Module 10 Plaatjes in drie dimensies

Onderwerp	Driedimensionale plots.
Voorkennis	Module 9.
Expressies	`plot3d, spacecurve, contourplot, gradplot, cylinderplot`
Bibliotheken	`plots`
Zie ook	Module 16.

10.1 Grafieken

`plot3d`

We kunnen grafieken van functies van twee variabelen zichtbaar maken. Dit gaat met het commando `plot3d`. We kunnen natuurlijk niet altijd de hele grafiek van een functie tekenen. Daarom moeten we opgeven tot welk (rechthoekig) domein we ons willen beperken. Stel dat f gedefinieerd is op $V \subset \mathbb{R}^2$, $(x, y) \mapsto f(x, y)$. Dan moeten we de rechthoek opgeven waarbinnen de x- en y-waarden liggen waarvoor we de grafiek willen tekenen. Dus: $(x, y) \in A = [a_1, b_1] \times [a_2, b_2]$ $(A \subset V)$.

Voorbeeldopgave

Maak een tekening van de grafiek van de functie

$$f(x, y) = y \sin(y) - x^2.$$

Neem $x \in [-2, 2]$ en $y \in [-\pi, \pi]$.

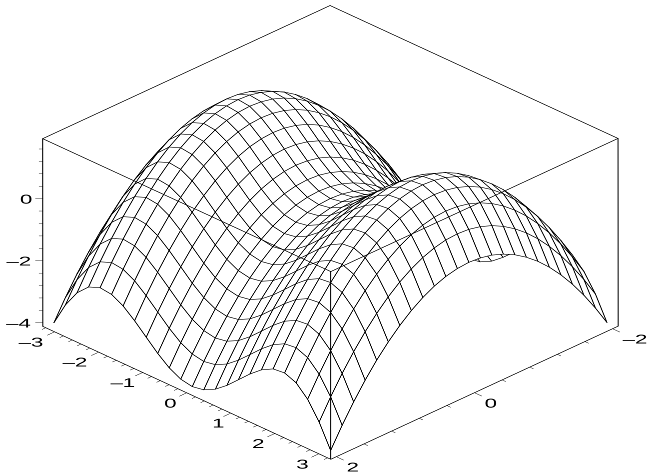
Voorbeeldsessie

```
>  plot3d( y*sin(y)-x^2, x=-2..2, y=-Pi..Pi,
           style=hidden, color=black, axes=boxed,
           tickmarks=[2,6,3], labels=["","",""] );
```

(Zie figuur 5)

Toelichting

De in de eerste regel van `plot3d` opgegeven parameters zijn *verplicht*, de overige argumenten zijn opties en kunnen ook worden weggelaten. Veel van de in `plot3d` mogelijke opties komen op een voor de hand liggende manier overeen met die van het (2D-)`plot`-commando (zie §9.4). De andere worden in §10.5 uitgelegd. ◇



FIGUUR 5. Zie de voorbeeldsessie op blz. 135

Het domein van de te tekenen grafiek hoeft niet per se een rechthoek te zijn. In het bovenstaande voorbeeld hadden we bijvoorbeeld voor de *ranges* voor x en y kunnen nemen

```
x=-1..2, y=-Pi..Pi/2*x
```

om de grafiek boven de driehoek met hoekpunten $(-2,-\pi)$, $(2,-\pi)$ en $(2,\pi)$ te krijgen. In deze vorm zijn de mogelijkheden echter beperkt; zie §10.3 voor een flexibeler aanpak.

Overigens is het net als bij tweedimensionale afbeeldingen mogelijk om meer dan één grafiek in een plaatje te krijgen door de te plotten expressies in een lijst of verzameling te plaatsen.

10.2 Ruimtekrommen

Ruimtekrommen kunnen in Maple getekend worden met de procedure
`spacecurve` `spacecurve` uit de `plots`-bibliotheek.
Stel $k : t \mapsto (x(t), y(t), z(t))$, $t \in [a, b]$ is een ruimtekromme, dan krijgen we een plaatje van k via

```
spacecurve( [x(t),y(t),z(t)], t=a..b );
```

Let op dat de opdracht voor het tekenen van een *ruimte*kromme op een heel andere manier gaat dan de opdracht voor het tekenen van een *vlakke* kromme (vergelijk §9.2).

