips in vier punten gesneden. Deze vier punten zullen de vier mogelijke banen opleveren. Zoals men ziet zal de directe baan de kortste baan zijn en afhankelijk van de andere punten waarin de ellips gesneden wordt, zullen er indirecte banen optreden.

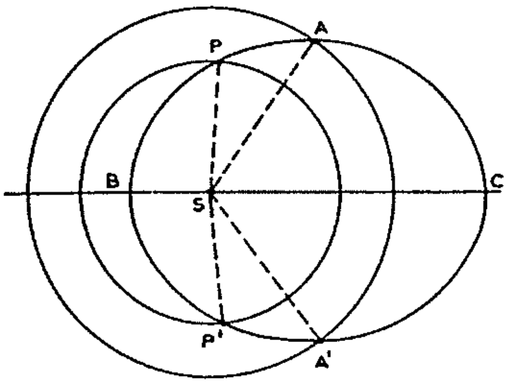



Fig. 2.2.6: de indirecte banen.

Mogelijke banen:
1. Route AP = de directe baan.
2. Route ABP' = de periheliumbaan.
3. Route A'CP = de apheliumbaan.
4. Route A'CBP' = de omwegbaan.

Bij deze vier banen kan men bemerken dat de tijdsduur verschillend zal zijn, maar de hoeveelheid energie die nodig zal zijn is gelijk.

## 2.2.4 Baancorrectie

Als men een reis naar een planeet zal maken, zal men eerst de baan die het ruimteschip zal volgen nauwkeurig berekenen. Deze baanberekening is ingewikkelder dan al het voorgaande, want men moet alle storende factoren in rekening brengen, zoals:

- De aantrekkingskracht van de planeten, want deze kunnen op miljoenen kilometers afstand nog altijd een invloed uitoefenen.
- De planetenbanen lopen niet in het vlak van de ecliptica en ze zijn ook geen cirkels.

Ook het feit dat we vertrekken vanaf de aarde of een ruimtestation zal de berekening niet makkelijker maken.
Hoe nauwkeurig men de baan ook berekent, er zal altijd een afwijking van de baan optreden tijdens de lancering. Tijdens de vlucht zal de piloot dus moeten kunnen ingrijpen om een correctie van de baan aan te brengen.
Daarom zullen wij nu de correctie van de banen bespreken.

In de onderstaande figuur zien we de oorspronkelijke bedoelde baan, ellips 0, afgebeeld:

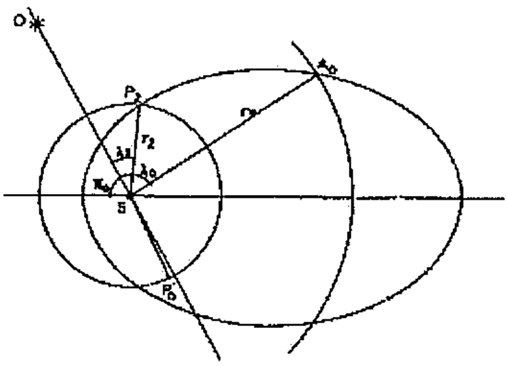


Fig. 2.2.7: ellips 0

Op een zeker tijdstip $t_1$ zal blijken dat op basis van drie waarnemingen de elementen van de elliptische baan die het ruimteschip in werkelijkheid volgt, zijn:

- excentriciteit $\varepsilon_1$
- halve grote as $a_1$
- periheliumafstand $\pi_1$

in plaats van:

- excentriciteit $\varepsilon_0$
- halve grote as $a_0$
- periheliumafstand $\pi_0$

Hierdoor moet de piloot de baan wijzigen zodat het ruimteschip de planeet toch op het oorspronkelijk bedoelde punt P2 ontmoet. Dit betekent dat de totale duur van de reis onveranderd blijft. De gecorrigeerde baan, ellips 2, wordt dan:

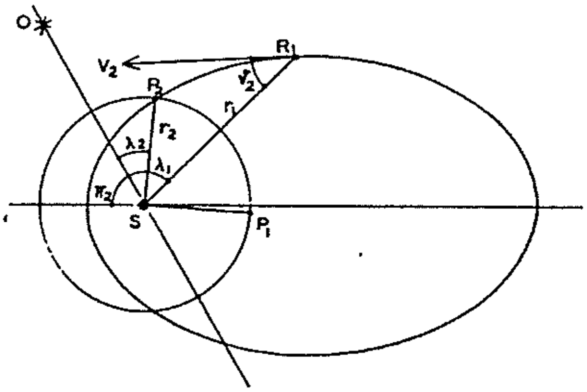



Fig. 2.2.8: ellips 2

Om verder te werken zullen we eerst het volgende moeten aannemen:
1. $T_{002}$ is de oorspronkelijk berekende duur van de reis.
2. $T_{X01}$ is de geregistreerde duur van de reis tot het punt $R_1$, dus de tijd $t_1$-$t_0$.
3. $T_{212}$ is de tijd die het ruimteschip nodig heeft om langs de gecorrigeerde ellips van het punt $R_1$ naar het punt $P_2$ te komen.

Hieruit volgt dat:

$$T_{X01} + T_{212} = T_{002}$$

Volgens de tijd die een lichaam nodig heeft om een gedeelte van een elliptische baan te doorlopen, is:

$$T_{212} = \frac{q_2^{3/2}}{2\pi}\{(\tau_1 - \tau_2) - \varepsilon_2(\sin\tau_1 - \sin\tau_2)\} \tag{2.2.11}$$

met:

$$\tau_1 = Bg\cos\frac{1-\frac{1}{q_2}}{\varepsilon_2} \qquad \text{en} \qquad \tau_2 = Bg\cos\frac{1-\frac{n}{q_2}}{\varepsilon_2}$$

Hierbij is $q_2$ de halve grote as in astronomische eenheden.

Als we nu verder gaan met de vergelijkingen van de gecorrigeerde ellips 2:

$$r_1 = \frac{a_2(1-\varepsilon_2^{\,2})}{1+\varepsilon_2\cos\varphi_1} \qquad \text{met: } \varphi_1 = \pi_2 + \lambda_1$$

$$a_2 = q_2 r_0$$

$$r_1 = n_1 r_0$$

Volgt eruit dat:

$$n_1 = \frac{q_2(1-\varepsilon_2^{\;2})}{1+\varepsilon_2\cos(\lambda_1+\pi_2)} \tag{2.2.12}$$

Een derde vergelijking die we kunnen opstellen is:

$$r_2 = \frac{a_2(1-\varepsilon_2^{\;2})}{1+\varepsilon_2\cos\varphi_2} \qquad \text{met:} \quad \varphi_2 = \lambda_2 + \pi_2$$

$$r_2 = n_2 r_0$$

$$a_2 = q_2 r_0$$

Waaruit volgt dat:

$$n_2 = \frac{q_2(1-\varepsilon_2^{\;2})}{1+\varepsilon_2\cos(\lambda_2+\pi_2)} \tag{2.2.13}$$

Met de drie vergelijkingen die we zojuist bepaald hebben kunnen we de onbekenden $q_2$, $\varepsilon_2$ en $\pi_2$ berekenen, zodat de gecorrigeerde ellips volledig bepaald is.
Deze berekening is echter niet eenvoudig en we ze zullen ons moeten baseren op het feit dat de onbekenden van de gecorrigeerde ellips weinig zullen afwijken van de vooropgestelde ellips 1. Wij kunnen dan stellen dat:

$$q_2 = q_1 + \Delta q$$
$$\varepsilon_2 = \varepsilon_1 + \Delta\varepsilon$$
$$\pi_2 = \varepsilon_1 + \Delta\pi$$

waarin $\Delta q$, $\Delta\varepsilon$ en $\Delta\pi$ kleine grootheden zijn.

Volgens de reeks van Mac-Laurin kunnen we stellen:

$$f(x) - f(0) = f'(0)x + \tfrac{1}{2} f''(0)x^2 + \ldots$$

Wanneer $\Delta x$ klein is, kunnen we de machten van $\Delta x$ verwaarlozen, zodat:

$$f(\Delta x) - f(0) \approx f'(0)\Delta x$$

Voor de twee veranderlijken krijgen we dan:

$$f(\Delta x, \Delta y) - f(0,0) \approx \left(\frac{\partial f}{\partial x}\right)_{00} \Delta x + \left(\frac{\partial f}{\partial y}\right)_{00} \Delta y$$

Als we de vergelijkingen op deze manier berekenen, kunnen we de onbekenden gemakkelijker zoeken. Een voorwaarde is echter dat de correctie van de baan klein moet zijn t.o.v. de oorspronkelijke grootheden. Nadat we de correcties op de elliptische baan berekend hebben, kunnen we de correctie op de snelheid berekenen:

Algemeen kunnen we aannemen dat:

$$tg\vartheta_1 = -\frac{1+\varepsilon_1 \cos\varphi_1}{\varepsilon_1 \sin\varphi_1} \qquad (2.2.14)$$

met: $\varphi_1 = \pi_1 + \lambda_1$

zodat:

$$\vartheta_1 = Bgtg - \frac{1+\varepsilon_1 \cos(\pi_1 + \lambda_1)}{\varepsilon_1 \sin(\pi_1 + \lambda_1)}$$

Na de correctie moet de snelheid tangentiaal zijn aan de ellips 2 in het punt R1, zodat:

$$\vartheta_2 = Bgtg - \frac{1+\varepsilon_2 \cos(\pi_2 + \lambda_1)}{\varepsilon_2 \sin(\pi_2 + \lambda_1)}$$

Aangezien we de waarden $\varepsilon_2$ en $\pi_2$ kennen, kunnen we de hoek berekenen.
De snelheid in het punt R1 is:

$$V_1 = \sqrt{\mu\left(\frac{2}{a_1} - \frac{1}{r_1}\right)} = \sqrt{\frac{\mu}{r_0}}\sqrt{\frac{2}{q_1} - \frac{1}{n_1}} = V_a\sqrt{\frac{2}{q_1} - \frac{1}{n_1}}$$

Waarin $V_a$ de snelheid is van de planeet in een circulaire baan met straal $r_0$. De gecorrigeerde snelheid $V_2$ in de gecorrigeerde ellips 2 moet dan zijn:

$$V_2 = \sqrt{\mu\left(\frac{2}{a_2} - \frac{1}{r_1}\right)} = \sqrt{\frac{\mu}{r_0}}\sqrt{\frac{2}{q_2} - \frac{1}{n_1}} = V_a\sqrt{\frac{2}{q_2} - \frac{1}{n_1}}$$

Daar we de waarde $q_2$ kennen, kunnen we de snelheid $V_2$ berekenen. We zullen een extra snelheid V toevoegen om de grootte en de richting van de snelheid $V_1$ te corrigeren.
We noemen verder:

$$\vartheta_1 - \vartheta_2 = \Delta\vartheta$$

Zodat we volgens de cosinus – regel kunnen stellen dat:

$$V^2 = V_2^{\,2} + V_1^{\,2} - 2V_1V_2 \cos\Delta\vartheta$$

Verder noemen we ook:

$$V_2 = V_1 + \Delta V$$

Waaruit volgt dat:

$$\begin{aligned} V^2 &= (V_1 + \Delta V)^2 + V_1^{\,2} - 2V_1(V_1 + \Delta V)\cos\Delta\vartheta \\ &= V_1^{\,2} + \Delta^2 V + 2V_1\Delta V + V_1^{\,2} - 2V_1^{\,2}\cos\Delta\vartheta - 2V_1\Delta V\cos\Delta\vartheta \end{aligned}$$

$$= 2V_1(V_1 + \Delta V)(1 - \cos\Delta\vartheta) + \Delta^2 V$$

Aangezien Δθ een kleine grootheid is, kunnen we schrijven dat:
$1 - \cos\Delta\vartheta \approx \frac{1}{2}(\Delta\vartheta)^2$ , zodat:

$$V^2 = V_1 V_2 (\Delta\vartheta)^2 + \Delta^2 V$$

Waaruit wij de waarde van de snelheid V kunnen halen:

$$V = \Delta\vartheta\sqrt{V_1 V_2 + \left(\frac{\Delta V}{\Delta\vartheta}\right)^2}$$

Verder krijgen we volgens de sinusregel dat:

$$\sin\alpha = \frac{\sin\Delta\vartheta}{V} V_2$$

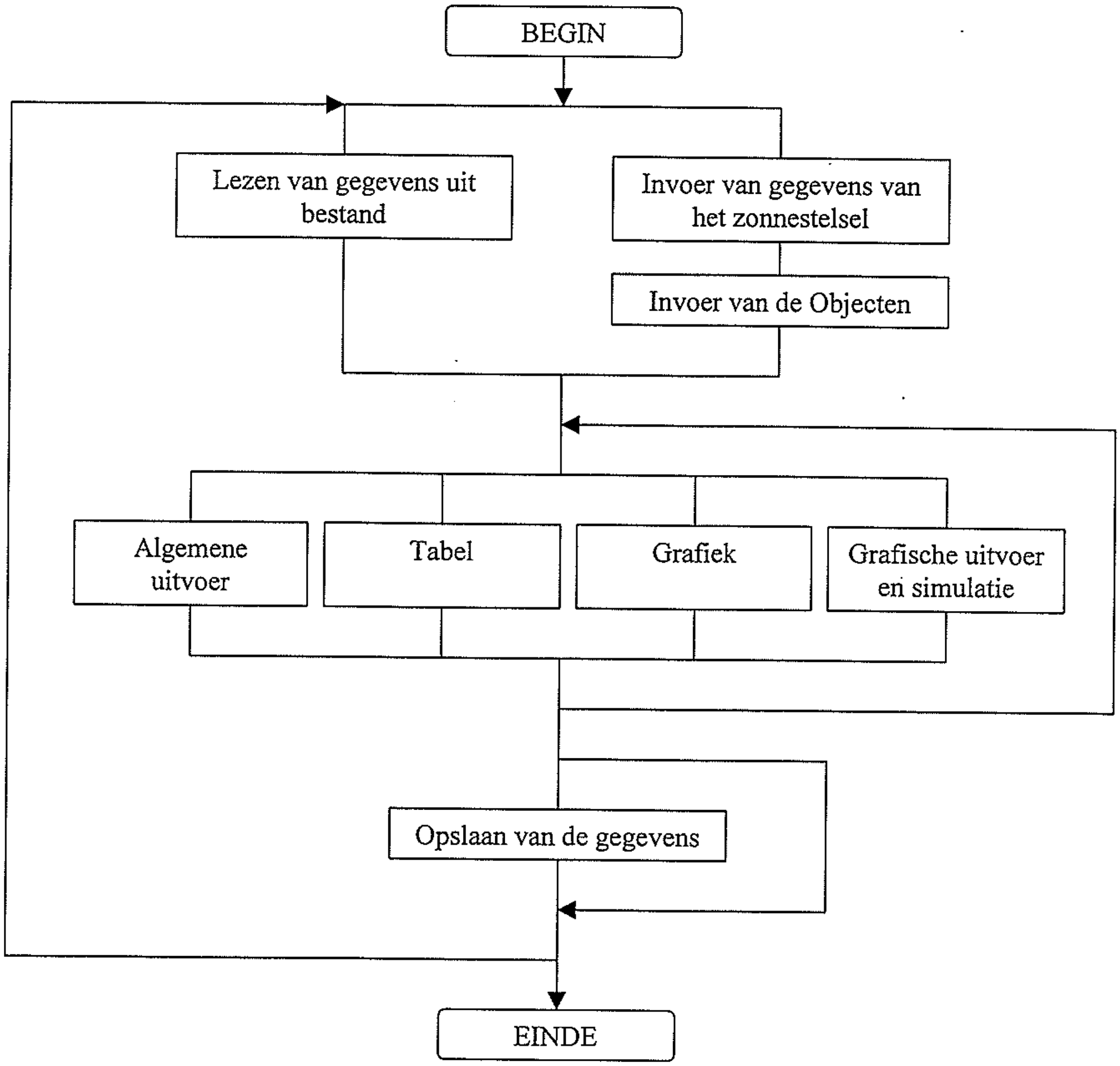
Daar Δθ een kleine hoek is, kunnen we stellen dat:
$\sin\Delta\vartheta \approx \Delta\vartheta$, zodat:

$$\sin\alpha = \frac{\Delta\vartheta}{V} V_2 \qquad (2.2.15)$$

# 3 Programma

## 3.1 Algemene omschrijving

Het programma kan via invoer van parameters van de planeten en een object dat een interplanetaire baan beschrijft de toestand van de planeten en het object berekenen. De parameters kunnen manueel worden ingevoerd of worden opgelezen uit een bestand dat gemaakt is met dit programma en na de manuele invoer kunnen de parameters eveneens worden opgeslagen.




