

# Plot af parametriserede mængder

```
> restart
> with(plots) :
> with(VektorAnalyse4)
```

[div, grad, kryds, prik, rot, vop]

(1)

## 1D-kurve i 2D (planen)

Kurven er givet ved 1 parameter:  $u$ .

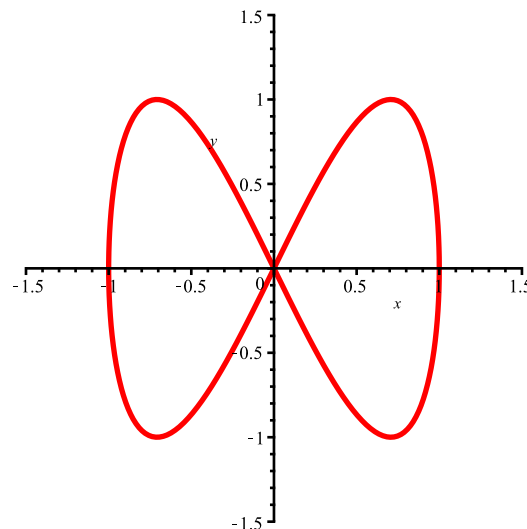
```
> r(u) := <cos(u), sin(2 u)> :
r(u)
```

$$\begin{bmatrix} \cos(u) \\ \sin(2u) \end{bmatrix}$$

(1.1)

hvor  $u \in [0, 2 \cdot \pi]$ .

```
> plot([vop(r(u)), u = 0 .. 2 * pi], labels = [x, y], view = [-1.5 .. 1.5, -1.5 .. 1.5], thickness = 2,
color = red)
```



## 2D-område i 2D (planen)

Området er givet ved 2 parametre:  $u$  og  $v$ .

```
> r(u, v) := <v*cos(u), v*sin(u)> :
r(u, v)
```

$$\begin{bmatrix} v \cos(u) \\ v \sin(u) \end{bmatrix}$$

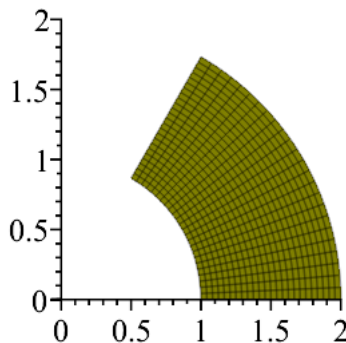
(2.1)

hvor  $u \in \left[0, \frac{\pi}{3}\right]$ ,  $v \in [1, 2]$ .

**Maple kan ikke direkte tegne et 2D parametriseret område i planen. Er meddelt til Maplesoft Support i marts 2019.**

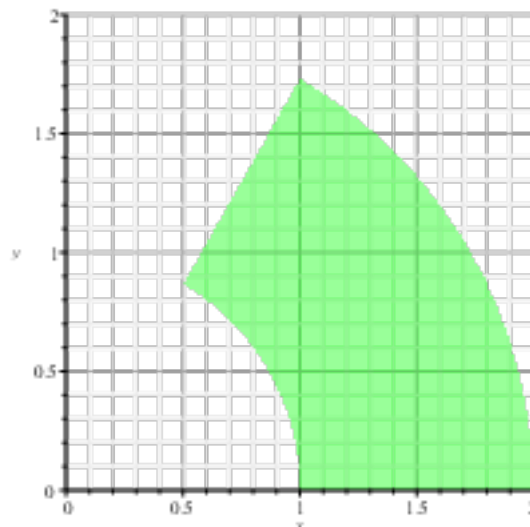
**Workaround er at tegne i 3D, og så tilpasse orienteringen med parameteren**  
 $orientation = [-90, 0]$ .

```
> plot3d(⟨vop(r(u, v)), 0⟩, u = 0 .. π/3, v = 1 .. 2, labels = [x, y, " "], axes = normal, orientation
= [-90, 0], view = [0 .. 2, 0 .. 2, -2 .. 2], color = yellow)
```