Voorbeeldopgave

Teken de kromme gegeven door $t \mapsto (\cos(t)\cos(2t),\ \sin(2t)\cos(t),\ t)$, met $t \in [0, 2\pi]$.

Voorbeeldsessie

```
> with(plots):
> spacecurve( [ cos(t)*cos(2*t), cos(t)*sin(2*t), t ],
  t=0..2*Pi, view=[-1..1,-1..1,0..2*Pi], axes=boxed,
  color=black, tickmarks=[5,5,3], orientation=[60,60] );
```

(resultaat niet getoond).

Toelichting

De in de eerste regel van `spacecurve` opgegeven parameters zijn *verplicht*, de overige (met de gelijktekens) kunnen ook worden weggelaten; het zijn opties die in §10.5 worden uitgelegd. ◇

`display` Met behulp van de procedure `display` (uit de `plots`-bibliotheek) kunnen we bijvoorbeeld ook een grafiek en een kromme in één plaatje tekenen.

Voorbeeldopgave

Teken in de grafiek van figuur 5 de kromme waarvan (x, y) op de lijn door de hoekpunten van de rechthoek ligt.

Voorbeeldsessie

```
> with(plots):
> f := (x,y) -> y*sin(y) - x^2;
```

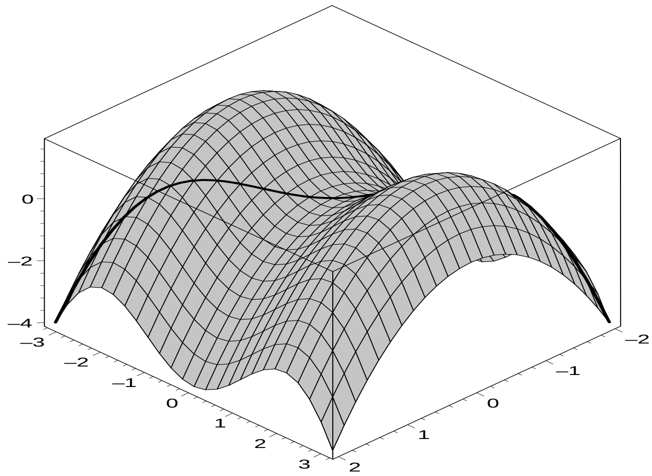
$$f := (x, y) \to y\sin(y) - x^2$$

```
> opp := plot3d( f(x,y), x=-2..2, y=-Pi..Pi, shading=zgreyscale ):
> kromme := spacecurve( [x,-Pi/2*x,f(x,-Pi/2*x)], x=-2..2,
         color=black, thickness=3 ):
> display( {opp,kromme}, tickmarks=[4,6,3],
         labels=["","",""], axes=boxed );
```

(zie figuur 6)

Toelichting

Bij het `plot3d`-commando geven we de opties mee die bij het te tekenen oppervlak horen, en bij `spacecurve` de opties voor de kromme.



FIGUUR 6. Zie de voorbeeldsessie op blz. 137

Bij het `display`-commando kunnen tenslotte de opties voor het hele plaatje worden gegeven. ◇

10.3 Geparametriseerde oppervlakken

We kunnen ook geparametriseerde oppervlakken tekenen:

Voorbeeldopgave

Maak een tekening van het oppervlak S gegeven door

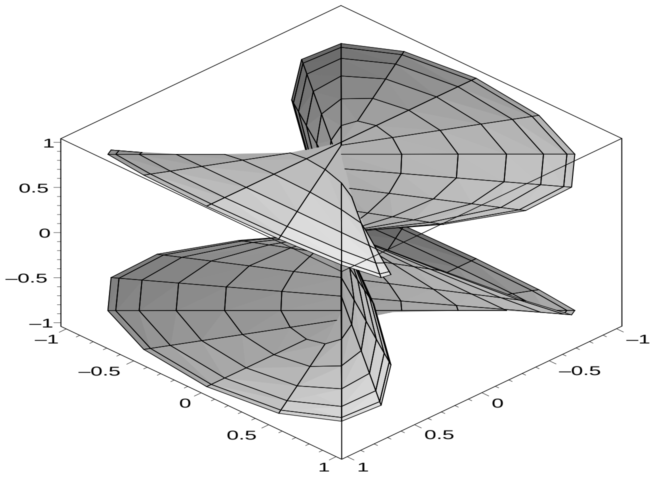
$$\begin{aligned} x(s,t) &= \cos(t)\sin(s) \\ y(s,t) &= \sin(2t)\sin(s) \\ z(s,t) &= \cos(t). \end{aligned}$$