Het programma omvat 4 mogelijkheden om uitvoer van resultaten te bekomen. Ten eerste is er de algemene uitvoer waarmee bijvoorbeeld de tijd waarbij het object onderweg is, de toestand op een bepaald moment, ... kan worden berekend. Het biedt tevens de mogelijkheid een tabel of een grafiek op te stellen met de toestand op verschillende momenten. Tenslotte kan men er een visuele voorstelling worden gemaakt van de banen die de planeten en de objecten omschrijven. Er is ook een mogelijkheid tot een simulatie voorzien.

## 3.2 Programmeertaal Visual Basic

### 3.2.1 Modules

*a) Module*

Een verzameling van programmacodes die onder andere declaraties en procedures bevat om bepaalde opdrachten te doen uitvoeren. Een module bestaat uit een declaratie gedeelte waarin de types, variabelen en constanten kunnen worden gedeclareerd. Daarna volgen de procedures waarin de opdrachten worden geformuleerd.

*b) Klasse*

Een klassemodule is de typedeclaratie van een structuur vergelijkbaar met een record (zie 3.2.4) met meerdere variabelen (eigenschappen) die ook extra functies en procedures (methodes) bevat die de klasse kunnen manipuleren. Een normale klasse moet manueel worden gecreëerd met het keyword New in de declaratie of toewijzing.

*c) Form*

Een Form is een speciale op zich zelf bestaande klasse, die een Windows-venster voorstelt, met de informatie die het venster kan produceren. Een form omvat een module waarin onder andere de code voor de events van de besturingselementen op die form worden uitgewerkt.

### 3.2.2 Procedures

{*Module.*}procedure
Een reeks van opeen acties die moet worden uitgevoerd in een programma. Een procedure kan in een andere procedure worden opgeroepen en uitgevoerd door de naam op te geven. Een procedure kan parameters bevatten, die worden opgegeven na de naam, die alleen maar bestaan in die procedure.

*a) Sub*

{Private of Public} Sub *naam* ({*parameter* As *gegevenstype*, …})
Een standaardprocedure die gewoon reeks acties kan uivoeren.

*b) Function*

{Private of Public} Function *naam* ({*parameter* As *gegevenstype*, …}) {As *gegevenstype*}
Een procedure die een reeks acties uitvoert en een waarde van een bepaald type kan afleveren na de uivoering van de functie.

### 3.2.3 Objecten

Een object is eigenlijk een andere naam voor een klasse maar wordt bij voorkeur niet gebruikt voor normale klassen maar wel voor speciale klassen zoals forms en besturingselementen op de forms. Een object heeft bepaalde eigenschappen en methoden. Het heeft tevens gebeurtenisprocedures die automatisch worden aangeroepen als de gebruiker een bepaalde handeling uitvoert.

***Besturingselementen***

{*Module.*}object
Speciale klasse die een element op een form voorstelt om de gebruiker de gegevens te laten invoeren of om de resultaten aan de gebruiker terug te geven. Een besturingselement kan op het scherm worden zichtbaar gemaakt. Door eigenschappen en methodes te veranderen kan het uiterlijk ervan worden gemanipuleerd.

*a) Commandbutton*

Opdrachtknop om een bepaalde actie te doen uitvoeren waneer men erop klikt.

*b) Label*

Besturingselement om tekst op het scherm te brengen.

*c) Textbox*

Besturingselement om bewerkbare tekst in te voeren of op het scherm te brengen in een kader.

*d) Listbox*

Besturingselement om een lijst met tekst op het scherm te brengen.

*e) Combobox*

Besturingselement om een keuzelijst het scherm te brengen en een keuze te maken tussen de verschillende mogelijkheden.

*f) Picturebox*

Object met een afbeelding of tekst die kan worden bewerkt door bepaalde methodes.

*g) Multipage*

Element dat een Form indeelt in meerdere tabbladen.

*h) Optionbutton*

Keuzerondje waarmee je kunt kiezen tussen meerdere opties.

*i) Checkbox*

Keuzevakje om een item te selecteren.

*j) Form*

Een form als object.

### *Eigenschappen*

*Object*.eigenschap = *waarde*
Karakteristiek van een klasse of object waarmee het uiterlijk kan worden veranderd door er een waarde aan toe te kennen.

*a) Caption*

Beschrijvende tekst op het object om het te kunnen herkennen of te beschrijven.

*b) Value*

Status of de inhoud van een object (tekst of booleanwaarde)

*c) Text*

Tekst in een Textbox.

*d) List*

Gegevens in de lijst van een Listbox of een Combobox.

*e) Listindex*

Nummer van het huidig geselecteerde item in een Listbox of Combobox

### *Methoden*

*Object*.methode
Een procedure die een bepaalde handeling uitvoert met een klasse of object.

*a) Clear*

Alle gegevens van een object verwijderen.

*b) Additem*

Een gegeven aan een nieuwe rij van een Listbox of Combobox toekennen.

*c) Line*

*Object*.Line (*beginx, beginy*)-(*eindex, eindey*), *kleur*
Een lijn teken op een picturebox of een form van een beginpunt naar een eindpunt.

*d) Circle*

*Object*.Circle (*x*, *y*), *straal*, *vulkleur*
Tekent een cirkel met een bepaalde straal en die gevuld is met een bepaald kleur.

*e) Show*

Een form op het scherm brengen.

### *Gebeurtenissen*

*{Private} Sub Object*_gebeurtennis ( )

Een procedure die wordt uitgevoerd als de gebruiker een bepaalde handeling doet met een object.

*a) Click*

Als de gebruiker het besturingselement uitvoert door er op te klikken of te enteren.

*b) Change*

Als de gebruiker de waarde van een besturingselement verandert.

*c) Load*

Als de gebruiker een form in het geheugen laadt door bijvoorbeeld het op het scherm te brengen met de methode Show.

*d) UnLoad*

Als de gebruiker een form terug uit het geheugen laadt bijvoorbeeld met de instructie UnLoad.

*e) Activate*

Als de form op het scherm wordt gebracht en geactiveerd wordt deze event-procedure aangeroepen.

### 3.2.4 Variabelen

Een variabele is een geheugenplaats waar bepaalde bewerkbare gegevens kunnen worden opgeslagen. Het bevat bepaalde informatie die de gebruiker nodig heeft om het programma aan te sturen. Elke variabele moet worden gedeclareerd in het programma, heeft een naam en kan een bepaalde soort informatie bevatten BV: een tekst of een getal afhankelijk van het gegevenstype.

***Gegevenstypes***

De soorten informatie die een variabele kan bevatten.

*a) Variant*

Variabele van gelijk welk type, een variabele waaraan geen type is aan toegewezen is van het type variant.

*b) String*

Variabele die een tekst bevat.

*c) Boolean*

Variabele die een booliaanse waarde heeft die True of False kan zijn.

*d) Integer*

Variabele die een natuurlijk getal bevat.

*e) Single*

Variabele die een kort reëel getal bevat.

*f) Double*

Variabele die een lang reëel getal bevat.

*g) Date*

Variabele die een datum en tijdstip bevat.

*h) Array*

*Naam ({rij, kolom, ...})*
Variabele van een bepaald type dat meerdere posities heeft voor waarden van dat type in op te slaan.
Het aantal posities in een array kan worden bepaald met de functie UBound.
Een dynamische array is een array die bij zijn declaratie geen grootte krijgt toegewezen. De grootte van een dynamische array kan worden vastgelegd met het statement ReDim of ReDim Preserve. Met ReDim wordt de inhoud gewist met ReDim Preserve niet.
ReDim {Preserve} *array(rijen{, kolommen, ...})*

*i) Record*

Type *typenaam ...onderdeel1* {As *gegevenstype*} ...End Type
Dit maakt een type voor een variabele dat meerdere onderdelen heeft van verschillende types. Elk onderdeel kan worden aangeroepen door de variabele op te geven gevolgd door een punt en het onderdeel.
*Variabele.onderdeel*

### *Declaratie*

Public, Private of Dim *Variabele* {As *gegevenstype*}
Variabelen worden bovenaan in een module of in een procedure (alleen met Dim) gedeclareerd met een declaratieregel waarin het bereik, de naam van de variabele en het gegevenstype worden opgegeven.

*a) Public*

Variabele die kan worden gebruikt in alle modules.

*b) Private*

Variabele die slechts kan worden gebruikt in de module waarin hij is gedeclareerd.

*c) Dim*

Variabele die alleen maar kan worden gebruikt in de procedure of module waarin hij is gedeclareerd.

### *Toewijzing*

{Let} Variabele = waarde
De inhoud van een variabele wordt opgegeven door een waarde, berekening of eigenschap van een object eraan gelijk te stellen met een gelijkteken.

### 3.2.5 Constanten

Een constante is een bepaald gegeven waarvan de waarde niet kan veranderen.

***Declaratie***

Const *Constante* {As *gegevenstype*} = *waarde*
Een constante wordt boven aan een module gedeclareerd waarbij de naam en de waarde worden opgegeven. Deze waarde kan niet meer worden veranderd tijdens de uivoering van het programma.

### 3.2.6 Instructies

***Selectie***

*a) Keuze*

If *expressie* Then *acties* ...{Else *acties*} ...{End If}
Instructie die een keuze maakt tussen 1 of 2 mogelijkheden. Als de expressie, die van het type boolean is, true is worden de acties na Then uitgevoerd anders worden de acties na Else uitgevoerd.

*b) Meervoudige keuze*

Select Case *expressie* ...Case *Waarde1 acties1* ...Case *waarde2 acties2* ... End Select
Instructie die een keuze maakt tussen meerdere mogelijkheden. De acties na de waarde die gelijk is aan de expressie worden uitgevoerd.

***Iteraties***

*a) Herhaling*

Do {While of Until *Expressie*} *acties* Loop {While of Until *Expressie*}
Instructie die een aantal keer acties uitvoert tot de expressie van het type boolean bij While false is of bij Until true is.
*b) Herhaling met teller*

For *expressie* = *beginwaarde* To *eindwaarde acties* next expressie
Instructie die een aantal keer acties uitvoert tot de expressie vanaf de beginwaarde de eindwaarde heeft bereikt

***Bestanden bewerken***

*a) Open*

Open *bestandsnaam* For Input of Output As *#nummer*
Bestand openen om gegevens te lezen (Input) of om gegevens weg te schrijven (Output). Het bestand krijgt een nummer toegewezen om te bewerken.

*b) Write*

Write *#nummer*, *variabele1*, *variabele2*, ...
Een gegevenreeks uit een reeks van variabelen wegschrijven naar een bestand.

*c) Input*

Input #*nummer*, *variabele1*, *variabele2*, ...
Een gegevenreeks lezen uit een bestand en toewijzen aan een reeks variabelen.

*d) Close*

Close #*nummer*
Een geopend bestand terug sluiten.

***Form***

*a) Load*

Load *form*
Een form in het geheugen laden maar nog niet op het scherm brengen.

*b) UnLoad*

UnLoad *form*
Een form terug van het scherm verwijderen en het ingenomen geheugen vrijmaken.

### 3.2.7 Standaardfuncties

Functies die standaard in Visual Basic aanwezig zijn en kunnen worden gebruikt in de procedures.

***Algemeen***

*a) MsgBox*

{*Variabele* =} MsgBox (*tekst*{, *opties*, *titel*})
Functie die een dialoogvenster op het scherm brengt met een tekst en een waarde van de aangeklikte besturingsknop teruggeeft.

*b) UBound*

{*Variabele*=}UBound(*array*, *dimensie*)
Functie die de grootte van een matrix teruggeeft.

*c) Format*

{*String*=}Format(*varibele*, *vorm*)
Functie die een string teruggeeft die in een bepaalde vorm gegoten is.

*d) CDate*

{*Date*=}CDate(*variabele*)
Zet een variabele om naar een date.

*e) DateDiff*

{*Variabele*=}DateDiff(*string*, *date1*, *date2*)
Berekend het tijdsverschil tussen 2 data in de eenheid in de string. ("s" staat voor seconden)

***Wiskundig***

*a) Cos*

{*Variabele* =} Cos (*variabele*)
Functie die de cosinus van een hoek berekent.

*a) Sin*

{*Variabele* =} Sin (*variabele*)
Functie die de sinus van een hoek berekent.

*b) Isnumeric*

{*Boolean* =} Isnumeric (*string*)
Functie die onderzoekt of een variabele numerisch is en een waarde teruggeeft.

## 3.3 Programmacode

### 3.3.1 Algemeen

De module HoofdMod bevat de algemene declaraties van de algemene variabelen en types en de startprocedure (Main).
Er worden 2 publieke matrices van klassen gedeclareerd voor de algemene invoergegevens in op te slaan (zie 3.3.2), Lijst (PlaneetType) voor de planeten waarbij Lijst(0) functie doet als de Zon en Objecten (ObjectType) voor de objecten. Er wordt daarbij een variabele BeginDatum (Date) gedeclareerd.
Tenslotte worden er nog 3 constanten gedeclareerd, Pi (double) met waarde 3.14159265358979, de gravitatieconstante G (single) met waarde 0.00000000006672 en DtStr (string) om datums te formatteren met waarde "dd/mm/yyyy hh:nn".

*a) Main*

De startprocedure waarmee het programma begint.

| |
|---|
| HoofdForm.Show |
| OpenForm.Show(1) |

```
Sub Main() 'Startprocedure van het programma
HoofdForm.Show 'HoofdForm tonen
OpenForm.Show 1 'OpenForm tonen
End Sub
```

*b) Getgegevens*

Hulpfunctie die tussen een planeet en object kiest en teruggeeft. Deze functie heeft 1 parameter n (integer).

| n < 0 | |
|---|---|
| Ja | Neen |
| GEEF TERUG Object(-n) | GEEF TERUG Lijst(n + 1) |

```
Public Function GetGegevens(n As Integer) As Object
If n < 0 Then
Set GetGegevens = Objecten(-n)
Else
Set GetGegevens = Lijst(n + 1)
End If
End Function
```

### *Bewerking van bestanden*

Alle procedures in verband met het bewerken van bestanden is ondergebracht in de module BestandsMod. Deze bevat een procedure om een nieuw project aan te maken, gegevens uit een bestand te lezen en om ernaar weg te schrijven.
Deze module bevat 2 publieke variabelen bestand (string) en projectnummer (integer).

*a) Nieuw*

Deze procedure maakt een nieuw project aan.

| |
|---|
| GROOTTE(Lijst) = 0 tot 0 |
| Lijst(0).Naam = "Zon" |
| GROOTTE(Objecten) = 0 tot 0 |
| ProjectNummer = ProjectNummer + 1 |
| Bestand = "Project " & ProjectNummer |

```
Public Sub Nieuw() 'Procedure bij nieuw project
ReDim Lijst(0) 'lijst naar 0 zetten
Lijst(0).naam = "Zon" 'eerste record aan de zon toewijzen
ReDim Objecten(0)
projectnummer = projectnummer + 1
bestand = "Project " & projectnummer 'tijdelijke naam toewijzen
End Sub
```

*a) Schrijven*

De procedure schrijven schrijft het project weg naar een bestand.
De procedure bevat een parameter BestandsNaam (string) en een teller a (integer).

| niet(bestandsnaam = niets) | |
|---|---|
| Ja | Neen |
| OPEN bestand met naam bestandsnaam voor uitvoer als #1 | |
| Bestand = BestandsNaam | |
| SCHRIJF "***Baanbeschrijvingsproject***" naar #1 | |
| SCHRIJF GROOTTE(Lijst) naar #1 | |
| VOOR a = 0 tot GROOTTE(Lijst) | |
| SCHRIJF gegevens van Lijst(a) naar #1 | |
| SCHRIJF GROOTTE(Objecten) naar #1 | |
| VOOR a = 1 tot GROOTTE(Objecten) | |
| SCHRIJF gegevens van Objecten(a) naar #1 | |

| | |
|---|---|
| SLUIT #1 | |

```
Public Sub Schrijven(BestandsNaam As String) 'Procedure voor wegschrijven bestand
If Not (BestandsNaam = "") Then
Bestand = BestandsNaam 'bestandsnaam opslaan
Dim a As Integer
Open Bestand For Output As #1 'bestand openen of maken
Write #1, "***Baanbeschrijvingsproject***"
Write #1, UBound(Lijst, 1) 'grootte lijst wegschrijven
Write #1, BeginDatum 'begindatum wegschrijven
For a = 0 To UBound(Lijst, 1)
'gegevens schrijven
Lijst(a).Serialize 1, True
Next a
Write #1, UBound(Objecten, 1)
For a = 1 To UBound(Objecten, 1)
'gegevens schrijven
Objecten(a).Serialize 1, True
Next a
Close #1 'bestand sluiten
End If
End Sub
```

