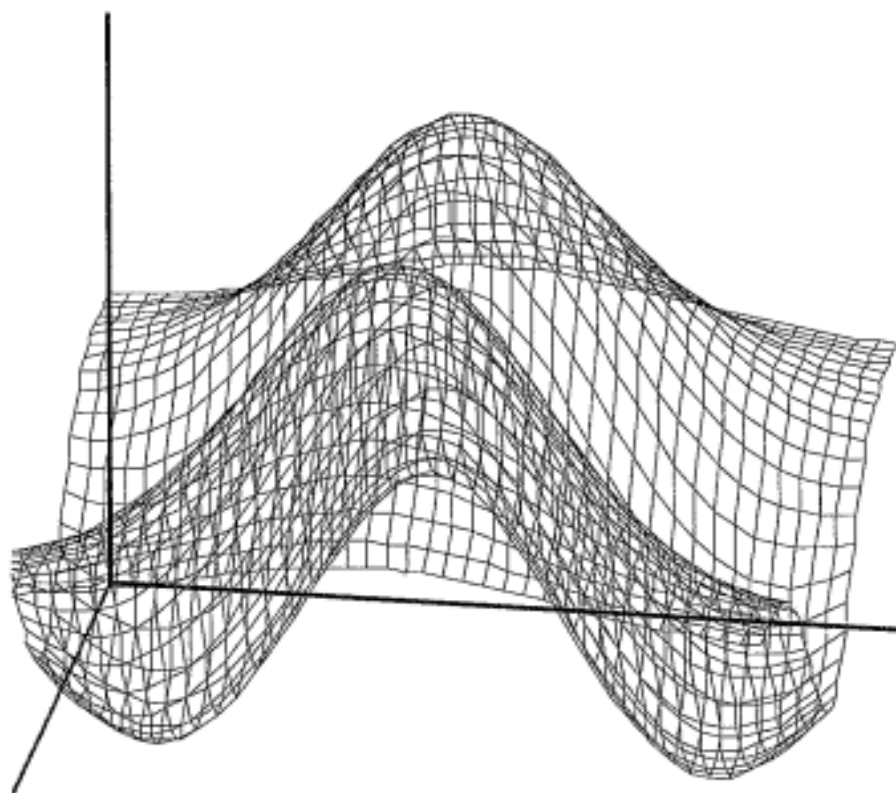


MATLAB

Pinnat

Timo Mäkelä



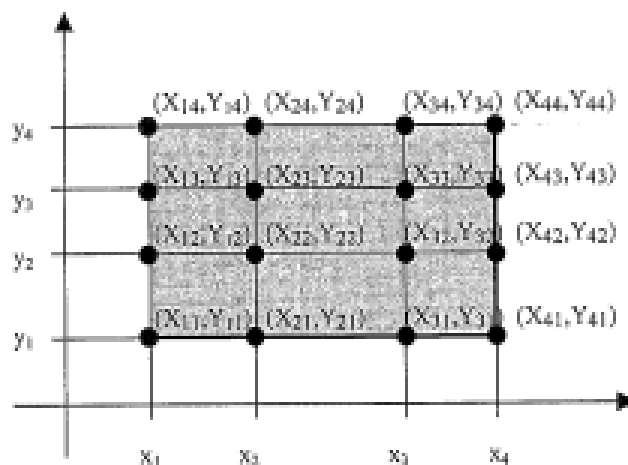
10. PINNAT

Tässä kappaleessa tarkastellaan pintojen piirtämistä 3-ulotteisessa avaruudessa.

Pintojen piirtämisessä joudutaan luomaan verkko 2 tai 3-ulotteiseen avaruuteen. Tähän voidaan käyttää seuraavia komentoja:

Komento	Toiminto
$[X,Y] = \text{meshgrid}(x,y)$	Muodostaa tasoon ruudukon. Ruudukon solmupisteet (x,y) määräytyvät vektoreista x ja y , jotka on annettava nousevassa järjestyksessä. Jos vektoreiden x ja y dimensiot ovat m ja n , niin matriisien X ja Y kertaluvut ovat $m \times n$ ja ne sisältävät solmupisteiden x - ja y -koordinaatit.
$[X,Y] = \text{meshgrid}(x)$	Sama kuin $[X,Y] = \text{meshgrid}(x,x)$
$[X,Y,Z] = \text{meshgrid}(x,y,z)$	Muodostaa verkon (ruudukon) 3-ulotteiseen avaruuteen samaan tapaan kuin tasoon.
$[X,Y,Z] = \text{cylinder}(r,n)$	Muodostaa xy -tasolla olevalle yksikön korkuiselle pyörähdyskappaleen pinnalle verkon. Pyörähdyskappaleen pyörähdysakseli on z -akseli. Vektorin r alkiot määräävät pyörähdyskappaleen säteen z -akselin suuntaisessa tasavälisessä jaossa. n määrittää verkon pisteiden lukumäärän tietyllä korkeudella ¹ . Verkon solmupisteiden koordinaatit määräytyvät matriiseista X , Y , Z .
$[X,Y,Z] = \text{sphere}(n)$	Muodostaa yksikköympyrän pinnalle n yhtä etäällä toisistaan olevien pisteiden verkon $(n+1) \times (n+1)$ -matriiseilla X , Y , Z

Seuraava kuva pyrkii havainnollistamaan komentoa $[X,Y] = \text{meshgrid}(x,y)$.



Pinnan 3-ulotteisen kuvan piirtoon on käytettävissä seuraavia komentoja:

Komento	Toiminto
$\text{surf}(X,Y,Z)$	Piirretään pinta, jonka määräävät pisteet (X_i, Y_i, Z_i)
$\text{surfc}(X,Y,Z)$	Sama kuin $\text{surf}(\dots)$, mutta piirretään myös tasa-arvokäyrästä.
$\text{surf1}(X,Y,Z,ls)$	Sama kuin $\text{surf}(\dots)$, mutta lisätään valo suuntaan ls .
$\text{surfnorm}(X,Y,Z)$	Sama kuin $\text{surf}(\dots)$, mutta piirretään pinnan normaalit.

¹ Katso tarkemmin MATLABin helpistä.

Pinnan katselusuunta voidaan määrittää komennolla

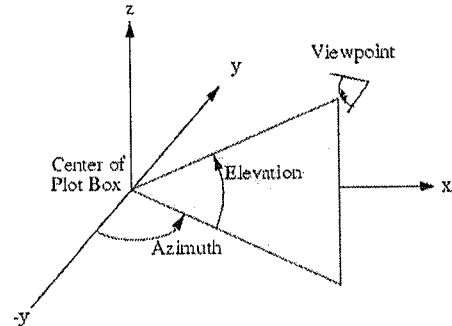
`view(θ , φ)`

tai

`view([θ , φ]),`

missä

- θ ilmoittaa katselusuunnan kiertokulman positiivisen z-akselin ympäri. Positiivinen kiertokulma mitataan asteina *vastapäivään negatiivisen y-akselin suunnasta* lähtien. Kuvassa **Azimuth**.
- φ ilmoittaa *katselusuunnan ja xy-tason välisen kulman* asteina. Positiivinen suunta xy-tason yläpuolella, negatiivinen xy-tason alapuolella. Kuvassa **Elevation**.



Komento

`view(2)`

asettaa oletusarvoisen 2-dimensioisen näkymän. Tämä on sama kuin komento `view(0,90)`.

Komento

`view(3)`

asettaa oletusarvoisen 3-dimensioisen näkymän. Tämä on sama kuin komento `view(-37.5,30)`.

Jos komennoista **cylinder** ja **sphere** jättää output-parametrin pois piirretään matriisien X, Y ja Z määräämä pinta.

Esim. Komennoilla

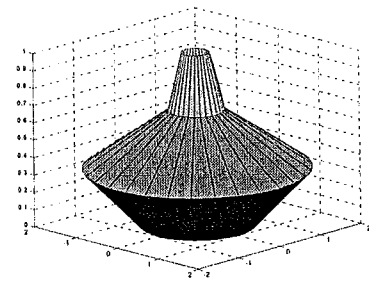
```
>> sphere(30)
>> axis equal
```

piirretään yksikköpallo. Kokeile komennon sphere parametrin vaikutusta. Mikä on komennon **axis equal** vaikutus?