Neem $0 \leq s \leq 2\pi$ en $0 \leq t \leq 2\pi$.

Voorbeeldsessie

```
> plot3d( [ cos(t)*sin(s), sin(2*t)*sin(s), cos(t) ],
    s=0..2*Pi, t=0..2*Pi, axes=boxed );
```

(zie figuur 7)



FIGUUR 7. Zie de voorbeeldsessie op blz. 138

Toelichting

Merk op dat de grenzen voor de *parameters* (s en t) moeten worden opgegeven. Maple zorgt er dan zelf voor dat we alle bijbehorende punten op het oppervlak te zien krijgen. Als we slechts een *gedeelte* van het oppervlak willen zien, moeten we gewenste x-, y- en z-coördinaten met een `view`-optie in `plot3d` aangeven, zie §10.5. ◇

Grafieken op een niet-rechthoekig gebied. Plotten als geparametriseerd oppervlak kan ook worden gebruikt voor de grafiek van een functie als het domein geen rechthoek is.

Voorbeeldopgave

Maak een tekening van de grafiek van de functie $f(x, y) = y\sin(y) - x^2$, voor (x, y) in de cirkel met straal 3 en de oorsprong als middelpunt.

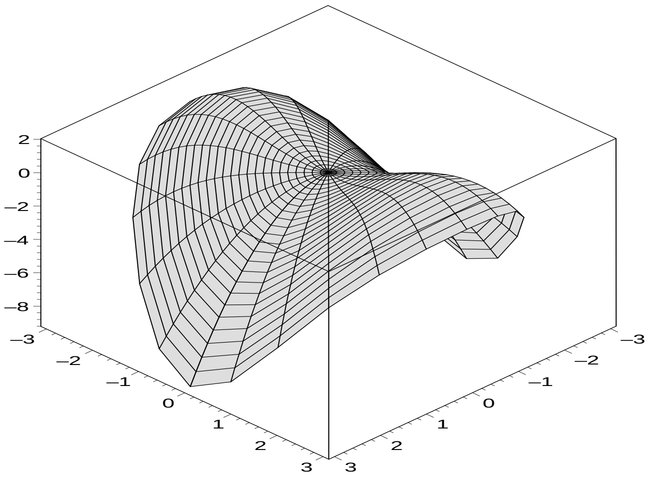
Voorbeeldsessie

```
>  f := (x,y) -> y*sin(y) - x^2;
```

$$f := (x, y) \to y\sin(y) - x^2$$

```
>  plot3d( [r*cos(t), r*sin(t), f(r*cos(t),r*sin(t))],
      r=0..3, t=0..2*Pi, axes=boxed );
```

(zie figuur 8)



FIGUUR 8. Zie de voorbeeldsessie op blz. 139

Toelichting

Dat (x, y) in de gewenste cirkel ligt kunnen we het gemakkelijkst vastleggen door middel van poolcoördinaten: $x = r\cos t, y = r\sin t$, met $0 \le r \le 3$ en $0 \le t \le 2\pi$. ◇

Omwentelingsoppervlakken. Een bijzondere vorm van een geparametriseerd oppervlak is het oppervlak dat ontstaat door een vlakke kromme te wentelen rond een as die in hetzelfde vlak als de kromme ligt. Als de as de z-as is, kunnen we de kromme beschrijven als functie van z. Met het commando `cylinderplot` (in de `plots`-bibliotheek) is zo'n omwentelingslichaam gemakkelijk te tekenen:

cylinderplot

```
cylinderplot( f(z), theta=0..2*Pi, z=a..b );
```

We geven een voorbeeld.