*b) Lezen*

Deze procedure leest de gegevens uit een bestand terug in het geheugen.
Er is een parameter BestandsNaam (string) en 3 lokale variabelen Controle (string), a en n (integer).

niet(BestandsNaam = "")

Ja | Neen

OPEN bestand met naam bestandsnaam voor invoer als #1

LEES Controle uit #1

Controle = "***Baanbeschrijvingsproject***"

Ja | Neen

LEES n uit #1 | BOODSCHAP Ongeldig bestand voor Baanbeschrijving

GROOTTE(Lijst) = 0 tot n

LEES BeginDatum uit #1

VOOR a van 0 tot n

LEES gegevens van Lijst(a) uit #1

LEES n uit #1

n > 0

Ja | Neen

GROOTTE (Objecten) = 1 tot n | GROOTTE (Objecten) = 0 tot 0

VOOR a van 1 tot n

LEES gegevens van Objecten(a) uit #1

SLUIT #1

```
Public Sub Lezen(BestandsNaam As String) 'Procedure voor inlezen bestand
If Not (BestandsNaam = "") Then
Open BestandsNaam For Input As #1 'bestand openen
Dim Controle As String
Input #1, Controle
If Controle = "***Baanbeschrijvingsproject***" Then
Bestand = BestandsNaam 'bestandsnaam opslaan
Dim n As Integer
Input #1, n 'grootte lijst lezen
ReDim Lijst(n) 'grootte lijst instellen
Input #1, BeginDatum 'begindatum inlezen
Dim a As Integer
For a = 0 To n
'gegevens lezen
Lijst(a).Serialize 1, False
Next a
Input #1, n
If n > 0 Then
ReDim Objecten(1 To n)
For a = 1 To n
'gegevens lezen
Objecten(a).Serialize 1, False
Next a
Else
ReDim Objecten(0)
End If
Else
MsgBox "Ongeldig bestand voor Baanbeschrijving", vbCritical, "Fout" 'fout bij ongeldig bestand
End If
Close #1 'bestand sluiten
End If
End Sub
```

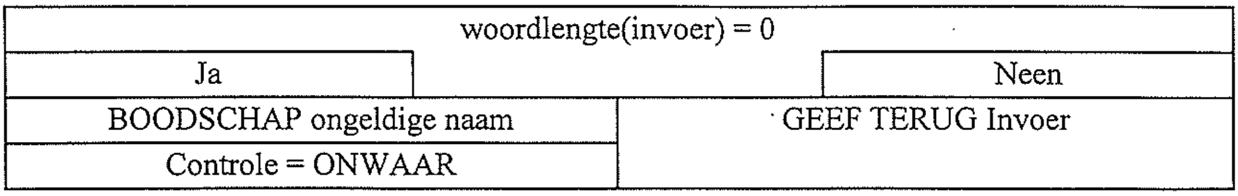
### *Controleprocedures*

De controleprocedures zijn ondergebracht in de module ContMod. De procedures controleren of de invoergegevens correct zijn en geven een foutbericht bij verkeerde invoer. Deze module bevat ook de declaratie het enumerieke type GetalType.

```
Public Enum GetalType
Reëel = 0
Positief_Reëel = 1
Strikt_Positief_Reëel = 2
Strikt_Positief_Geheel = 3
End Enum
```

#### *a) Contnaam*

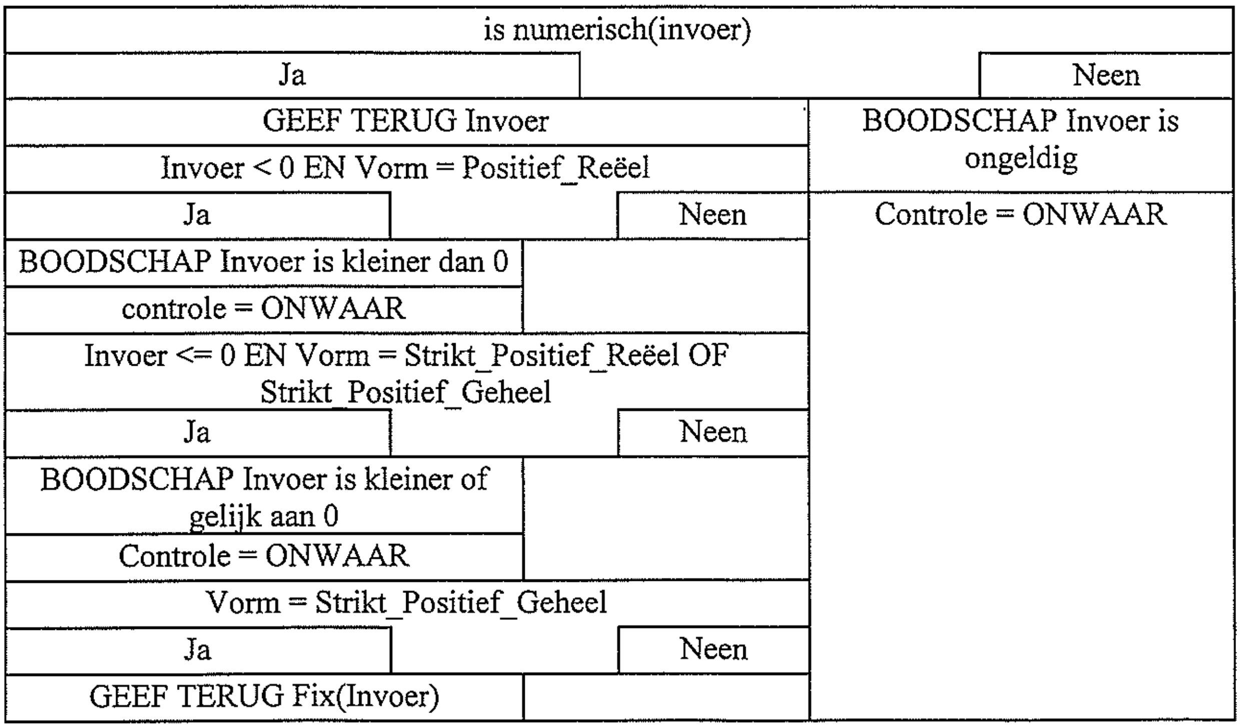
Controleert of de invoer een geldig woord is. De procedure bevat 2 parameters namelijk Invoer en Controle (boolean) en geeft een string terug.




```
Public Function ContNaam(Controle As Boolean, Invoer As String) As String
If Len(Invoer) = 0 Then 'controleren of de lengte van de invoer gelijk is aan 0
MsgBox "ongeldige naam", vbExclamation, "Fout" 'boodschap bij fout
Controle = False 'controle onwaar maken
Else
ContNaam = Invoer 'als de lengte van de invoer groter is dan 0 toewijzen aan woord
End If
End Function
```

### *a) Contfout*

Controleert of de invoer een geldig getal is en geeft het getal terug. De functie bevat 4 parameters Invoer (variant), Vorm (GetalType), Naam (string) en Controle (boolean).




```
Public Function ContFout(Invoer, Vorm As GetalType, Optional Naam As String, Optional Controle As Boolean)
On Error GoTo Fout
If IsNumeric(Invoer) Then 'controleren of de invoer numerisch is
ContFout = Invoer 'als de invoer numerisch is toewijzen aan getal
If (Invoer < 0) And (Vorm = 1) Then 'bijkomende controle
If Not IsMissing(Naam) Then MsgBox Naam & " is kleiner dan nul", vbExclamation, "Fout" 'boodschap bij fout
If Not IsMissing(Controle) Then Controle = False 'controle onwaar maken
End If
If (Invoer <= 0) And (Vorm >= 2) Then 'bijkomende controle
If Not IsMissing(Naam) Then MsgBox Naam & " is kleiner of gelijk aan nul", vbExclamation, "Fout" 'boodschap bij fout
```

```
If Not IsMissing(Controle) Then Controle = False 'controle onwaar maken
End If
If (Vorm = 3) Then ContFout = Fix(ContFout)
Else
If Not IsMissing(Naam) Then MsgBox Naam & " is ongeldig", vbExclamation, "Fout" 'boodschap bij fout
If Not IsMissing(Controle) Then Controle = False 'controle onwaar maken
End If
Exit Function
Fout:
ContFout = 0
If Not IsMissing(Naam) Then MsgBox Naam & " is ongeldig", vbExclamation, "Fout" 'boodschap bij fout
If Not IsMissing(Controle) Then Controle = False 'controle onwaar maken
End Function
```

### *HoofdForm*

HoofdForm is het hoofdvenster van het programma. Het is een MDI-form waarin de uitvoervensters als kindvensters in zichtbaar worden gemaakt.
Het venster heeft een menu dat toegang biedt tot de invoer en uitvoer van het programma. Links op het venster is er een balk waarmee de uitvoer geregeld wordt (zie 3.3.4). Deze balk bevat een opdrachtknop Bereken, een combobox Uitvoer om te kiezen tussen de verschillende uitvoermogelijkheden, 2 optionbutons Planeet en Object om te kiezen tussen uitvoer van planeten of objecten, een listbox LijstUit met de planeten of objecten en 3 textboxen Begin, Einde, Stappen en een combobox Resultaat om de uitvoergegevens in te geven.

*a) Klik op nieuw project in menu*

<table>
<tr><td colspan="2">MsgBox("Wilt u de wijzigingen in " & Bestand & " eerst opslaan")</td></tr>
<tr><td>vbCancel</td><td>vbYes</td></tr>
<tr><td>VERLAAT PROCEDURE</td><td>Opslaan_Click</td></tr>
<tr><td colspan="2">Nieuw</td></tr>
<tr><td colspan="2">HoofdForm.Caption = "Baanbeschrijving - " & Bestand</td></tr>
</table>

```
Private Sub Nieuw_Click() 'bestand/nieuw
Select Case MsgBox("Wilt u de wijzigingen in " & Bestand & " eerst opslaan", vbExclamation +_
vbYesNoCancel, "Bestand sluiten")
Case vbCancel
Exit Sub
Case vbYes
Opslaan_Click
End Select
End Sub
```

*b) Klik op project openen in menu*

<table>
<tr><td colspan="2">MsgBox("Wilt u de wijzigingen in " & Bestand & " eerst opslaan")</td></tr>
<tr><td>vbCancel</td><td>vbYes</td></tr>
<tr><td>VERLAAT PROCEDURE</td><td>Opslaan_Click</td></tr>
<tr><td colspan="2">DialoogVenster.DialogTitle = "Project openen"</td></tr>
<tr><td colspan="2">DialoogVenster.Filter = "Baanbeschrijvingsproject (*.BBP)|*.bbp|Alle bestanden (*.*)|*.*"</td></tr>
<tr><td colspan="2">DialoogVenster.ShowOpen</td></tr>
<tr><td colspan="2">Lezen(DialoogVenster.FileName)</td></tr>
</table>

| |
|---|
| HoofdForm.Caption = "Baanbeschrijving - " & Bestand |

```
Private Sub openen_Click() 'bestand/openen
Select Case MsgBox("Wilt u de wijzigingen in " & Bestand & " eerst opslaan", vbExclamation +_
vbYesNoCancel, "Bestand sluiten")
Case vbCancel
Exit Sub
Case vbYes
Opslaan_Click
End Select
AllesSluit
DialoogVenster.DialogTitle = "Project openen"
DialoogVenster.Filter = "Baanbeschrijvingsproject (*.BBP)|*.bbp|Alle bestanden (*.*)|*.*"
DialoogVenster.ShowOpen
Lezen DialoogVenster.FileName
HoofdForm.Caption = "Baanbeschrijving - " & Bestand
BeginSit
End Sub
```

*c) Klik op Project opslaan in menu*

| |
|---|
| DialoogVenster.DialogTitle = "Project opslaan" |
| DialoogVenster.Filter = "Baanbeschrijvingsproject (*.BBP)\|*.bbp\|Alle bestanden (*.*)\|*.*" |
| DialoogVenster.ShowSave |
| Schrijven(DialoogVenster.FileName) |

```
Private Sub Opslaan_Click() 'bestand/opslaan
DialoogVenster.DialogTitle = "Project opslaan"
DialoogVenster.Filter = "Baanbeschrijvingsproject (*.BBP)|*.bbp|Alle bestanden (*.*)|*.*"
DialoogVenster.ShowSave
Schrijven DialoogVenster.FileName
End Sub
```

### 3.3.2 Berekeningen

De procedures voor de berekeningen zijn ondergebracht als methodes van de klassen PlaneetType en ObjectType. Er is tevens een klasse, ToestandType, om de toestand op een bepaald moment in op te slaan.

***PlaneetType***

PlaneetType is de klasse waarin de variabelen van een planeet worden in opgeslagen. Het bevat eveneens methodes om de toestand op een bepaald moment uit te rekenen.

*a) Eigenschappen*

```
Public Naam As String
Public Massa As Double
Public Straal As Double
Public RevolutieDraaizin As Boolean
Public BaanStraal As Double
Public HoekX As Single 'regulushoek
Public Excentriciteit As Single
Public BeginHoek As Single
```

### *b) Hoek in functie van de tijd*

Deze functie berekent de hoek ten opzichte van de Zon. De excentriciteit van de baan van een planeet is zeer klein waardoor men de hoeksnelheid als constant kan aanzien en de berekeningen eenvoudiger zijn.

| | |
|---|---|
| $\text{HoekFTijd} = \text{TijdStip} * \sqrt{\dfrac{G * \text{Lijst(0).Massa}}{(\text{BaanStraal} * 1000)^3}}$ | |
| RevolutieDraaizin | |
| Ja | Neen |
| HoekFTijd = -HoekFTijd | |
| GEEF TERUG HoekFTijd + BeginHoek | |

```
Public Function HoekFTijd(TijdStip As Single) As Single 'berekenen hoek in functie van tijd
HoekFTijd = TijdStip * (G * Lijst(0).Massa / (BaanStraal * 1000) ^ 3) ^ (1 / 2)
If RevolutieDraaizin Then HoekFTijd = -HoekFTijd
HoekFTijd = HoekFTijd + BeginHoek
End Function
```

### *c) Positie*

Berekent de positie van de planeet in functie van de tijd. Deze methode heeft 1 parameter Tijdstip (single) en geeft een variabele van het type ToestandType terug.

| |
|---|
| GEEF TERUG PositieHoek(HoekFTijd(TijdStip)) |

```
Public Function Positie(TijdStip As Single) As ToestandType
Set Positie = PositieHoek(HoekFTijd(TijdStip))
End Function
```

### *d) Positie in functie van de hoek*

Berekent de positie in functie van de hoek en geeft die terug als een ToestandType.

| |
|---|
| PositieHoek.Hoek = Hoek |
| $\text{PositieHoek.r} = \dfrac{\text{BaanStraal} * \left(1 - \text{Excentriciteit}^2\right)}{1 + \text{Excentriciteit} * \text{Cos}(\text{Hoek} - \text{HoekX} * \text{Pi}/180)}$ |
| PositieHoek.X = PositieHoek.r * Cos(Hoek) |
| PositieHoek.Y = PositieHoek.r * Sin(Hoek) |

```
Public Function PositieHoek(Hoek As Single) As ToestandType
Set PositieHoek = New ToestandType
PositieHoek.Hoek = Hoek
PositieHoek.r = BaanStraal * (1 - Excentriciteit ^ 2) / (1 + Excentriciteit * Cos(Hoek - HoekX * Pi / 180))
PositieHoek.X = PositieHoek.r * Cos(Hoek)
PositieHoek.Y = PositieHoek.r * Sin(Hoek)
End Function
```

### *e) Toestand*

:ent de toestand op basis van een tijdstip die wordt doorgegeven als parameter en geeft :rug. Deze methode heeft 1 variabele met de naam Rico (single).

| | |
|---|---|
| Toestand = Positie(TijdStip) | |
| $Toestand.v = \sqrt{\frac{G * Lijst(0).Massa}{1000} * \left(\frac{2}{Toestand.r} - \frac{1}{BaanStraal}\right)}$ | |
| Toestand.Y <> 0 | |
| Ja | Neen |
| ɔ = Atn(-(1 - Excentriciteit ^ 2) * (Excentriciteit * aanStraal + 2 * Toestand.X) / (2 * Toestand.Y)) | Toestand.vy = Toestand.v |
| Toestand.vx = Toestand.v * Cos(Rico) | |
| Toestand.vy = Toestand.v * Sin(Rico) | |
| Toestand.a = G * Lijst(0).Massa / (Toestand.r * 1000) ^ 2 | |
| Toestand.ax = Toestand.a * Toestand.X / Toestand.r | |
| Toestand.ay = Toestand.a * Toestand.Y / Toestand.r | |

```
ıblic Function Toestand(TijdStip As Single) As ToestandType
:t Toestand = Positie(TijdStip)
ɔestand.v = (G * Lijst(0).Massa * (2 / Toestand.r - 1 / BaanStraal) / 1000) ^ (1 / 2)
im Rico As Single
'Toestand.Y <> 0 Then
ico = Atn(-(1 - Excentriciteit ^ 2) * (Excentriciteit * BaanStraal + 2 * Toestand.X) / (2 * Toestand.Y))
'oestand.vx = Toestand.v * Cos(Rico)
'oestand.vy = Toestand.v * Sin(Rico)
:lse
Toestand.vy = Toestand.v
End If
Toestand.a = G * Lijst(0).Massa / (Toestand.r * 1000) ^ 2
Toestand.ax = Toestand.a * Toestand.X / Toestand.r
Toestand.ay = Toestand.a * Toestand.Y / Toestand.r
End Function
```

### *f) Revolutietijd*

Berekent de tijd die nodig is voor de planeet om 1 revolutie rond de zon te doen.

| |
|---|
| $GEEF\ TERUG\ 2 * Pi * \sqrt{\frac{(1000 * BaanStraal)^3}{Lijst(0).Massa * G}}$ |

```
Public Function RevolutieTijd() As Single 'revolutietijd in s berekenen
RevolutieTijd = 2 * Pi * (BaanStraal * 1000) ^ (3 / 2) / (Lijst(0).Massa * G) ^ (1 / 2)
End Function
```

## ***ObjectType***

ObjectType is de klasse de objecten en heeft een gelijkaardige structuur als de klasse PlaneetType.