**Bedre (anbefales): man kan konvertere plottet til et ægte 2D-plot, og så f.eks. sætte tern på.**  
 NB: *transform* fra *plottools*-pakken fjerner 3. koordinaten fra 3D-plottet!

```
> plot3D := plot3d(⟨vop(r(u, v)), 0⟩, u = 0 .. π/3, v = 1 .. 2) :
plot2D := plottools[transform]((x, y, z) → [x, y])(plot3D) :
display(plot2D, color = green, gridlines, style = surface, transparency = 0.6, view = [0 .. 2, 0
.. 2], labels = [x, y])
```



**Bedre et det, at anvende Steens nye pakke "plots2D3D".  
 Se separat fil, som viser anvendelsen af pakken.**

## 1D-kurve i 3D (rummet)

Kurven er givet ved 1 parameter:  $u$ .

$$\begin{aligned} > r(u) := \langle \cos(u), \sin(2 \cdot u), \cos(3 \cdot u) \rangle : \\ & r(u) \end{aligned}$$

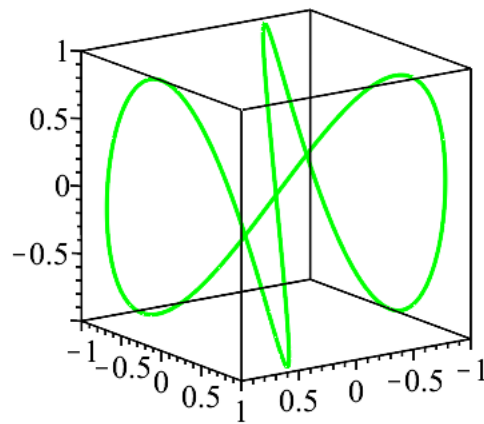
$$\begin{bmatrix} \cos(u) \\ \sin(2u) \\ \cos(3u) \end{bmatrix}$$

(3.1)

hvor  $u \in [0, 2 \cdot \pi]$

**Parameteren *numpoints* (antal punkter) gør at kurven bliver rund, ikke kantet.**

$$\begin{aligned} > \text{spacecurve}(r(u), u = 0 .. 2 \cdot \pi, \text{axes} = \text{box}, \text{labels} = [x, y, z], \text{thickness} = 3, \text{view} = [-1 .. 1, -1 .. 1, \\ & -1 .. 1], \text{color} = \text{green}, \text{numpoints} = 1000) \end{aligned}$$



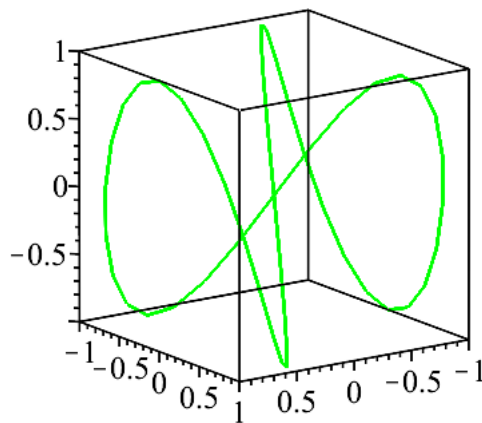
I **Maple 2018** kan man anvende *plot3d* til at tegne kurven.

**NB:** *style = line* er vigtig: ellers får kurven automatisk farven **sort!**

**NB:** Selv med parameteren *numpoints* = 10000 bliver kurven kantet.

**Anbefaling: brug *spacecurve*.**

$$\begin{aligned} > \text{plot3d}([ \text{vop}(r(u)) ], u = 0 .. 2 \cdot \pi, \text{labels} = [x, y, z], \text{view} = [-1 .. 1, -1 .. 1, -1 .. 1], \text{thickness} = 2, \\ & \text{color} = \text{green}, \text{style} = \text{line}) \end{aligned}$$



## 2D-flade i 3D (rummet)

Fladen er givet ved 2 parametre:  $u$  og  $v$ .