Voorbeeldopgave

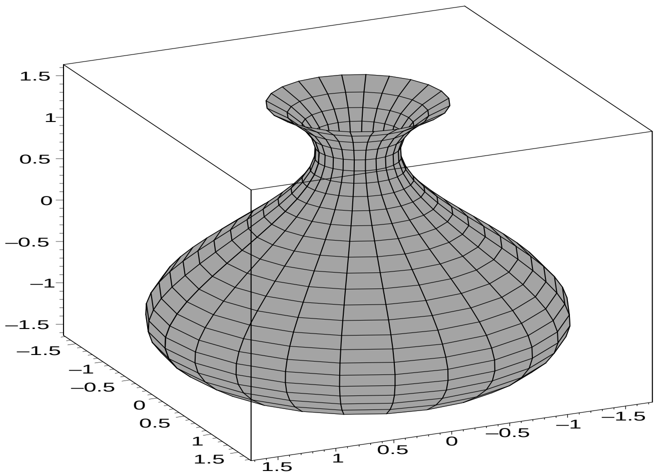
Teken het omwentelingsoppervlak dat ontstaat door wenteling van de kromme $f(z) = \frac{1}{3}x^3 - x + 1$ rond de z-as. Neem $-\frac{\pi}{2} \le z \le \frac{\pi}{2}$.

Voorbeeldsessie

```
> restart:  with(plots):
> f := x -> x^3/3 - x + 1:
> cylinderplot( f(z), theta=0..2*Pi, z=-Pi/2..Pi/2, axes=boxed );
```

Dit geeft hetzelfde plaatje als

```
>  plot3d( [f(z)*cos(theta), f(z)*sin(theta), z], theta=0..2*Pi,
   z=-Pi/2..Pi/2, axes=boxed );
```



Toelichting

Bij `cylinderplot` is de *volgorde* waarin het bereik van de parameters wordt opgegeven van belang: éérst de hoek, en daarna de z. Kijk maar eens wat er gebeurt als u de volgorde verwisselt.
Met een 'gewone' parametrisering is het trouwens bijna net zo gemakkelijk. Neem $x(\theta, z) = f(z)\cos\theta$, $y(\theta, z) = f(z)\sin\theta$, $z(\theta, z) = z$. ◇

10.4 Niveaukrommen, gradientveld

Niveaukrommen. Om *niveaukrommen* van een functie van twee variabelen te tekenen zijn er verschillende mogelijkheden. In de eerste plaats kan men natuurlijk voor elke gewenste c een `implicitplot` van $f(x, y) = c$ maken en deze met `display` laten tekenen, zie Module 9. In Module 27 zullen we zien hoe dat ook met één commando gedaan kan worden.

`contourplot` De procedure `contourplot` dat in de `plots`-bibliotheek aanwezig is, is speciaal hiervoor bedoeld. Het commando

```
contourplot( x^2 - y*sin(y), x=-2..2, y=-Pi..Pi );
```

tekent een aantal niveaukrommen van de functie $f(x, y) = x^2 - y\sin y$. Als men meer of minder niveaukrommen getekend wil hebben, kan de
`contours` optie `contours=`n worden meegegeven; hierin is n het aantal niveau-

krommen dat getekend moet worden. Met `contours=[-1,0,1,2,3]` worden de niveaukrommen $f(x, y) = c$ getekend voor $c = -1, 0, 1, 2, 3$.

> **!** Het resultaat van een `contourplot` is een *twee*dimensionaal plaatje, en kan daarom niet door middel van een `display`-commando met bijvoorbeeld een `plot3d` worden gecombineerd.

Via de `style`-optie bij `plot3d` kan men trouwens ook contourinformatie in het plaatje verwerken.

`gradplot`

Gradiëntveld. Verder is er de functie `gradplot`, waarmee men van een functie $f(x, y)$ het vectorveld grad $f(x, y)$ kan tekenen. Het verdient aanbeveling om de optie `arrows = SLIM` te gebruiken als in hetzelfde plaatje ook nog iets anders getekend moet worden (bijvoorbeeld niveaukrommen).