### *a) Eigenschappen*

```
Public Naam As String
```

```
Public Massa As Double
Public BaanSoort As Integer
Public StartPlaneet As Integer
Public StartDatum As Date
Public DoelPlaneet As Integer
```

### *b) Berekenen parameters van de baan*

Methode die de parameters van de baan berekent. Deze methode heeft 5 parameters; BaanStraal (double), Excentriciteit (single), RegulusHoek (single) en Pos1 en Pos2 (ToestandType).

| |
|---|
| Pos1 = Lijst(StartPlaneet).Positie(DateDiff("s", BeginDatum, StartDatum)) |
| RegulusHoek = Pos1.Hoek |
| Pos2 = Lijst(DoelPlaneet).PositieHoek(RegulusHoek + Pi) |
| Excentriciteit = (Pos2.r - Pos1.r) / (Pos1.r + Pos2.r) |
| BaanStraal = (Pos1.r + Pos2.r) / 2 |

```
Private Function BerekenVar(BaanStraal As Double, Optional Excentriciteit As Single, Optional_
RegulusHoek As Single, Optional Pos1 As ToestandType, Optional Pos2 As ToestandType) As Single
Set Pos1 = Lijst(StartPlaneet).Positie(DateDiff("s", BeginDatum, StartDatum))
RegulusHoek = Pos1.Hoek
Set Pos2 = Lijst(DoelPlaneet).PositieHoek(RegulusHoek + Pi)
Excentriciteit = (Pos2.r - Pos1.r) / (Pos1.r + Pos2.r)
BaanStraal = (Pos1.r + Pos2.r) / 2
End Function
```

### *c) Mogelijke startdatums berekenen*

Methode die de vertrekdatums berekent waarbij een Hohmannbaan mogelijk is. Het heeft 1 parameter k (integer) en 3 locale variabelen c1, c2 en T (double).

<table>
<tr><td colspan="2">c1 = (G * Lijst(0).Massa / (Lijst(StartPlaneet).BaanStraal * 1000) ^ 3) ^ (1 / 2)</td></tr>
<tr><td colspan="2">Lijst(StartPlaneet).RevolutieDraaizin</td></tr>
<tr><td>Ja</td><td>Neen</td></tr>
<tr><td>c1 = -c1</td><td></td></tr>
<tr><td colspan="2">c2 = (G * Lijst(0).Massa / (Lijst(DoelPlaneet).BaanStraal * 1000) ^ 3) ^ (1 / 2)</td></tr>
<tr><td colspan="2">Lijst(StartPlaneet).RevolutieDraaizin</td></tr>
<tr><td>Ja</td><td>Neen</td></tr>
<tr><td>c2 = -c2</td><td></td></tr>
<tr><td colspan="2">T = Pi * ((Lijst(StartPlaneet).BaanStraal + Lijst(DoelPlaneet).BaanStraal) * 500) ^ (3 / 2) / (Lijst(0).Massa * G) ^ (1 / 2)</td></tr>
<tr><td colspan="2">c1 - c2</td></tr>
<tr><td>Ja</td><td>Neen</td></tr>
<tr><td>k = -k</td><td></td></tr>
<tr><td colspan="2">GEEF TERUG DateAdd("s", ((Pi * (2 * k - 1) + c2 * T - Lijst(beginplaneet).BeginHoek + Lijst(DoelPlaneet).BeginHoek) / (c1 - c2)), BeginDatum)</td></tr>
</table>

```
Public Function MogelijkHohmann(ByVal k As Integer) As Date
Dim c1, c2 As Double
```

```
Dim T As Double
c1 = (G * Lijst(0).Massa / (Lijst(StartPlaneet).BaanStraal * 1000) ^ 3) ^ (1 / 2)
If Lijst(StartPlaneet).RevolutieDraaizin Then c1 = -c1
c2 = (G * Lijst(0).Massa / (Lijst(DoelPlaneet).BaanStraal * 1000) ^ 3) ^ (1 / 2)
If Lijst(StartPlaneet).RevolutieDraaizin Then c2 = -c2
T = Pi * ((Lijst(StartPlaneet).BaanStraal + Lijst(DoelPlaneet).BaanStraal) * 500) ^ (3 / 2) / (Lijst(0).Massa *
G) ^ (1 / 2)
If c1 - c2 < 0 Then k = -k
MogelijkHohmann = DateAdd("s", ((Pi * (2 * k - 1) + c2 * T - Lijst(beginplaneet).BeginHoek +
Lijst(DoelPlaneet).BeginHoek) / (c1 - c2)), BeginDatum)
End Function
```

### *d) Positie*

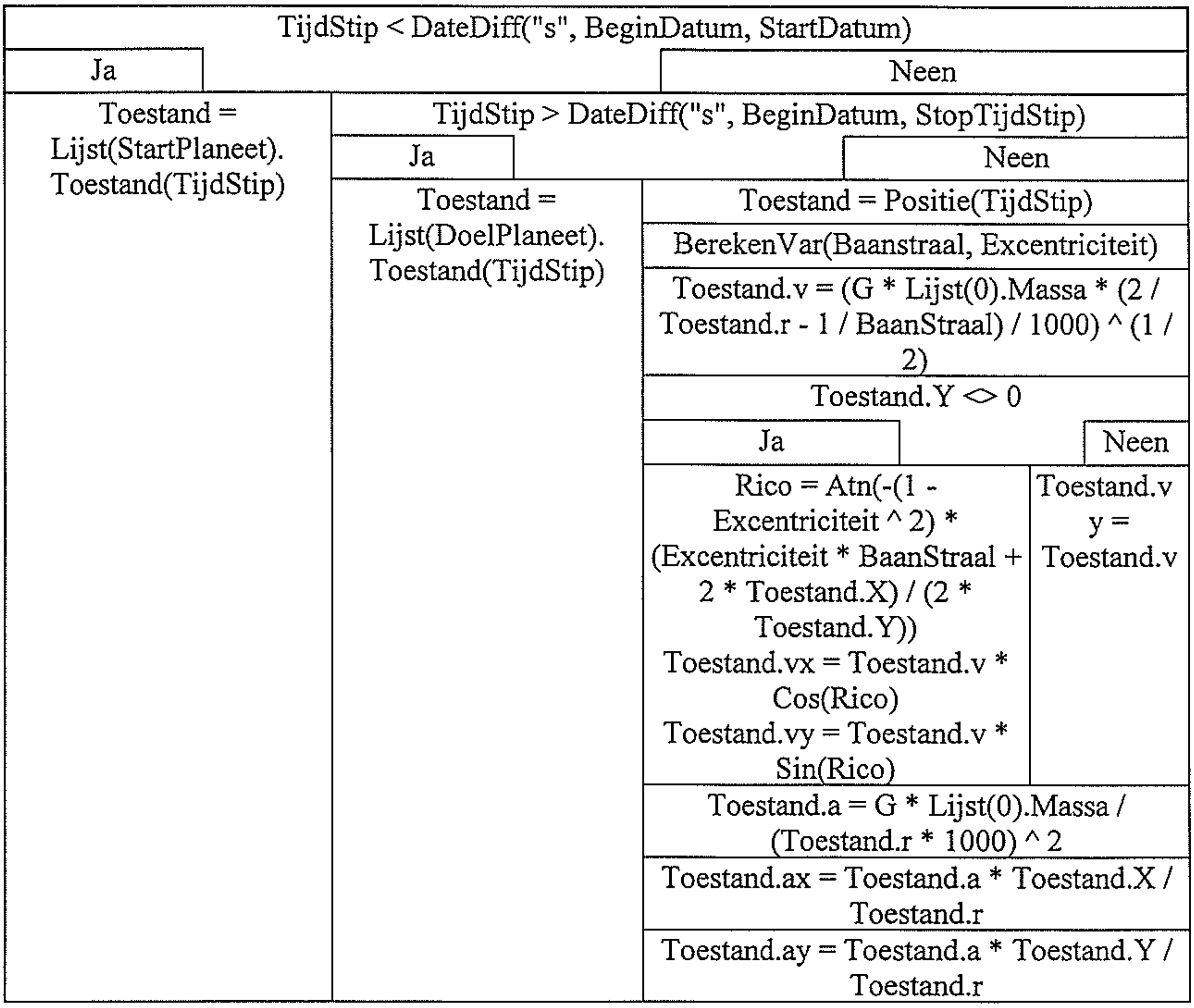
Berekent de positie van een object in een Hohmannbaan op basis van een bepaald tijdstip. Deze methode heeft 1 parameter Tijdstip (single) en geeft een variabele van het type ToestandType terug. Het bevat tenslotte nog 3locale variabelen BaanStraal (double), Excentriciteit (single) en RegulusHoek (single).

<table>
<tr><td colspan="3">TijdStip < DateDiff("s", BeginDatum, StartDatum)</td></tr>
<tr><td>Ja</td><td colspan="2">Neen</td></tr>
<tr><td rowspan="7">Positie = Lijst(StartPlaneet). Positie(TijdStip)</td><td colspan="2">TijdStip > DateDiff("s", BeginDatum, StopTijdStip)</td></tr>
<tr><td>Ja</td><td>Neen</td></tr>
<tr><td rowspan="5">Positie = Lijst(DoelPlaneet). Positie(TijdStip)</td><td>BerekenVar(BaanStraal, Excentriciteit, RegulusHoek)</td></tr>
<tr><td>Positie.Hoek = (TijdStip - DateDiff("s", BeginDatum, StartDatum)) * (G * Lijst(0).Massa / (BaanStraal *_ 1000) ^ 3) ^ (1 / 2) + RegulusHoek</td></tr>
<tr><td>Positie.r = BaanStraal * (1 - Excentriciteit ^ 2) / (1 + Excentriciteit * Cos(Positie.Hoek - RegulusHoek))</td></tr>
<tr><td>Positie.X = Positie.r * Cos(Positie.Hoek)</td></tr>
<tr><td>Positie.Y = Positie.r * Sin(Positie.Hoek)</td></tr>
</table>

```
Public Function Positie(TijdStip As Single) As ToestandType 'naar tijdstip
If TijdStip < DateDiff("s", BeginDatum, StartDatum) Then
Set Positie = Lijst(StartPlaneet).Positie(TijdStip)
ElseIf TijdStip > DateDiff("s", BeginDatum, StopTijdStip) Then
Set Positie = Lijst(DoelPlaneet).Positie(TijdStip)
Else
Dim BaanStraal As Double
Dim Excentriciteit As Single
Dim RegulusHoek As Single
Set Positie = New ToestandType
BerekenVar BaanStraal, Excentriciteit, RegulusHoek
Positie.Hoek = (TijdStip - DateDiff("s", BeginDatum, StartDatum)) * (G * Lijst(0).Massa / (BaanStraal *_
1000) ^ 3) ^ (1 / 2) + RegulusHoek
Positie.r = BaanStraal * (1 - Excentriciteit ^ 2) / (1 + Excentriciteit * Cos(Positie.Hoek - RegulusHoek))
Positie.X = Positie.r * Cos(Positie.Hoek)
Positie.Y = Positie.r * Sin(Positie.Hoek)
End If
End Function
```

*e) Toestand*

Methode die de toestand op een bepaald tijdstip teruggeeft. Het bevat 3 locale variabelen BaanStraal (double), Excentriciteit (single) en Rico (single) en heeft 1 parameter TijdStip (single).




```
Public Function Toestand(TijdStip As Single) As ToestandType
If TijdStip < DateDiff("s", BeginDatum, StartDatum) Then
Set Toestand = Lijst(StartPlaneet).Toestand(TijdStip)
ElseIf TijdStip > DateDiff("s", BeginDatum, StopTijdStip) Then
Set Toestand = Lijst(DoelPlaneet).Toestand(TijdStip)
Else
Set Toestand = Positie(TijdStip)
Dim BaanStraal As Double
Dim Excentriciteit As Single
BerekenVar BaanStraal, Excentriciteit
Toestand.v = (G * Lijst(0).Massa * (2 / Toestand.r - 1 / BaanStraal) / 1000) ^ (1 / 2)
Dim Rico As Single
If Toestand.Y <> 0 Then
Rico = Atn(-(1 - Excentriciteit ^ 2) * (Excentriciteit * BaanStraal + 2 * Toestand.X) / (2 * Toestand.Y))
Toestand.vx = Toestand.v * Cos(Rico)
Toestand.vy = Toestand.v * Sin(Rico)
Else
Toestand.vy = Toestand.v
End If
```

```
Toestand.a = G * Lijst(0).Massa / (Toestand.r * 1000) ^ 2
Toestand.ax = Toestand.a * Toestand.X / Toestand.r
Toestand.ay = Toestand.a * Toestand.Y / Toestand.r
End If
End Function
```

### *ToestandType*

*a) Eigenschappen*

```
Public Hoek As Single ' in rad
Public r As Double 'in km
Public x As Double
Public y As Double
Public v As Double 'in m/s
Public vx As Double
Public vy As Double
Public a As Double 'in m/s^2
Public ax As Double
Public ay As Double
```

*b) Kinetische energie*

```
Public Function Ek() As Double ' in J/kg
Ek = v ^ 2 / 2
End Function
```

*c) Potentiële energie*

```
Public Function Ep() As Double
Ep = G * Lijst(0).Massa / (r * 1000)
End Function
```

*d) Totale energie*

```
Public Function Etot() As Double
Etot = Abs(Ek - Ep)
End Function
```

## 3.3.3 Invoer

### ***Invoer van zonnestelsel***

De invoer van het zonnestelsel gebeurt met behulp van de form StelselForm en PlaneetForm. De form StelselForm bevat 1 globale variabele x (integer) die de te bewerken planeet bijhoudt. Het bevat 3 textboxen begindt om de begindatum op te geven, massa en straal om de karakteristieken van de zon op te geven.

*a) Laden StelslForm*

| |
|---|
| Massa.tekst = lijst(0).Massa |
| Straal.tekst = lijst(0).Straal |
| BeginDt.Text = Format(BeginDatum, DtStr) |

```
Private Sub Form_Load() ' Procedure bij het activeren van Stelselform
Opstellen 'lijst van planeten toewijzen aan PlaneetLijst
'alle besturingselementen op tablad Algemeen instellen
```

```
Massa.Text = Lijst(0).Massa
Straal.Text = Lijst(0).Straal
BeginDt.Text = Format(BeginDatum, DtStr)
End Sub
```

### *a) Klik op OK*

De OK-procedure slaat telkens de gegevens op. Het bevat 1 variabele Controle (boolean).

<table>
<tr><td colspan="2">Controle = True</td></tr>
<tr><td colspan="2">Lijst(0).Massa = Contfout(Massa.Text, Strikt_Positief_Reëel, "massa", controle)</td></tr>
<tr><td colspan="2">Lijst(0).Straal = Contfout(Straal.Text, Strikt_Positief_Reëel, "straal", controle)</td></tr>
<tr><td colspan="2">IsDate(BeginDt.Text)</td></tr>
<tr><td>Ja</td><td>Neen</td></tr>
<tr><td rowspan="2">BeginDatum = BeginDt.Text</td><td>Controle = False</td></tr>
<tr><td>BOODSCHAP begindatum/tijd is ongeldig</td></tr>
<tr><td colspan="2">Controle</td></tr>
<tr><td>Ja</td><td>Neen</td></tr>
<tr><td>UNLOAD(StelselForm)</td><td></td></tr>
</table>

```
Private Sub OK_Click()
Dim Controle As Boolean
Controle = True 'Controle voor fouten op waar zetten
'Zon toewijzen aan positie 0 in de lijst en controles uitvoeren
Lijst(0).Massa = ContFout(Massa.Text, Strikt_Positief_Reëel, "massa", Controle)
Lijst(0).Straal = ContFout(Straal.Text, Strikt_Positief_Reëel, "straal", Controle)
If IsDate(BeginDt.Text) Then
BeginDatum = BeginDt.Text
Else
Controle = False
MsgBox "begindatum/tijd is ongeldig", vbExclamation, "Fout"
End If
If Controle Then
Unload StelselForm 'Stelselform verwijderen van het scherm als er geen fouten zijn
End If
End Sub
```

### *b) klik op Toevoegen*

Deze procedure voegt een planeet toe aan de lijst.