$$\mathbf{r}(u, v) := \langle v \cdot \cos(u), v \cdot \sin(u), v^2 \rangle :$$

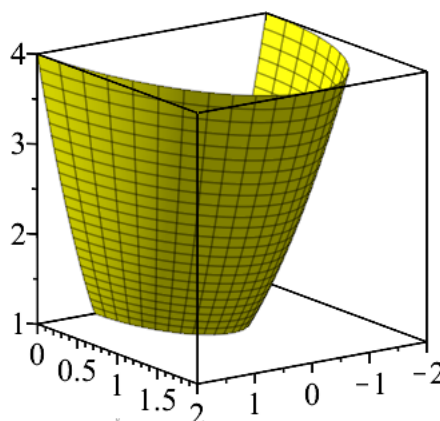
$\mathbf{r}(u, v)$

$$\begin{bmatrix} v \cos(u) \\ v \sin(u) \\ v^2 \end{bmatrix}$$

(4.1)

hvor  $u \in [0, \pi]$ ,  $v \in [1, 2]$

$\mathbf{r}(u, v)$ ,  $u = 0 \dots \pi$ ,  $v = 1 \dots 2$ ,  $axes = box$ ,  $labels = [x, y, z]$ ,  $color = yellow$ )



## 3D-område i 3D (rummet)

Området er givet ved 3 parametre:  $u$ ,  $v$  og  $w$ .

$$\begin{aligned} > r(u, v, w) := \left\langle \sin(u), \cos(u) + v, w \cdot \frac{u}{2} \right\rangle : \\ & r(u, v, w) \end{aligned}$$

$$\begin{bmatrix} \sin(u) \\ \cos(u) + v \\ \frac{w u}{2} \end{bmatrix}$$

(5.1)

hvor  $u \in [0, 2 \cdot \pi]$ ,  $v \in [0, 1]$ ,  $w \in [0, 1]$ .

**Maple kan ikke direkte tegne det rummelige område. Er meddelt til Maplesoft Support i marts 2019.**

**Man er nødt til selv at parametrisere de enkelte sideflader, plotte dem hver for sig, og så sammensætte med *display*.**

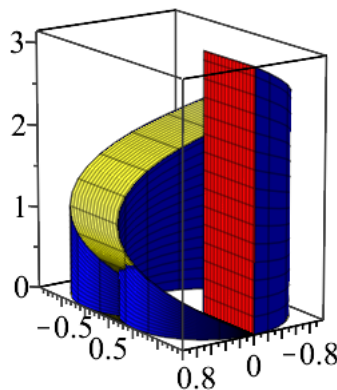
Langt lettere er det at anvende *Integrator8*-pakken:

Parameterområderne indskrives i en liste som 2. parameter i kaldet til *sideFlader* fra *Integrator8*-pakken.

**NB: Man kan ikke skifte farver på fladerne!**

**> plot1 := Integrator8[sideFlader](r, [0, 2 · π, 0, 1, 0, 1], [10, 10, 10]) :**

**> display(plot1, axes = box, labels = [x, y, z])**

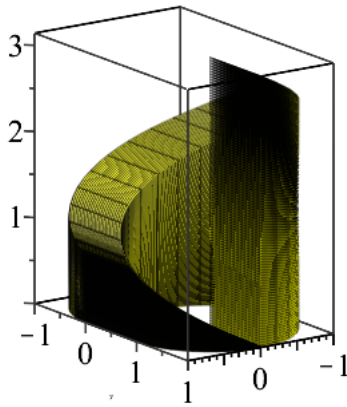


**Metode (fra Laurits Nicolaisen, s194330, MatTek1, 2019/2020):**

**Der genereres MANGE plots af tætliggende flader, som tilsammen viser den rumlige figur.**

NB: Det grafiske billede bruger meget lagerplads!

**> display(seq(plot3d( r(u, v, ω), u = 0 .. 2 · π, v = 0 .. 1, labels = [x, y, z], color = yellow, scaling = constrained), ω = 0 .. 1, 0.01))**



**Bedre et det, at anvende Steens nye pakke "plots2D3D".  
Se separat fil, som viser anvendelsen af pakken.**