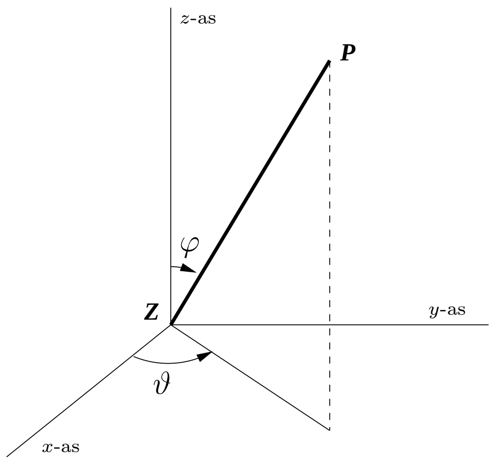
10.5 Opties

Veel opties voor het `plot3d`-commando zijn analoog aan die voor `plot`. We behandelen hier de opties die specifiek zijn voor ‘driedimensionale’ plaatjes.
Aan de tot nu toe getoonde voorbeelden is te zien dat de plaatjes uit een groot aantal (vlakke) polygoontjes bestaan. Deze polygoontjes vormen een benadering van het getekende oppervlak. In principe wordt alleen dat deel van het oppervlak getekend dat we ook daadwerkelijk kunnen zien. We hebben twee mogelijkheden om toch andere delen van het oppervlak te zien te krijgen: verandering van gezichtspunt en het doorzichtig maken van de polygoontjes. Bovendien zijn er nog enkele andere manieren om de presentatie van het plaatje aan te passen.
Dit gebeurt door een *optie* mee te geven aan het commando `plot3d` (zoals in de voorbeelden ook al is gebeurd) of aan het commando `display`. Hieronder zullen we enkele opties kort bespreken.

`orientation`

Verandering van gezichtspunt. We kunnen het oppervlak vanuit verschillende hoeken bekijken. Voor het gezichtspunt is de optie `orientation=[`ϑ, φ`]` van belang. De parameters ϑ en φ geven (in graden) de bolcoördinaten van een punt P op de eenheidsbol met middelpunt ongeveer in het zwaartepunt Z van de grafiek (respectievelijk het oppervlak, de ruimtekromme). Het oog bevindt zich op een

rechte door de punten Z en P. De betekenis van de getallen ϑ en φ wordt duidelijk in figuur 9.[30]
Dus als $\varphi = 0$ kijken we boven op de grafiek, als $\vartheta = 0$ kijken we langs de x-as enzovoort.



FIGUUR 9. De hoekparameters ϑ en φ in de `orientation`-optie

Als u méér grafieken in een plaatje combineert moet de `orientation`-optie in het `display`-commando gegeven worden. Zie het eerstvolgende voorbeeld.

Doorzichtig maken van de polygoontjes We kunnen er ook voor zorgen dat de polygoontjes als het ware doorzichtig worden. Dan worden alleen de randen getekend, en we kunnen de achterliggende polygoontjes ook nog zien. Dit is soms handig wanneer we diverse grafieken in één figuur willen tekenen en willen zien hoe het gebied eruitziet dat door deze grafieken wordt ingesloten. We moeten dan
`WIREFRAME` de optie `style = WIREFRAME` gebruiken.
Het ondoorzichtig laten zijn van de polygoontjes kan middels de optie
`HIDDEN` `style = HIDDEN`. In het algemeen zal het niet nodig zijn deze optie op te geven omdat dit op de meeste systemen de standaardinstelling is.
`PATCH` De optie `style = PATCH` zorgt ervoor dat de polygoontjes worden opgevuld met een kleur die afhankelijk is van de coördinaten van het betreffende punt op het oppervlak. Varianten hiervan zijn `PATCHNOGRID` en `PATCHCONTOUR`.

[30] Merk op dat de rollen van ϑ en φ hier precies zijn verwisseld ten opzichte van de gebruikelijke manier waarop bolcoördinaten worden genoteerd.

`transparency` `glossiness` Met de optie `transparency=t` kan de grafiek ‘doorschijnend’ worden gemaakt. Kies voor t een getal tussen 0 (ondoorzichtig) en 1 (geheel doorzichtig). De optie `glossiness`, met ook weer een waarde tussen 0 en 1, kan ook een fraai effect opleveren.