<table>
<tr><td colspan="2">GROOTTE(Lijst) = GROOTTE(Lijst) + 1</td></tr>
<tr><td colspan="2">LOAD(PlaneetForm)</td></tr>
<tr><td colspan="2">PlaneetForm.x = GROOTTE(Lijst)</td></tr>
<tr><td colspan="2">PlaneetForm.Show 1</td></tr>
<tr><td colspan="2">Lijst(GROOTTE(Lijst)).naam = ""</td></tr>
<tr><td>Ja</td><td>Neen</td></tr>
<tr><td>GROOTTE(Lijst) = GROOTTE(Lijst) - 1</td><td></td></tr>
</table>

```
Private Sub Toevoegen_Click() 'Planeet toevoegen aan de lijst
ReDim Preserve Lijst(UBound(Lijst, 1) + 1)
```

```
Load PlaneetForm
PlaneetForm.X = UBound(Lijst, 1)
PlaneetForm.Show 1
If Lijst(UBound(Lijst, 1)).naam = "" Then ReDim Preserve Lijst(UBound(Lijst, 1) - 1)
Opstellen
End Sub
```

*c) Klik op Wijzigen*

Deze procedure staat toe de parameters van een planeet in de lijst te wijzigen.

| X > 0 | |
|---|---|
| Ja | Neen |
| LOAD(PlaneetForm) | |
| PlaneetForm.x = x | |
| PlaneetForm.Show(1) | |

```
Private Sub Wijzigen_Click() 'Planeet in de lijst veranderen
If X > 0 Then
Load PlaneetForm
PlaneetForm.X = X
PlaneetForm.Show 1
End If
End Sub
```

*d) Klik op Verwijderen*

Deze procedure verwijdert een planeet uit de lijst.

| X > 0 | |
|---|---|
| Ja | Neen |
| Lijst(x) = Lijst(GROOTTE(Lijst)) | |
| GROOTTE(Lijst) = GROOTTE(Lijst) - 1 | |

```
Private Sub Verwijderen_Click()
If X > 0 Then
Set Lijst(X) = Lijst(UBound(Lijst, 1))
ReDim Preserve Lijst(UBound(Lijst, 1) - 1)
Opstellen
End If
End Sub
```

### *Invoer van een planeet*

De invoer van de parameters van een planeet gebeurt met behulp van de form StelselForm. De form bevat een publieke variabele x (integer) en een private variabele Planeet (PlaneetType).

*a) Activeren PlaneetForm*

| |
|---|
| Naam.Text = Lijst(X).Naam |
| Massa.Text = Lijst(X).Massa |
| Straal.Text = Lijst(X).Straal |

| | |
|---|---|
| BaanStraal.Text = Lijst(X).BaanStraal | |
| HoekX.Text = Lijst(X).HoekX | |
| Excentriciteit.Text = Lijst(X).Excentriciteit | |
| BeginHoek.Text = 180 * Lijst(X).BeginHoek / Pi | |
| Lijst(X).RevolutieDraaizin | |
| Ja | Neen |
| RevolutieWI.Value = WAAR | RevolutieTWI.Value = WAAR |

```
Private Sub Form_Activate()
'eigenschappen instellen
Naam.Text = Lijst(X).Naam
Massa.Text = Lijst(X).Massa
Straal.Text = Lijst(X).Straal
BaanStraal.Text = Lijst(X).BaanStraal
HoekX.Text = Lijst(X).HoekX
Excentriciteit.Text = Lijst(X).Excentriciteit
BeginHoek.Text = 180 * Lijst(X).BeginHoek / Pi
If Lijst(X).RevolutieDraaizin Then RevolutieWI.Value = True Else RevolutieTWI.Value = True
End Sub
```

*a) klik op OK*

| | |
|---|---|
| Controle = WAAR | |
| Toewijs(Controle) | |
| Controle | |
| Ja | Neen |
| Lijst(X) = Planeet | |
| UNLOAD(PlaneetForm) | |

```
Private Sub OK_Click()
Dim Controle As Boolean
Controle = True
'eigenschappen opslaan in tijdelijke variabele met controle
Toewijs Controle
If Controle Then
Set Lijst(X) = Planeet
Unload PlaneetForm
End If
End Sub
```

*b) Toewijs*

Procedure die de parameters controleert en toewijst aan de variabele Planeet. Deze procedure heeft 1 parameter Controle(boolean).

| |
|---|
| Planeet.Naam = ContNaam(Controle, Naam.Text) |
| Planeet.Massa = ContFout(Massa.Text, Strikt_Positief_Reëel, "massa", Controle) |
| Planeet.Straal = ContFout(Straal.Text, Strikt_Positief_Reëel, "straal", Controle) |
| Planeet.BaanStraal = ContFout(BaanStraal.Text, Strikt_Positief_Reëel, "baanstraal", Controle) |
| Planeet.HoekX = ContFout(HoekX.Text, Reëel, "X-hoek", Controle) |

| |
|---|
| Planeet.Excentriciteit = ContFout(Excentriciteit.Text, Positief_Reëel, "excentriciteit", Controle) |
| Planeet.BeginHoek = Pi * ContFout(BeginHoek.Text, Reëel, "beginhoek", Controle) / 180 |
| Planeet.RevolutieDraaizin = RevolutieWI.Value |

```
Public Sub Toewijs(Controle As Boolean)
Planeet.Naam = ContNaam(Controle, Naam.Text)
Planeet.Massa = ContFout(Massa.Text, Strikt_Positief_Reëel, "massa", Controle)
Planeet.Straal = ContFout(Straal.Text, Strikt_Positief_Reëel, "straal", Controle)
Planeet.BaanStraal = ContFout(BaanStraal.Text, Strikt_Positief_Reëel, "baanstraal", Controle)
Planeet.HoekX = ContFout(HoekX.Text, Reëel, "X-hoek", Controle)
Planeet.Excentriciteit = ContFout(Excentriciteit.Text, Positief_Reëel, "excentriciteit", Controle)
Planeet.BeginHoek = Pi * ContFout(BeginHoek.Text, Reëel, "beginhoek", Controle) / 180
Planeet.RevolutieDraaizin = RevolutieWI.Value
End Sub
```

### *Invoer van objecten*

De invoer van de objecten gebeurt met behulp van de form ObjectenForm en ObjectForm. De form ObjectenForm bevat eveneens 1 globale variabele x (integer) die het te bewerken object bijhoudt.

*a) klik op Toevoegen*

<table>
<tr><td colspan="2">GROOTTE(Objecten) = GROOTTE(Objecten) + 1</td></tr>
<tr><td colspan="2">LOAD(ObjectForm)</td></tr>
<tr><td colspan="2">ObjectForm.x = GROOTTE(Objecten)</td></tr>
<tr><td colspan="2">ObjectForm.Show 1</td></tr>
<tr><td colspan="2">Objecten (GROOTTE(Objecten)).naam = ""</td></tr>
<tr><td>Ja</td><td>Neen</td></tr>
<tr><td>GROOTTE(Objecten) = GROOTTE(Objecten) - 1</td><td></td></tr>
</table>

```
Private Sub Toevoegen_Click()
If Not (UBound(Objecten, 1) = 0) Then
ReDim Preserve Objecten(1 To UBound(Objecten, 1) + 1)
Else
ReDim Objecten(1 To 1)
End If
Load ObjectForm
ObjectForm.X = UBound(Objecten, 1)
ObjectForm.Show 1
If Objecten(UBound(Objecten, 1)).Naam = "" Then
If UBound(Objecten, 1) > 1 Then
ReDim Preserve Objecten(1 To UBound(Objecten, 1) - 1)
Else
ReDim Objecten(0)
End If
End If
Opstellen
End Sub
```

*a) Klik op Wijzigen*

| |
|---|
| X > 0 |

| Ja | Neen |
|---|---|
| LOAD(ObjectForm) | |
| ObjectForm.x = x | |
| ObjectForm.Show(1) | |

```
Private Sub Wijzigen_Click()
If X > 0 Then
Load ObjectForm
ObjectForm.X = X
ObjectForm.Show 1
End If
Opstellen
End Sub
```

*b) Klik op Verwijderen*

| X > 0 | |
|---|---|
| Ja | Neen |
| Objecten(x) = Objecten(GROOTTE(Objecten)) | |
| GROOTTE(Objecten) = GROOTTE(Objecten) - 1 | |

```
Private Sub Verwijderen_Click()
If X > 0 Then
If UBound(Objecten, 1) > 1 Then
Set Objecten(X) = Objecten(UBound(Objecten, 1))
ReDim Preserve Objecten(1 To UBound(Objecten, 1) - 1)
Else
ReDim Objecten(0)
End If
Opstellen
End If
End Sub
```

### *Invoer van een object*

De invoer van de parameters van een object gebeurt met behulp van PlaneetForm. Deze form heeft 2 variabelen nl. x(integer) en Object(ObjectType).

*a) Activeren ObjectForm*

| |
|---|
| Naam.Text = Objecten(X).Naam |
| Massa.Text = Objecten(X).Massa |
| BaanSoort.ListIndex = Objecten(X).BaanSoort |
| VOOR n van 0 tot GROOTTE(Lijst) |
| StartPlaneet.AddItem(Lijst(n).Naam) |
| DoelPlaneet.AddItem(Lijst(n).Naam) |
| StartPlaneet.ListIndex = Objecten(X).StartPlaneet - 1 |
| StartDatum.Text = Format(Objecten(X).StartDatum, DtStr) |
| DoelPlaneet.ListIndex = Objecten(X).DoelPlaneet - 1 |

```
Private Sub Form_Activate()
Dim n As Integer
Naam.Text = Objecten(X).Naam
Massa.Text = Objecten(X).Massa
BaanSoort.ListIndex = Objecten(X).BaanSoort
For n = 1 To UBound(Lijst, 1)
StartPlaneet.AddItem Lijst(n).Naam
DoelPlaneet.AddItem Lijst(n).Naam
Next n
On Error Resume Next
StartPlaneet.ListIndex = Objecten(X).StartPlaneet - 1
StartDatum.Text = Format(Objecten(X).StartDatum, DtStr)
DoelPlaneet.ListIndex = Objecten(X).DoelPlaneet - 1
End Sub
```

*b) klik op OK*

<table>
<tr><td colspan="2">IsDate(StartDatum.Text)</td></tr>
<tr><td>Ja</td><td>Neen</td></tr>
<tr><td>Object.StartDatum = CDate(StartDatum.Text)</td><td rowspan="2">BOODSCHAP startdatum is ongeldig</td></tr>
<tr><td>Controle = WAAR</td></tr>
<tr><td colspan="2">Object.Naam = ContNaam(Controle, Naam.Text)</td></tr>
<tr><td colspan="2">Object.Massa = ContFout(Massa.Text, Strikt_Positief_Reëel, "massa", Controle)</td></tr>
<tr><td colspan="2">Object.BaanSoort = BaanSoort.ListIndex</td></tr>
<tr><td colspan="2">Object.StartPlaneet = StartPlaneet.ListIndex + 1</td></tr>
<tr><td colspan="2">Object.DoelPlaneet = DoelPlaneet.ListIndex + 1</td></tr>
<tr><td colspan="2">Controle</td></tr>
<tr><td>Ja</td><td>Neen</td></tr>
<tr><td>Objecten(X) = Object</td><td rowspan="2"></td></tr>
<tr><td>Unload(ObjectForm)</td></tr>
</table>

```
Private Sub OK_Click()
Dim Controle As Boolean
If IsDate(StartDatum.Text) Then
Object.StartDatum = CDate(StartDatum.Text)
Controle = True
Else
MsgBox "startdatum is ongeldig", vbExclamation, "Fout"
End If
Object.Naam = ContNaam(Controle, Naam.Text)
Object.Massa = ContFout(Massa.Text, Strikt_Positief_Reëel, "massa", Controle)
Object.BaanSoort = BaanSoort.ListIndex
Object.StartPlaneet = StartPlaneet.ListIndex + 1
Object.DoelPlaneet = DoelPlaneet.ListIndex + 1
If Controle Then
Set Objecten(X) = Object
Unload ObjectForm
End If
End Sub
```

### 3.3.4 Uitvoer

*a) klik op bereken*

Deze gebeurtenis kiest tussen de uitvoermogelijkheden aan de hand van het geselecteerde item in de combobox Uitvoer en doet de nodige berekeningen uivoeren.

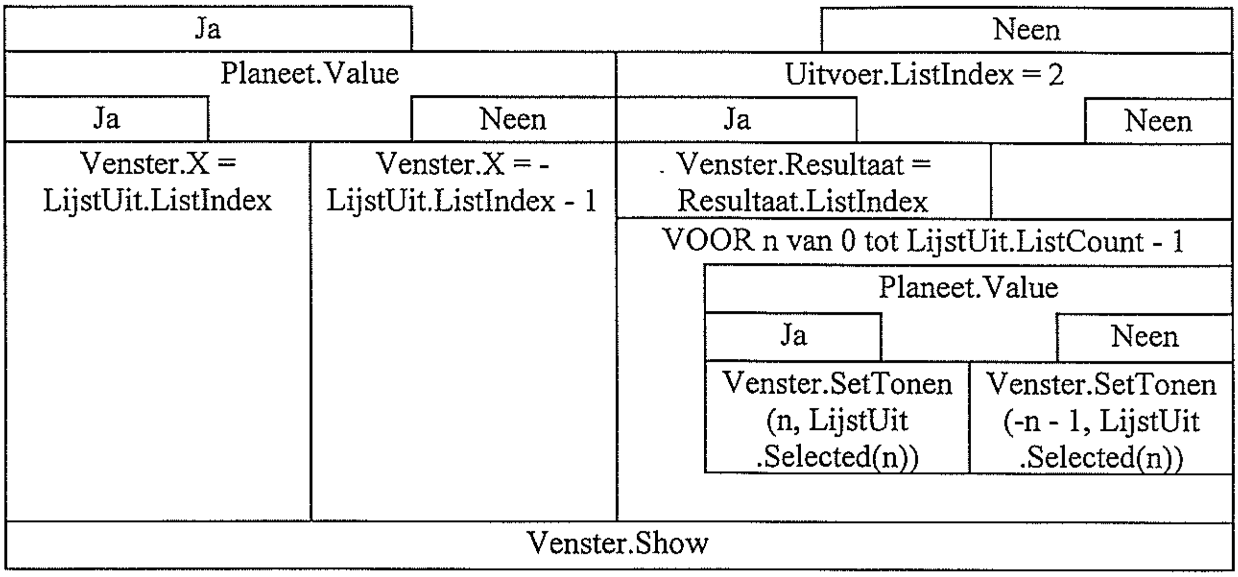
| Uitvoer.ListIndex | | | |
|---|---|---|---|
| 0 | 1 | 2 | 3 |
| SetUitvoer (AlgemeenForm) | SetUitvoer (TabelForm) | SetUitvoer (GrafiekForm) | GrafischForm.Dag = GrafischForm.Begin |
| AlgemeenForm .Bereken | TabelForm.Bereken | GrafiekForm.Bereken | GrafischForm .Tekenen |
| | | | SimulatieForm.Show , HoofdForm |
| | | | SimulatieForm.Instel |

```
Private Sub Bereken_Click()
Select Case Uitvoer.ListIndex
Case 0
SetUitvoer AlgemeenForm
AlgemeenForm.Bereken
Case 1
SetUitvoer TabelForm
TabelForm.Bereken
Case 2
SetUitvoer GrafiekForm
GrafiekForm.Bereken
Case 3
SetUitvoer GrafischForm
GrafischForm.Dag = GrafischForm.Begin
GrafischForm.Tekenen
SimulatieForm.Show , HoofdForm
SimulatieForm.Instel
End Select
End Sub
```

*b) Setuitvoer*

De methode setuitvoer van HoofdForm slaat de gegevens voor de uitvoer op in de bijhorende form en brengt ze op het scherm. De procedure heeft 1 parameter Venster (form) en 1 locale variabele n (integer).

| |
|---|
| LOAD(Venster) |
| Venster.SetUit(Object.Value) |
| Venster.Begin = CDate(Begin.Text) |
| Venster.Einde = CDate(Einde.Text) |
| Venster.Stappen = Fix(Stappen.Text) |
| Uitvoer.ListIndex < 2 |




```
Public Sub SetUitvoer(Venster As Form)
If Not NietUit Then
Load Venster
Venster.SetUit Object.Value
Venster.Begin = CDate(Begin.Text)
Venster.Einde = CDate(Einde.Text)
Venster.Stappen = Fix(Stappen.Text)
If Uitvoer.ListIndex < 2 Then
If Planeet.Value Then
Venster.X = LijstUit.ListIndex
Else
Venster.X = -LijstUit.ListIndex - 1
End If
Else
Dim n As Integer
If Uitvoer.ListIndex = 2 Then Venster.Resultaat = Resultaat.ListIndex
For n = 0 To LijstUit.ListCount - 1
If Planeet.Value Then
Venster.SetTonen n, LijstUit.Selected(n)
Else
Venster.SetTonen -n - 1, LijstUit.Selected(n)
End If
Next n
End If
Venster.Show
End If
End Sub
```

### *Algemene uivoer*

De algemene resultaten zijn te verkrijgen met behulp van AlgemeenForm. Op AlgemeenForm bevindt zich 1 textbox met de naam Uitvoer waarop de algemene uitvoergegevens worden op afgedrukt. Het heeft 4 globale variabelen: Begin en Einde (date) die de begin- en einddatum bijhouden, Stappen (integer) en x (integer) die het nummer van de planeet of het object bijhoudt.

*a) Laden AlgemeenForm*

| Begin = BeginDatum |
|---|

| |
|---|
| Einde = BeginDatum |

```
Private Sub Form_Load()
HoofdForm.AlgemeenTonen.Checked = True

Begin = BeginDatum
Einde = BeginDatum
End Sub
```

*b) Bereken*

Deze methode berekent de algemene uitvoer en stelt de tekst in van de tekstbox Uitvoer. Het heeft 1 locale variabele Toestand (ToestandType).

| |
|---|
| Uitvoer.Text = GetGegevens(x).Naam |
| Uitvoer.Text = Uitvoer.Text & GetGegevens(x).UitvoerTekst & vbCrLf |
| Toestand = GetGegevens(x).Toestand(DateDiff("s", BeginDatum, Begin)) |
| Uitvoer.Text = Uitvoer.Text & vbCrLf & "Begin:" & vbCrLf |
| Uitvoer.Text = Uitvoer.Text & Toestand.UitvoerTekst |
| Toestand = GetGegevens(x).Toestand(DateDiff("s", BeginDatum, Einde)) |
| Uitvoer.Text = Uitvoer.Text & vbCrLf & "Einde:" & vbCrLf |
| Uitvoer.Text = Uitvoer.Text & Toestand.UitvoerTekst |

```
Public Sub Bereken()
Dim Toestand As ToestandType
Uitvoer.Text = GetGegevens(x).Naam
Uitvoer.Text = Uitvoer.Text & GetGegevens(x).UitvoerTekst & vbCrLf
Set Toestand = GetGegevens(x).Toestand(DateDiff("s", BeginDatum, Begin))
Uitvoer.Text = Uitvoer.Text & vbCrLf & "Begin:" & vbCrLf
Uitvoer.Text = Uitvoer.Text & Toestand.UitvoerTekst
Set Toestand = GetGegevens(x).Toestand(DateDiff("s", BeginDatum, Einde))
Uitvoer.Text = Uitvoer.Text & vbCrLf & "Einde:" & vbCrLf
Uitvoer.Text = Uitvoer.Text & Toestand.UitvoerTekst
End Sub
```

### *Tabel*

Er kan een tabel met uitvoergegevens van 1 planeet of object in functie van het tijdstip worden opgesteld met behulp van de form Tabelform. Deze bevat een flexgrid-besturingselement die de tabel voorstelt. Het heeft eveneens 4 globale variabelen: Begin en Einde (date), Stappen (integer) en x (integer).

*a) Laden TabelForm*

Er worden 2 locale variabelen gebuikt bij het laden van deze form: n (integer) als teller, hoofding (variant) om de hoofding van de tabel in te stellen.

| |
|---|
| Begin = BeginDatum |
| Einde = BeginDatum |
| Stappen = 100 |
| hoofding = Array("Tijdstip", "Hoek (rad)", "r (km)", "X (km)", "Y (km)", "V (m/s)", "Vx (m/s)","Vy (m/s)", "a (m/s^2)", "ax (m/s^2)", "ay (m/s^2)", "Ep (J)", "Ek (J)", "Etot (J)") |

| Tabel.Row = 0 | |
|---|---|
| VOOR n van 1 tot 14 | |
| | Tabel.ColWidth(n) = 2000 |
| | Tabel.Col = n |
| | Tabel.Text = hoofding(n - 1) |

```
Private Sub Form_Load()
HoofdForm.TabelTonen.Checked = True

Begin = BeginDatum
Einde = BeginDatum
Stappen = 100
Dim hoofding
hoofding = Array("Tijdstip", "Hoek (rad)", "r (km)", "X (km)", "Y (km)", "V (m/s)", "Vx (m/s)", "Vy (m/s)",_
"a (m/s^2)", "ax (m/s^2)", "ay (m/s^2)", "Ep (J)", "Ek (J)", "Etot (J)")
Dim n As Integer
Tabel.Row = 0
For n = 1 To 17
Tabel.ColWidth(n) = 2000
Tabel.Col = n
Tabel.Text = hoofding(n - 1)
Next n
End Sub
```

*b) Bereken*

De methode bereken stelt de tabel op en heeft 3 variabelen n (integer), Tijdstip (single) en Toestand (ToestandType).

| Tabel.Rows = 1 | |
|---|---|
| VOOR n van 0 tot Stappen | |
| | TijdStip = n * DateDiff("s", Begin, Einde) / Stappen - DateDiff("s", BeginDatum, Begin) |
| | Toestand = GetGegevens(X).Toestand(TijdStip) |
| | Tabel.AddItem uitvoerresultaten van Toestand |

```
Public Sub Bereken()
Dim n As Integer
Dim TijdStip As Single
Dim Toestand As ToestandType
Tabel.Rows = 1
For n = 0 To Stappen
TijdStip = n * DateDiff("s", Begin, Einde) / Stappen - DateDiff("s", BeginDatum, Begin)
Set Toestand = GetGegevens(X).Toestand(TijdStip)
Tabel.AddItem n & Chr(9) & Format(DateAdd("n", TijdStip / 60, Begin), DtStr) & Chr(9) & Toestand.Hoek_
& Chr(9) & Toestand.r & Chr(9) & Toestand.X & Chr(9) & Toestand.Y & Chr(9) & Toestand.v & Chr(9) &_
Toestand.vx & Chr(9) & Toestand.vy & Chr(9) & Toestand.a & Chr(9) & Toestand.ax & Chr(9) &_
Toestand.ay & Chr(9) & Toestand.Ep & Chr(9) & Toestand.Ek & Chr(9) & Toestand.EtotToestand.Etot
Next n
End Sub
```

### *Grafiek*

GrafiekForm kan een grafiek opstellen van een toestandsvariabele in functie van de tijd. Deze Form bevat een zelfgemaakt besturingselement AsstelselCont (niet verder besproken) met naam Grafiek waarop de grafiek wordt getekend. De varabelen van deze form zijn: Begin en Einde (date), stappen (integer), resultaat (integer) en zichtbaar (dynamische array van booleans).

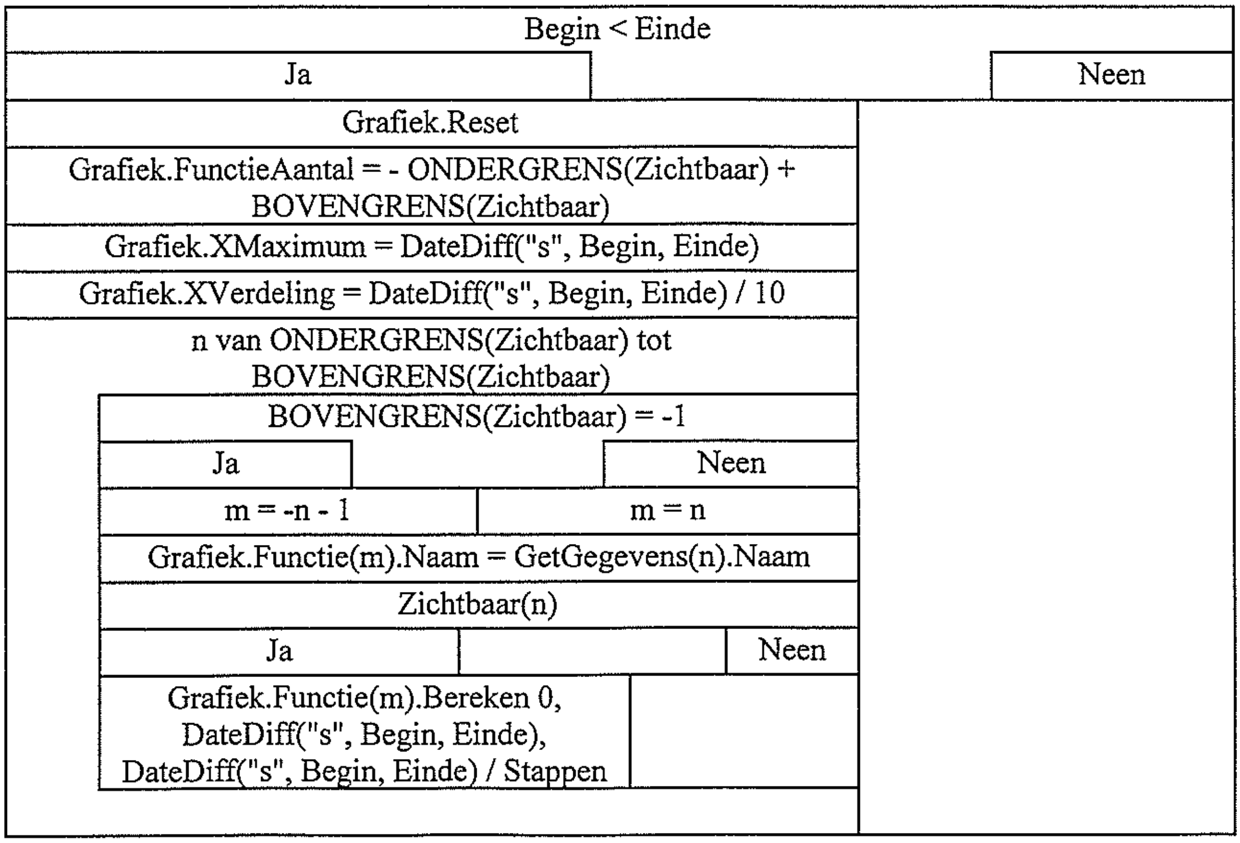
*a) Laden GrafiekForm*

| |
|---|
| Begin = BeginDatum |
| Einde = BeginDatum |
| Stappen = 100 |
| GROOTTE(Zichtbaar) = 0 tot GROOTTE(Lijst) - 1 |

```
Private Sub Form_Load()
HoofdForm.GrafiekTonen.Checked = True

Begin = BeginDatum
Einde = BeginDatum
Stappen = 100
ReDim Zichtbaar(0 To UBound(Lijst, 1) - 1)
End Sub
```

*b) Bereken*

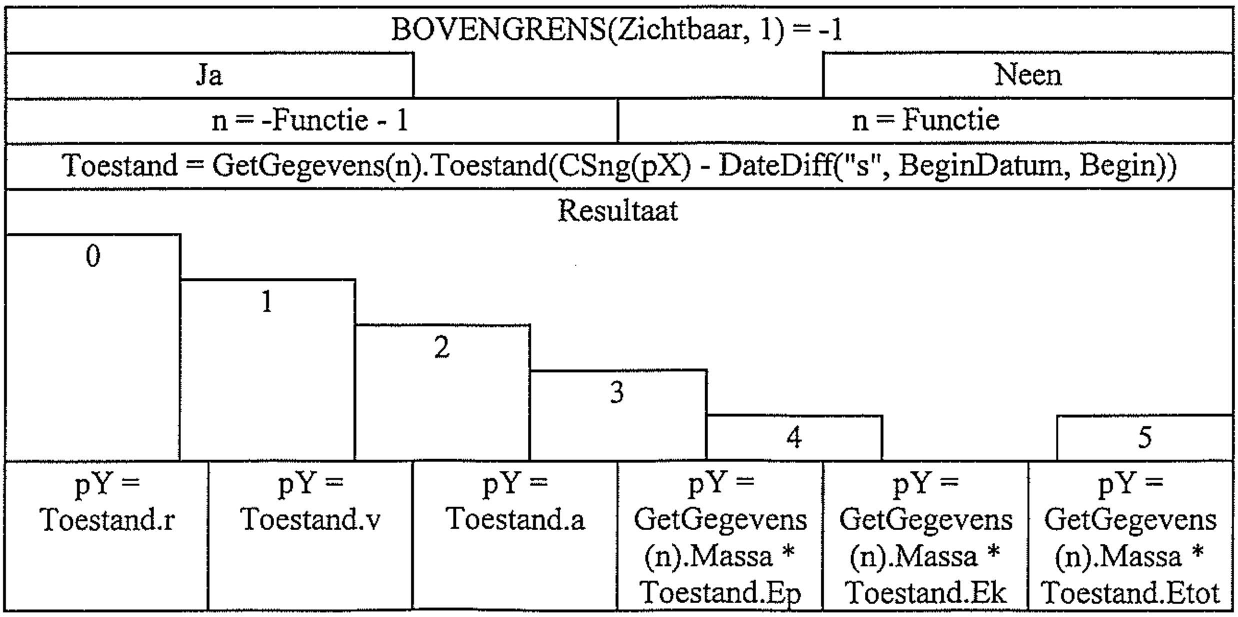



```
Public Sub Bereken()
Dim n As Integer
Dim m As Byte
```

```
If Begin < Einde Then
Grafiek.Reset
Select Case Resultaat
Case 0
Grafiek.YNaam = "r (km)"
Case 1
Grafiek.YNaam = "v (m/s)"
Case 2
Grafiek.YNaam = "a (m/s^2)"
Case 3
Grafiek.YNaam = "Ep (J)"
Case 4
Grafiek.YNaam = "Ek (J)"
Case 5
Grafiek.YNaam = "Etot (J)"
End Select
Grafiek.FunctieAantal = -LBound(Zichtbaar, 1) + UBound(Zichtbaar, 1)
Grafiek.XMaximum = DateDiff("s", Begin, Einde)
Grafiek.XVerdeling = DateDiff("s", Begin, Einde) / 10
For n = LBound(Zichtbaar, 1) To UBound(Zichtbaar, 1)
If UBound(Zichtbaar, 1) = -1 Then
m = -n - 1
Else
m = n
End If
Grafiek.Functie(m).Naam = GetGegevens(n).Naam
If Zichtbaar(n) Then
Grafiek.Functie(m).Bereken 0, DateDiff("s", Begin, Einde), DateDiff("s", Begin, Einde) / Stappen
End If
Next n
End If
End Sub
```

### *c) Berekening*

De gebeurtenis Grafiek_Berekening wordt automatisch aangeroepen als de methode Bereken wordt uitgevoerd en berekent en slaat de waarden voor x en y op. Er worden 2 locale variabelen gebruikt: Toestand (ToestandType) en n (integer). De gebeurtenis heeft 3 parameters: pX en pY (double) en Functie (Byte).



```
Private Sub Grafiek_Berekening(ByVal pX As Double, pY As Double, ByVal Functie As Byte)
Dim Toestand As ToestandType
Dim n As Integer
If UBound(Zichtbaar, 1) = -1 Then
n = -Functie - 1
Else
n = Functie
End If
Set Toestand = GetGegevens(n).Toestand(CSng(pX) - DateDiff("s", BeginDatum, Begin))
Select Case Resultaat
Case 0
pY = Toestand.r
Case 1
pY = Toestand.v
Case 2
pY = Toestand.a
Case 3
pY = GetGegevens(n).Massa * Toestand.Ep
Case 4
pY = GetGegevens(n).Massa * Toestand.Ek
Case 5
pY = GetGegevens(n).Massa * Toestand.Etot
End Select
End Sub
```

*d) SetTonen*

```
Public Sub SetTonen(ByVal n As Integer, Toon As Boolean)
Zichtbaar(n) = Toon
End Sub
```

### *Grafische uitvoer*

De grafische uitvoer gebeurt met behulp van de form GrafischForm die een picturebox GrafischTek bevat waarin de visuele voorstelling van de banen in wordt getekend. Het heeft tevens 7 variabelen: Dag (date) met de huidig afgebeelde datum, Schaal (double), Begin en Einde (date) met respectievelijk de begindatum en einddatum voor de simulatie, Stappen (integer), Zichtbaar (dynamische array van booleans) met of de objecten al dan niet zichtbaar zijn, Figuren (dynamische array van controls) met vormen die de objecten voorstellen.