`CONTOUR` Ten slotte is er de `style`-optie `style = CONTOUR`. Hiermee wordt precies hetzelfde bereikt als met `contourplot3d`.

De optie `scaling`. Voor de optie `scaling` zijn er twee mogelijkheden, namelijk

`scaling` `scaling=CONSTRAINED` of `scaling=UNCONSTRAINED`.

Wanneer `scaling = CONSTRAINED` wordt gegeven, worden de plaatjes in de normale verhoudingen weergegeven. In het andere geval worden de plaatjes zo opgerekt, dat ze het hele scherm vullen. Daarbij kunnen de verhoudingen tussen lengte, breedte en hoogte weleens veranderen!

`shading` **De optie `shading = Z`.** Bij deze optie krijgen alle punten met gelijke z-coördinaat dezelfde kleur. Dit is ook interessant als optie bij `contourplot3d`.

`view` **De optie `view`.** Met de optie `view` kan worden aangegeven welk gedeelte van het plaatje we willen bekijken. Wanneer we opgeven

```
view = [c1..d1,c2..d2,c3..d3]
```

dan wordt het gedeelte van het plaatje getoond dat ligt in het blok $[c_1, d_1] \times [c_2, d_2] \times [c_3, d_3]$. Als u alleen de z-waarde wilt beperken, dan kan dat ook met `view = c3..d3` (dus zonder vierkante haken).

`grid` **De optie `grid`.** Met de optie `grid` kan het aantal polygoontjes worden vergroot. Standaard is `grid=[25,25]`. Door dit aantal te verhogen kunnen we soms een mooier plaatje krijgen.

Raadpleeg vooral `?plot3d,options` om meer opties te weten te komen. Bovendien is het zo dat diverse opties ook nog achteraf in te stellen zijn via het menu dat verschijnt wanneer de grafiek wordt getekend.
Een goede manier om gewend te raken aan de diverse opties is om deze in verschillende combinaties uit te proberen op de grafiek van een *eenvoudige* functie.

Voorbeeldopgave

Maak een schets van het gebied gegeven door:

$$\{(x, y, z) \mid x^2 + y^2 + z^2 \leq 1; \quad x + y + z \geq 1\}$$

Voorbeeldsessie

```
> with(plots):
```

Parametrisering van de bol $x^2 + y^2 + z^2 = 1$:

```
> X := [sin(phi)*cos(theta), sin(phi)*sin(theta), cos(phi)];
```

$$X := [\sin(\phi)\cos(\theta), \sin(\phi)\sin(\theta), \cos(\phi)]$$

```
> plot1 := plot3d( X, phi=0..Pi, theta=0..2*Pi,
  grid=[40,40], style=patchnogrid, shading=zgreyscale,
  glossiness=1, transparency=0.3 ):
```

Parametrisering van het vlak $x + y + z = 1$:

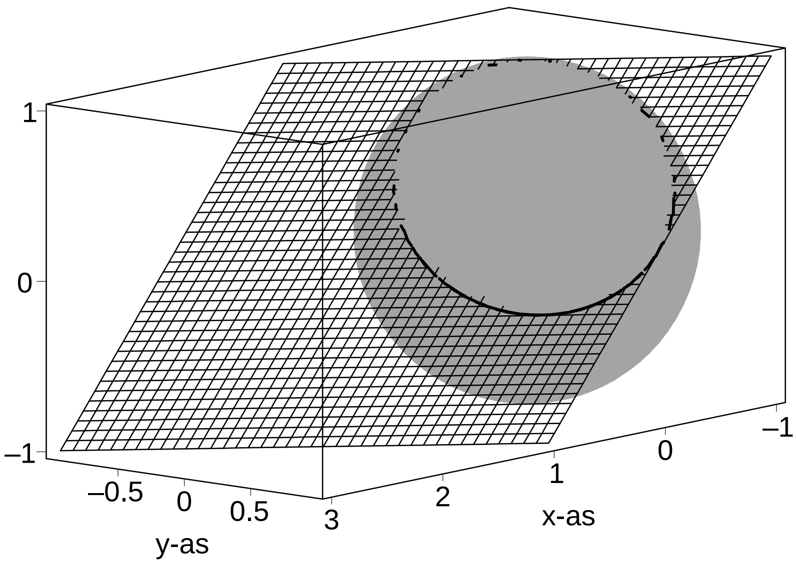
```
> Y := [1-y-z, y, z];
```

$$Y := [1 - y - z, y, z]$$

```
> plot2 := plot3d( Y, y=-1..1, z=-1..1,
  grid=[20,20], style=wireframe, color=red):
```

Doorsnijding van X en Y:

```
> plot3 := intersectplot( surface(X, phi=0..Pi, theta=0..2*Pi),
  surface(Y, y=-1..1, z=-1..1), color=blue, thickness=5 ):
> display( {plot1,plot2,plot3}, orientation=[40,80],
  axes=BOXED, scaling=CONSTRAINED,
  labels=["x-as","y-as",""],
  tickmarks=[[-1,0,1,2,3],[-1/2,0,1/2],[-1,0,1]] );
```



FIGUUR 10. Zie de voorbeeldsessie op blz. 145

Toelichting

intersectplot

`X` is de bol en `Y` is het vlak. Beide zijn als geparametriseerde oppervlakken getekend. We hebben de snijkromme apart getekend met behulp van `intersectplot`. De beide oppervlakken moeten we in dit geval als `surface` opgeven.
Het opgegeven gebied ligt ingesloten tussen het vlak en de bol. ◇

Opgave 10.1

Maak een plaatje van de grafiek van de functie

$$(x, y) \mapsto x^2 y \sin(xy)$$

voor $(x, y) \in [0, 2\pi] \times [0, 1]$.
Neem het gezichtspunt achtereenvolgens

(a) op de y-as
(b) op de x-as
(c) op de z-as
(d) op de lijn door de oorsprong en met richtingsvector $(1, 1, 1)$.

Maak ook een hoogtekaart van de functie.

Opgave 10.2

Maak een schets van de ruimtekromme

$$t \mapsto (\sin(t),\ \cos(t)\sin(t),\ \cos^2(t))$$

Kies zelf een geschikt interval voor de parameter t.

Opgave 10.3

(a) Beschouw de functie $f : \mathbb{R}^2 \to \mathbb{R}$ met

$$f(\mathbf{x}) = x_1^2 + x_2^2\,,$$

waarbij $x_1 \in [-3/4, 3/4],\ x_2 \in [-3/4, 3/4]$

(1) Teken de grafiek van f;
(2) Teken met `contourplot( )` de hoogtelijnen bij $f(\mathbf{x}) = 1/100,\ 1/10,\ 2/10,\ 3/10,\ 4/10$;
(3) Teken met `gradplot( )` het gradiëntveld van f in hetzelfde plaatje als de hoogtelijnen. Merk op dat de gradiënt overal loodrecht op de niveaulijnen staat.

(b) Dezelfde vragen voor

$$f(\mathbf{x}) = x_1^2 - x_2^2\,;$$

teken hoogtelijnen bij $f(\mathbf{x}) = 0,\ \pm 1/10,\ \pm 2/10$.

(c) Nu is

$$f(\mathbf{x}) = x_1^2 - x_2^3\,,$$

met $x_1 \in [-3/4, 3/4],\ x_2 \in [-3/4, 3/4]$.

(1) Teken de grafiek van f;

(2) Hoogtelijnen bij $f(\mathbf{x}) = 0,\ \pm 1/10,\ \pm 2/10$.
Merk op dat de hoogtelijn bij $f(\mathbf{x}) = 0$ niet door het punt $(0, 0)$ gaat. Dat is het gevolg van afrondingsfouten. U kunt dat verbeteren door de hoogtelijn bij $f(\mathbf{x}) = 0$ apart te tekenen als functie van x_1 (let erop dat x_2 ook bij negatieve waarden van x_1 moet bestaan). Een gewone `plot` en een `contourplot` kunnen met `display` in één plaatje worden getekend.

(3) Teken het gradiëntveld erbij.

Opgave 10.4

Maak een tekening van het oppervlak gegeven door

$$(t, u) \mapsto (\cos(t)(1 + \frac{1}{5}\sin(u)), \sin(t)(1 + \frac{1}{5}\sin(u)), \frac{1}{5}\sin(t)\cos(u)).$$

Opgave 10.5

Teken in één figuur de eenheidsbol en de kromme van opgave 10.2. (Kies `color=black` voor de kromme.)