*a) Laden GrafischForm*

Deze gebeurtenisprocedure heeft 1 locale variabele m (integer).

| |
|---|
| Dag = BeginDatum |
| Begin = BeginDatum |
| Einde = BeginDatum |
| Schaal = 100000 |
| Stappen = 100 |
| (GROOTTE(Objecten) + GROOTTE(Lijst)) > 0 |
| GROOTTE(Zichtbaar) = -GROOTTE(Objecten) tot GROOTTE(Lijst) - 1 |
| GROOTTE(Figuren) = -GROOTTE(Objecten) tot GROOTTE(Lijst) - 1 |

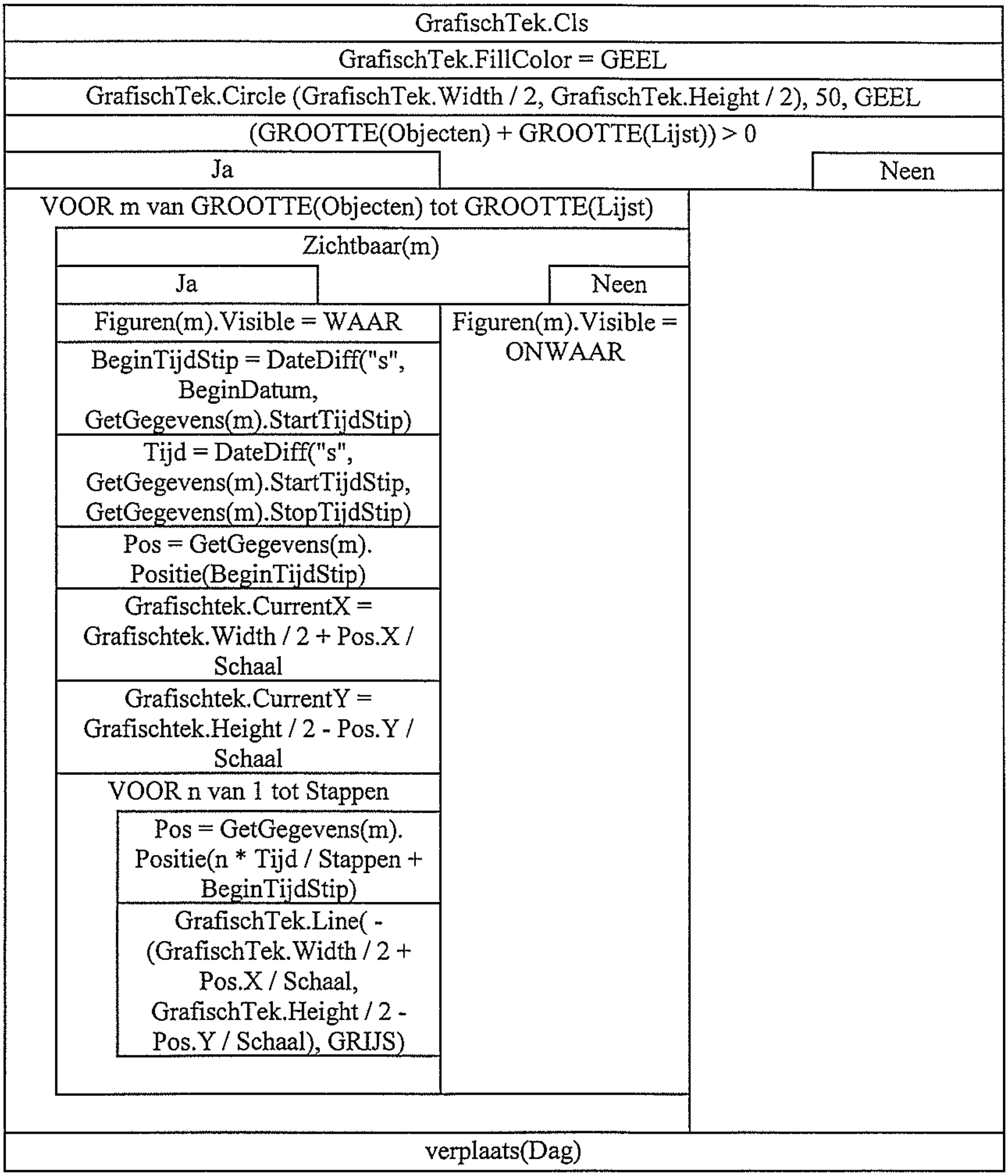
| m = van -GROOTTE(Objecten) tot GROOTTE(Lijst, 1) - 1 | |
|---|---|
| Zichtbaar(m) = WAAR | |
| Figuren(m) = Controls.Add("vb.shape", "figuren" & CStr(m + GROOTTE(Objecten))) | |
| Figuren(m).Container = GrafischTek | |
| Figuren(m).Shape = 3 | |
| Figuren(m).FillStyle = 0 | |
| m < 0 | |
| Ja | Neen |
| Figuren(m).Width = 100 | Figuren(m).Width = 150 |
| Figuren(m).Height = 100 | Figuren(m).Height = 150 |
| Figuren(m).FillColor = ROOD | Figuren(m).FillColor = BLAUW |

```
Private Sub Form_Load()
HoofdForm.GrafischTonen.Checked = True
HoofdForm.GrafInstelTonen.Enabled = True
HoofdForm.SimTonen.Enabled = True

Dag = BeginDatum
Begin = BeginDatum
Einde = BeginDatum
Schaal = 100000
Stappen = 100
Dim m As Integer
If (UBound(Objecten, 1) + UBound(Lijst, 1)) > 0 Then
ReDim Zichtbaar(-UBound(Objecten) To UBound(Lijst, 1) - 1)
ReDim Figuren(-UBound(Objecten) To UBound(Lijst, 1) - 1)
For m = -UBound(Objecten) To UBound(Lijst, 1) - 1
Zichtbaar(m) = True
Set Figuren(m) = Controls.Add("vb.shape", "figuren" & CStr(m + UBound(Objecten)))
With Figuren(m)
Set .Container = GrafischTek
.Shape = 3
.FillStyle = 0
If m < 0 Then
.Width = 100
.Height = 100
.FillColor = vbRed
Else
.Width = 150
.Height = 150
.FillColor = vbBlue
End If
End With
Next m
End If
End Sub
```

*b) Tekenen*

Deze procedure tekent de banen van de planeten en objecten als ze zichtbaar zijn. Deze procedure heeft 5 locale variabelen: m en n (integer), BeginTijdstip en Tijd (single) en Pos (ToestandType).




```
Public Sub Tekenen()
Dim m As Integer
Dim n As Integer
Dim BeginTijdStip As Single
Dim Tijd As Single
Dim Pos As ToestandType
GrafischTek.Cls
GrafischTek.Circle (GrafischTek.Width / 2, GrafischTek.Height / 2), 75, vbYellow
If (UBound(Objecten, 1) + UBound(Lijst, 1)) > 0 Then
For m = -UBound(Objecten, 1) To UBound(Lijst, 1) - 1
If Zichtbaar(m) Then
```

```
Figuren(m).Visible = True
BeginTijdStip = DateDiff("s", BeginDatum, GetGegevens(m).StartTijdStip)
Tijd = DateDiff("s", GetGegevens(m).StartTijdStip, GetGegevens(m).StopTijdStip)
Set Pos = GetGegevens(m).Positie(BeginTijdStip)
GrafischTek.CurrentX = GrafischTek.Width / 2 + Pos.X / Schaal
GrafischTek.CurrentY = GrafischTek.Height / 2 - Pos.Y / Schaal
For n = 1 To Stappen
Set Pos = GetGegevens(m).Positie(n * Tijd / Stappen + BeginTijdStip)
GrafischTek.Line -(GrafischTek.Width / 2 + Pos.X / Schaal, GrafischTek.Height / 2 - Pos.Y / Schaal),
8421504
Next n
Else
Figuren(m).Visible = False
End If
Next m
End If
Verplaats Dag
End Sub
```

*c) Verplaats*

De procedure verplaats verplaatst de planeten en objecten naar de positie die ze zouden hebben op een bepaalde datum. Het heeft 1 parameter Datum (date) en 3 variabelen m (integer), Tijdstip (single), Pos (ToestandType).

| (GROOTTE(Objecten) + GROOTTE(Lijst)) > 0 | |
|---|---|
| Ja | Neen |
| Tijdstip = DateDiff("s", BeginDatum, Datum) | |
| VOOR m van -GROOTTE(Objecten) tot GROOTTE(Lijst) | |
| Pos = GetGegevens(m).Positie(TijdStip) | |
| Figuren(m).Left = Grafischtek.Width / 2 + posbaan.X / Schaal - Figuren(m).Width / 2 | |
| Figuren(m).Top = Grafischtek.Height / 2 - posbaan.Y / Schaal - Figuren(m).Height / 2 | |

```
Public Sub Verplaats(Datum As Date)
Dag = Datum
Dim m As Integer
Dim TijdStip As Single
Dim Pos As ToestandType
If (UBound(Objecten, 1) + UBound(Lijst, 1)) > 0 Then
TijdStip = DateDiff("s", BeginDatum, Datum)
For m = -UBound(Objecten, 1) To UBound(Lijst, 1) - 1
Set Pos = GetGegevens(m).Positie(TijdStip)
Figuren(m).Left = GrafischTek.Width / 2 + Pos.X / Schaal - Figuren(m).Width / 2
Figuren(m).Top = GrafischTek.Height / 2 - Pos.Y / Schaal - Figuren(m).Height / 2
Next m
End If
HuidigLab.Caption = Format(Dag, DtStr)
End Sub
```

*d) SetTonen*

```
Public Sub SetTonen(ByVal n As Integer, Toon As Boolean)
```

```
Zichtbaar(n) = Toon
End Sub
```

# *Appendix*

## A. Kegelsneden

De ellips is de meetkundige plaats (verzameling) van de punten, waarvan de som van de afstanden tot 2 vaste punten (brandpunten) constant is. Zijn $F_1$ en $F_2$ de brandpunten, 2a de waarde van de constante som, dan geldt voor elk punt Q van de ellips: $|F_1Q| + |F_2Q| = Cte = 2a$

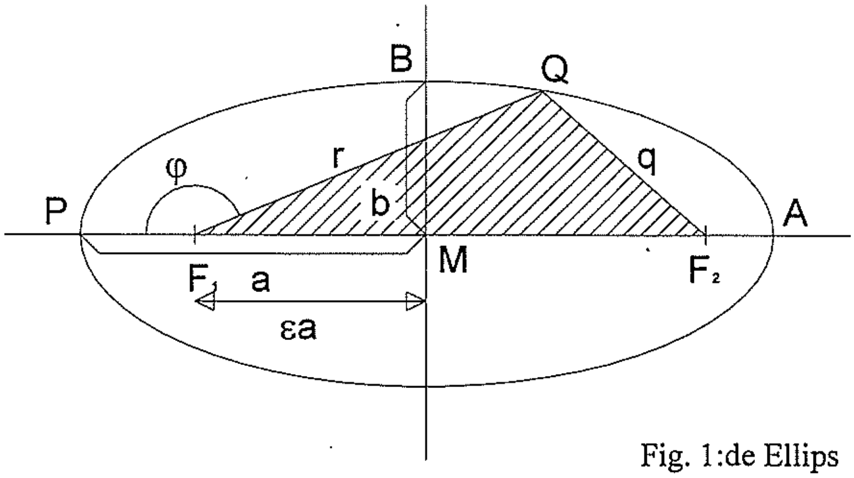



Fig. 1:de Ellips

$$|F_1Q| + |F_2Q| = Cte = 2a$$

Nu is de excentriciteit per definitie:

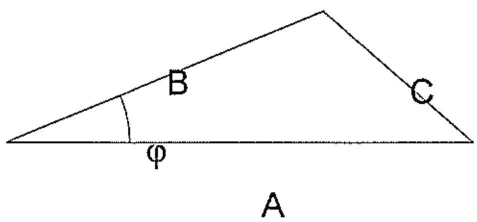
$$\varepsilon = \frac{F_1M}{a} \rightarrow FM = a\varepsilon$$

Uit driehoek FMB halen we:

$$b^2 + (a\varepsilon)^2 = a^2$$

$$b^2 + a^2\varepsilon^2 = a^2 \rightarrow \varepsilon = \frac{\sqrt{a^2 - b^2}}{a}$$

Uit de cos-regel halen we:




$$C^2 = A^2 + B^2 - 2AB\cos\varphi$$

Toegepast op Fig. 1:de ellips krijgen we:

$$q^2 = r^2 + (2\varepsilon a)^2 + 2r2\varepsilon a\cos\varphi$$
$$(2a-r)^2 = r^2 + 4\varepsilon^2 a^2 + 4\varepsilon ra\cos\varphi$$
$$4a^2 - 4ar + r^2 = r^2 + 4\varepsilon^2 a^2 + 4\varepsilon ra\cos\varphi$$
$$a^2 - \varepsilon^2 a^2 = ar + \varepsilon ra\cos\varphi$$
$$a^2(1-\varepsilon^2) = ar(1+\varepsilon\cos\varphi)$$

$$r = \frac{a(1-\varepsilon^2)}{1+\varepsilon\cos\varphi}$$

indien: $p = a(1-\varepsilon^2) \rightarrow r = \dfrac{p}{1+\varepsilon\cos\varphi}$

Dit kunnen we annaloog bewijzen voor de parabool en de hyperbool.

***Stelling***

De grafiek van een kwadratische vergelijking is algemeen :

$$Ax^2 + 2Bxy + Cy^2 + Dx + Ey + F = 0$$

waarvoor geldt

a) $AC - B^2 > 0 \rightarrow$ ellips, punt of $\varnothing$
b) $AC - B^2 < 0 \rightarrow$ hyperbool, 2 snijdende rechten
c) $AC - B^2 = 0 \rightarrow$ parabool, 2 evenwijdige rechten, 1 rechte, $\varnothing$

$r = \dfrac{p}{1+e\cos\vartheta}$ kegelsnede $(0 \le \vartheta \le 2\pi)$

$x = r\cos\vartheta$
$y = r\sin\vartheta$ r, $\vartheta$ poolcoördinaten
$\Leftrightarrow x^2 + y^2 = r^2$
$\Leftrightarrow r = \dfrac{px}{r+ex}$
$\Leftrightarrow \quad r + ex = p$
$\Leftrightarrow \quad r = p - ex$
$\Leftrightarrow \quad r^2 = (p-ex)^2$
$\Leftrightarrow \quad x^2 + y^2 = p^2 - 2pex + e^2x^2$
$\Leftrightarrow \quad (1-e^2)x^2 + y^2 + 2pex - p^2 = 0$

Nu is in bovenstaande vergelijking

$$AC - B^2 = 1 - e^2,$$

dus

a) $1 - e^2 > 0$ : ellips
$\rightarrow 0 < e < 1$

b) $1-e^2=0$ : parabool
$\rightarrow e=1$
c) $1-e^2<0$ : hyperbool
$\rightarrow e>1$

## B. Tabel met de parameters van de planeten

| Planeet | Massa ($kg x 10^{exp}$) | Diameter (km) | Halve grote as * | Omloop-tijd ** | Baanhel-ling (°) *** | Excentriciteit van de baan | Helling (°) |
|---|---|---|---|---|---|---|---|
| Zon | 1.99 (30) | 1.391.400 | - | - | - | - | 7.25 |
| Mercurius | 3.30 (23) | 4878 | 0.387 | 87.97 | 7.00 | 0.206 | 0 |
| Venus | 4.87 (24) | 12104 | 0.723 | 224.70 | 3.39 | 0.007 | 178 |
| Aarde | 5.98 (24) | 12756 | 1.000 | 365.26 | 0.00 | 0.017 | 23.45 |
| Maan | 7.35 (22) | 3476 | 384.4 | 27.32 | 18 à 29 | 0.055 | 6.68 |
| Mars | 6.42 (23) | 6796 | 1.523 | 686.98 | 1.85 | 0.093 | 23.98 |
| Jupiter | 1.90 (27) | 143800 | 5.203 | 11.86A | 1.30 | 0.049 | 3.07 |
| Saturnus | 5.69 (26) | 120660 | 9.539 | 29.46A | 2.49 | 0.056 | 27 |
| Uranus | 8.66 (25) | 52290 | 19.182 | 84.01A | 0.77 | 0.047 | 97 |
| Neptunus | 1.03 (26) | 49500 | 3.057 | 164.79A | 1.77 | 0.009 | 28.8 |
| Pluto | (1) (22) | 2400-3800 | 39.44 | 247.7A | 17.2 | 0.250 | (65?) |

* AE voor planeten; $10^3$ km voor de Maan.
** in dagen behalve wanneer speciaal aangeduid
*** t.o.v. de ecliptica voor planeten; t.o.v. de equator van de Aarde voor de Maan

| Planeet | Aswentelingtijd (dagen; R= teruglopende zin) | Gemiddelde baansnelheid (km/s) | Ontsnappingssnelheid aan het opp. (m/s) | Soortelijk gewicht ($kg/m^3$) |
|---|---|---|---|---|
| Zon | 25.38 | - | 617800 | 1410 |
| Mercurius | 58.65 | 47.9 | 4300 | 5420 |
| Venus | 243.01R | 35.0 | 10300 | 5250 |
| Aarde | 1 | 29.8 | 11200 | 5520 |
| Maan | 27.32 | - | 2380 | 3340 |
| Mars | 1.03 | 24.1 | 5000 | 3940 |
| Jupiter | 0.410 | 13.1 | 59500 | 1314 |
| Saturnus | 0.426 | 9.7 | 35600 | 690 |
| Uranus | (0.45?) | 6.8 | 21200 | (1190) |
| Neptunus | (0.66?) | 5.4 km/s | 23800 | 1660 |
| Pluto | (6.39?) | 4.7 km/s | (1270?) | ? |

## C. Vertaling

Een veel voorkomende situatie is dat een ruimtetuig van baan moet veranderen. Hierbij gaat de gedetailleerde uitleg van deze beweging verder dan deze tekst. Maar toch is het hier gepast de eigenschappen van simpele, impulsieve bewegingen bij Keplerbanen te overwegen.