Opgave 10.6

De functie $f:\ \mathbb{R}^2 \to \mathbb{R}$ wordt gegeven door

$$f(\mathbf{x}) = \sqrt[3]{x_1^2 x_2^2}\,.$$

(a) Teken de niveaukrommen $f(\mathbf{x}) = c$, met $c = \frac{2}{3},\ c = \frac{1}{3},\ c = \frac{1}{10}$ en $c = 0$. Neem $\mathbf{x} \in [-1, 1] \times [-1, 1]$ en kies `axes=BOXED`. Maak aparte 2D-plots van de niveaulijnen waar u niet tevreden mee bent.

(b) Schets de doorsnijdingen van de grafiek van f met het vlak $\{(x_1, x_2, x_3) \mid x_2 = x_1\}$ en met het vlak $\{(x_1, x_2, x_3) \mid x_2 = \sqrt{3}\,x_1\}$. Ga daarbij als volgt te werk. Bedenk dat de afbeelding $t \mapsto (t\cos\varphi,\ t\sin\varphi)$ voor vaste φ een parametervoorstelling is van de rechte door de oorsprong die een hoek φ met de positieve

x_1-as maakt. Neem $\varphi = \frac{\pi}{4}$ en plot de expressie $f(t\cos\varphi,\, t\sin\varphi)$. Idem met $\varphi = \frac{\pi}{3}$ in dezelfde figuur. (*Aanwijzing:* Zie ook opgave 9.2.) Het voordeel van deze omslachtige methode is dat in beide grafieken de schaal langs de t-as hetzelfde is, namelijk de afstand tot de oorsprong.

Is aan de grafieken 'te zien' of deze krommen differentieerbaar zijn in $t = 0$? Geef in uw tekening bij (a) aan waar in de tweede tekening de 't-assen' liggen.

(c) Schets het verloop van de functiewaarden als (x_1, x_2) de cirkel $x_1^2 + x_2^2 = 1$ doorloopt.
(Parametriseer de cirkel met $(x_1, x_2) = (\cos t, \sin t),\ t \in [-\pi, \pi]$; met andere woorden plot $f(\cos t, \sin t)$ op het interval $[-\pi, \pi]$.)

(d) Maak nu een 3D-plot van f. Is het gemakkelijk te raden of f in $\mathbf{0}$ differentieerbaar is?
Teken in hetzelfde plaatje de doorsnijding met de cilinder
$$x_1^2 + x_2^2 = 1\,.$$
Opmerking: Een kromme die precies op het oppervlak ligt is vaak niet overal te zien omdat het oppervlak 'ondoorzichtig' is en de kromme (door afrondingsfouten) er nét boven of er nét onder ligt. Men zou van de kromme twee versies kunnen tekenen: eentje die 0.01 *boven* het oppervlak zweeft en eentje die er 0.01 *onder* hangt: men is er dan van verzekerd dat overal één van beide te zien is.

Opgave 10.7

Een plaatje van twee snijdende cilinders.

(a) Teken een cilinder met de z-as als as en straal $\frac{1}{2}$ als geparametriseerd oppervlak:
$$x = \frac{1}{2}\cos(\phi), \quad y = \frac{1}{2}\sin(\phi), \quad z = \zeta.$$
Neem $\zeta \in [-1, 1]$.

(b) Teken in hetzelfde plaatje een cilinder met straal $\frac{1}{2}$ om de x-as.

(c) Bepaal de snijkromme van beide figuren als parametervoorstelling.
Aanwijzing: Ga uit van de vergelijkingen $x^2+y^2 = \frac{1}{4}$, $y^2+z^2 = \frac{1}{4}$ en druk y en z uit in x. De snijkromme bestaat uit vier stukken.

(d) Teken de (vier) snijkromme(n) in één figuur. (`color=black`).

(e) Teken ten slotte de beide cilinders mét de snijkrommen in één plaatje. Zorg dat de 'polygoontjes' niet te zien zijn. Neem `scaling=CONSTRAINED`.