Als een enkelvoudige, impulsieve beweging wordt uitgevoerd, moeten de begin- en eindbaan elkaar snijden, dus een enkelvoudige beweging kan alleen een ruimtetuig tussen snijdende banen verplaatsen. Om een ruimtetuig te verplaatsen tussen twee niet-snijdende banen zullen er minstens twee bewegingen nodig zijn.

Een verandering van baan A naar baan B gebeurt als er op een bepaald punt: vb. R1, de snelheidsvector onmiddellijk veranderd wordt van de waarde op A naar de waarde die er zou zijn op baan B. Het simpelste geval van baanverandering in een plat vlak is van een cirkelvormige baan naar een elliptische baan. Als op een cirkelvormige baan de snelheid vergroot, wordt de halve grote as vergroot.

Een Hofmann – verandering maakt gebruik van zulke bewegingen om te veranderen van een cirkelvormige baan naar een grotere coplanaire baan, zoals op figuur 4.9 . Het is de verandering met twee bewegingen met minimum – energie en ze is optimaal als $r_2/r_1 < 11,8$. De vergroting van de snelheid die nodig is voor de twee bewegingen kan benaderd worden door ( vis – viva ) integralen te gebruiken. De eerste beweging heeft een vergroting van de circulaire snelheid ( $\sqrt{\mu / r_1}$ ) $v_p$ nodig in het pericentrum van de ellips van de verandering,

waar: $V_p^{\,2} = 2\mu\left(1/r_1 + 1/(r_1 + r_2)\right)$

Dit vertegenwoordigt een snelheidsverandering, gegeven in:

$$\Delta V = \sqrt{\mu / r_1}\left(\sqrt{\frac{2.r_2}{r_1 + r_2}} - 1\right)$$

Een tweede dergelijke beweging zal het ruimtetuig in een circulaire baan brengen in het apocentrum met een straal gelijk aan $r_2$. De benodigde snelheidsverandering is:

$$\Delta V = \sqrt{\mu / r_2}\left(1 - \sqrt{\frac{2.r_1}{r_1 + r_2}}\right)$$

Tabel 4.2 levert details van Hohmannveranderingen in het zonnestelsel. In deze tabel is de tijd van de verandering simpelweg de helft van de periode van de elliptische baan. De snelheidsvertraging die aangeduid zijn, zijn van een aangenomen 185 km hoge circulaire baan om de aarde. Rotatiebewegingen van vliegtuigen en rotatie van de lijn van de apsis kunnen soortgelijk worden behandeld als de bovenstaande analyse. Dit kan een weerslag hebben op het brandstofgebruik. Als er bijgevolg geen snelheidsverandering is, dan is de snelheidsvermindering die nodig is evenredig met de snelheid op het moment van de beweging en deze verandering is volbracht wanneer de snelheid op zijn minimum is.

Dit punt zal later ontwikkeld worden in hoofdstuk 5, waar aspecten van zendingsplanning en analyses beschouwd worden.

## *Besluit*

We hebben als leerlingen van het laatste jaar een geïntegreerde proef samengesteld. Als onderwerp kozen we voor baanbeschrijvingen.
Eerst hebben we de wiskundige kegelsneden besproken (ellips, hyperbool en parabool) omdat alle lichamen in de ruimte zich volgens kegelsneden voortbewegen.
Daarna hebben we de wetten van Kepler besproken, die ons meer informatie gaven over het bewegen van een lichaam in de ruimte. (vb.: in gelijke tijden beschrijft de voerstraal gelijke oppervlakten)
We zijn ook tot het besluit gekomen dat de totale energie op elk moment gelijk blijft.
We hebben ook de formules opgesteld voor de elliptische, circulaire en parabolische snelheden en daarbij aansluitend ook de ontsnappingssnelheid. Daarna hebben we deze formules toegepast op een voorbeeld.
Tot nu toe hadden we nog maar de invloed van twee lichamen in rekening gebracht dus hebben we het drie-lichamenprobleem ook bestudeerd.
Vervolgens hebben we het algemeen principe voor het berekenen van de baan uitgelegd, dit toegepast op een voorbeeld en iets gezegd over de nauwkeurigheid van de resultaten.
We hebben de mogelijke banen besproken (de directe- Hohmann- en indirecte banen) waarop volgend de baancorrectie die moet toegepast worden na de lancering.
Tenslotte werd dit geheel in een computerprogramma gegoten. Dit programma kan nu a.d.h.v. bepaalde parameters de baan van planeten en objecten berekenen.

## *Logboek*

Frank

| Nr. | Datum | Titel | Tijd |
|---|---|---|---|
| 1 | '99/09/08 | Opzoekwerk bibliotheek | 1 uur |
| 2 | '99/09/10 | Boeken lezen over baanberekening | 2 uur |
| 3 | '99/09/11 | Boeken lezen over baanberekening | 4 uur |
| 4 | '99/09/15 | Boeken lezen over baanberekening | 3 uur |
| 5 | '99/09/19 | Boeken lezen over baanberekening | 2 uur |
| 6 | '99/09/20 | Lezen + begrijpen over kegelsneden | 1 uur |
| 7 | '99/09/22 | Lezen + begrijpen over kegelsneden | 5 uur |
| 8 | '99/09/25 | Lezen + begrijpen over kegelsneden | 2 uur |
| 9 | '99/09/27 | Intikken over kegelsneden | 2 uur |
| 10 | '99/09/29 | Intikken over kegelsneden | 6 uur |
| 11 | '99/09/30 | Intikken over kegelsneden | 1 uur |
| 12 | '99/10/01 | Lezen + intikken Appendix | 3 uur |
| 13 | '99/10/02 | Maken van bijhorende figuren | 4 uur |
| 14 | '99/10/06 | Maken van bijhorende figuren | 4 uur |
| 15 | '00/01/12 | Lezen over wetten van Kepler | 1 uur |
| 16 | '00/01/13 | Lezen over 2$^{de}$ wet van Kepler | 2 uur |
| 17 | '00/01/19 | Lezen over 2$^{de}$ wet van Kepler | 1 uur |
| 18 | '00/01/22 | Lezen over 2$^{de}$ wet van Kepler | 3 uur |
| 19 | '00/01/13 | Intikken van 2$^{de}$ wet van Kepler | 2 uur |
| 20 | '00/01/25 | Intikken van 2$^{de}$ wet van Kepler | 2 uur |
| 21 | '00/02/02 | Intikken van 2$^{de}$ wet van Kepler | 3 uur |
| 22 | '00/02/16 | Intikken van 2$^{de}$ wet van Kepler | 1 uur |
| 23 | '00/02/19 | Lezen over 1$^{ste}$ wet van Kepler | 4 uur |
| 24 | '00/02/20 | Lezen over 1$^{ste}$ wet van Kepler | 2 uur |
| 25 | '00/02/16 | Intikken van 1$^{ste}$ wet van Kepler | 2 uur |
| 26 | '00/02/17 | Intikken van 1$^{ste}$ wet van Kepler | 1 uur |
| 27 | '00/02/19 | Intikken van 1$^{ste}$ wet van Kepler | 3 uur |
| 28 | '00/02/22 | Lezen over 3$^{de}$ wet van Kepler | 1 uur |
| 29 | '00/02/25 | Intikken van 3$^{de}$ wet van Kepler | 1 uur |
| 30 | '00/02/27 | Lezen over energiebalans bij elliptische bewegingen | 3 uur |
| 31 | '00/03/01 | Lezen over energiebalans bij elliptische bewegingen | 2 uur |
| 32 | '00/03/02 | Lezen over energiebalans bij elliptische bewegingen | 2 uur |
| 33 | '00/03/12 | Intikken over energiebalans bij elliptische bewegingen | 5 uur |
| 34 | '00/03/13 | Intikken over energiebalans bij elliptische bewegingen | 1 uur |
| 35 | '00/03/13 | Lezen over snelheden bij de elliptische beweging | 3 uur |
| 36 | '00/03/13 | Intikken snelheden bij de elliptische beweging | 2 uur |
| 37 | '00/04/12 | Lezen De Circulaire en de parabolische snelheid | 3 uur |
| 38 | '00/04/15 | Lezen De Circulaire en de parabolische snelheid | 4 uur |
| 39 | '00/04/16 | Lezen De Circulaire en de parabolische snelheid | 1 uur |
| 40 | '00/04/19 | Intikken De Circulaire en de parabolische snelheid | 2 uur |
| 41 | '00/04/20 | Intikken De Circulaire en de parabolische snelheid | 1 uur |
| 42 | '00/04/21 | Intikken De Circulaire en de parabolische snelheid | 3 uur |
| 43 | '00/04/22 | Intikken De Circulaire en de parabolische snelheid | 4 uur |

| | | | |
|---|---|---|---|
| 44 | '00/04/29 | Lezen berekenen van de baan | 3 uur |
| 45 | '00/05/01 | Lezen berekenen van de baan | 2 uur |
| 46 | '00/05/02 | Intikken berekenen van de baan | 2 uur |
| 47 | '00/05/03 | Intikken berekenen van de baan | 4 uur |
| 48 | '00/05/05 | Lezen over het Drie lichamen probleem | 2 uur |
| 49 | '00/05/06 | Lezen over het Drie lichamen probleem | 4 uur |
| 50 | '00/05/06 | Intikken van het Drie lichamen probleem | 2 uur |
| 51 | '00/05/07 | Intikken van het Drie lichamen probleem | 3 uur |
| 52 | '00/05/10 | Intikken van het logboek | 1 uur |
| 53 | '00/05/23 | Samenvoegen van het volledige eindwerk | 2 uur |
| 54 | '00/05/24 | Samenvoegen van het volledige eindwerk | 2 uur |

Leander

| Nr. | Datum | Titel | Tijd |
|---|---|---|---|
| 1 | 10/01/00 | Klad schrijven van voorwoord en inleiding | 1 uur |
| 2 | 17/01/00 | Intypen inleiding + voorwoord | 1 uur |
| 3 | 10/02/00 | Vertalen van de Engelse tekst | 1 uur |
| 4 | 17/02/00 | Verder vertaling Engels | 1 uur |
| 5 | 24/02/00 | Verder vertaling Engels | 1 uur |
| 6 | 26/02/00 | Intypen vertaling Engels | 1 uur |
| 7 | 01/03/00 | Inlezen van de mogelijke banen | 2 uur |
| 8 | 04/03/00 | Verder inlezen mogelijke banen | 2 uur |
| 9 | 08/03/00 | Intypen van mogelijke banen | 2 uur |
| 10 | 09/03/00 | Verder intypen van mogelijke banen | 1 uur |
| 11 | 18/03/00 | Verbeteren van de mogelijke banen | 3 uur |
| 12 | 08/04/00 | Inlezen van de baancorrectie | 2 uur |
| 13 | 09/04/00 | Verder inlezen baancorrectie | 1 uur |
| 14 | 12/04/00 | Intypen baancorrectie | 2 uur |
| 15 | 19/04/00 | Verder intypen baancorrectie | 2 uur |
| 16 | 29/04/00 | Verbeteren baancorrectie | 1 uur |
| 17 | 03/05/00 | Intypen van een voorbeeld van baanberekening | 2 uur |
| 18 | 10/05/00 | Intypen van een voorbeeld van snelheid | 2 uur |
| 19 | 22/05/00 | Intypen logboek | 1 uur |

Stefaan

| Nr. | Datum | Titel | Tijd |
|---|---|---|---|
| 1 | 25-09-1999 | Onderzoek informatie baanbeschrijving | 4 uur |
| 2 | 02-10-1999 | Onderzoek i.v.m. Visual Basic | 6 uur |
| 3 | 16-10-1999 | Onderzoek i.v.m. Visual Basic | 3 uur |
| 4 | 30-10-1999 | Schrijven basis van programma | 5 uur |
| 5 | 05-11-1999 | Schrijven basis van programma | 4 uur |
| 6 | 13-11-1999 | Schrijven van programma | 4 uur |
| 7 | 20-11-1999 | Uitleg over Visual Basic | 5 uur |
| 8 | 29-12-1999 | Schrijven van programma | 4 uur |
| 9 | 04-01-2000 | Schrijven van programma | 4 uur |

| | | | |
|---|---|---|---|
| 10 | 06-01-2000 | Schrijven van programma | 5 uur |
| 11 | 08-01-2000 | Schrijven van programma | 3 uur |
| 12 | 15-01-2000 | Schrijven van programma | 4 uur |
| 13 | 22-01-2000 | Schrijven van programma | 4 uur |
| 14 | 23-01-2000 | Uitleg over Visual Basic | 4 uur |
| 15 | 05-02-2000 | Schrijven van programma | 3 uur |
| 16 | 06-02-2000 | Schrijven van programma | 4 uur |
| 17 | 19-02-2000 | Onderzoek wetten i.v.m. baanbeschrijving | 3 uur |
| 18 | 20-02-2000 | Schrijven van programma | 5 uur |
| 19 | 26-02-2000 | Schrijven van programma | 4 uur |
| 20 | 27-02-2000 | Programma verwerken in eindwerk | 3 uur |
| 21 | 04-03-2000 | Schrijven van programma | 4 uur |
| 22 | 08-03-2000 | Maken control voor grafieken | 6 uur |
| 23 | 09-03-2000 | Maken control voor grafieken | 5 uur |
| 24 | 12-03-2000 | Schrijven van programma | 4 uur |
| 25 | 18-03-2000 | Programma verwerken in eindwerk | 3 uur |
| 26 | 12-04-2000 | Schrijven van programma | 4 uur |
| 27 | 18-04-2000 | Schrijven van programma | 3 uur |
| 28 | 29-04-2000 | Schrijven van programma | 4 uur |
| 29 | 30-04-2000 | Programma verwerken in eindwerk | 5 uur |
| 30 | 06-05-2000 | Programma verwerken in eindwerk | 4 uur |
| 31 | 07-05-2000 | Programma schrijven en verwerken in eindwerk | 3 uur |
| 32 | 10-02-2000 | Onderzoek wetten i.v.m. baanbeschrijving | 3 uur |
| 33 | 14-05-2000 | Programma schrijven en verwerken in eindwerk | 3 uur |
| 34 | 15-05-2000 | Programma schrijven en verwerken in eindwerk | 4 uur |
| 35 | 20-05-2000 | Afwerken programma | 5 uur |
| 36 | 21-05-2000 | Afwerken programma en verwerken in eindwerk | 7 uur |
| 37 | 22-05-2000 | Samenvoegen en afwerken einwerk | 3 uur |
| 38 | 07-05-2000 | Afwerken programma en verwerken in eindwerk | 7 uur |

## *Bibliografie*


Natuurkunde: 6 periodieke verschijnselen (5de druk)
Uitgeverij: "De Garve" Brugge

Eindwerk Arens Nick en Debever Pieter
Klas: 614 IW  schooljaar: 1998-'99

Spacecraft Systems Engineering
By P.W.Fortescue and J.P.W.Stork

Visual Basic
By Microsoft

Beginselen van de astronautiek
Geleend aan Dhr. Verhaeghe