Esim.

```
>> cylinder([1 2 .5 .25],30)
>> view(45,20)
```

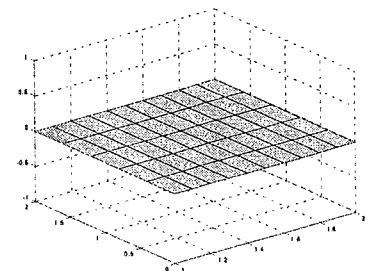
Selvitä itsellesi komennon cylinder parametrien merkitystä muuttamalla niiden arvoja ja piirtämällä kuvia. Voiko vektorin alkiot olla negatiivisia? Kokeile myös komentoon view erilaisia parametreja.



Esim. Tutkitaan komennon meshgrid toimintaa. Aja seuraavat komennot.

```
>> x = 1:.1:2;
>> y = 0:.4:2;
>> [X,Y]=meshgrid(x,y);
>> Z = zeros(size(X));
>> surf(X,Y,Z)
>> view(3)
```

Muuta vektoreita x ja y. Miten ruudukko muuttuu? Kokeile komentoa view(2). Miten kuva muuttuu?



10.1 Funktioesitys

Funktioesityksessä pinta on annettu muodossa

$$z = f(x, y),$$

missä $a \leq x \leq b$, $c \leq y \leq d$.

Funktioesityksen piirto suoritetaan seuraavasti:

1. Määritetään x - ja y -akselin välit vaakavektoreina esim. komennolla

$$x = \text{linspace}(a,b,n), y = \text{linspace}(c,d,n),$$

missä n ilmoittaa jakopisteiden lukumäärän.

2. Muodostetaan tason suorakulmion ruudukko komennolla

$$[X,Y] = \text{meshgrid}(x,y)$$

3. Lasketaan pinnan z -koordinaatit

$$Z = f(X,Y).$$

4. Piirtokomento riippuu siitä millainen kuva piirretään.

10.1.1 Perspektiivikuva

Perspektiivikuva piirretään seuraavasti:

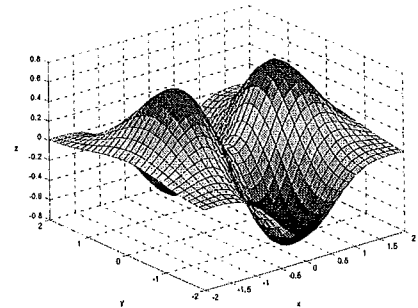
4. Piirretään kuva komennolla

$$\text{surf}(X,Y,Z,\dots)$$

Esim. Piirretään pinnan

$$z = (2x^2 - y^2)e^{-x^2-0.5y^2}, \quad -2 \leq x \leq 2, \quad -2 \leq y \leq 2$$

perspektiivikuva.



Luodaan tason suorakulmion $-2 \leq x \leq 2$, $-2 \leq y \leq 2$ ruudukko

```
>> [X,Y] = meshgrid(linspace(-2,2,30));
```

Määritetään pinnan z -koordinaatit ja piirretään pinta:

```
>> Z = (2*X.^2-Y.^2).*exp(-X.^2-0.5*Y.^2);
```

```
>> surf(X,Y,Z)
```

Perspektiivikuvan piirto voidaan tehdä omaksi funktioksi:

```
function own_surf(fun_handle,xmin,xmax,ymin,ymax,n)
% OWN_SURF Perspektiivikuvan piirto
% own_surf(fun_handle,xmin,xmax,ymin,ymax,n)
% Kutsuparametrit:
% fun_handle    Otin piirrettävään funktioon
% xmin, xmax    x-akselin rajat
% ymin, ymax    y-akselin rajat
% n             jakopisteiden lukumäärä x- ja y-akselilla

x = linspace(xmin,xmax,n);
y = linspace(ymin,ymax,n);

[X,Y] = meshgrid(x,y);
Z = feval(fun_handle,X,Y);
```

```
surf(X,Y,Z);
```

Jos edellisen esimerkin funktio tallennetaan M-tiedostoon fun.m, piirto tapahtuu komennolla

```
>> own_surf(@fun,-2,2,-2,2,30)
```

TEHTÄVIÄ

Piirrä seuraavien funktioiden kuvaajat. Kokeile view-komentoa.

- $f(x,y) = \sin(x)\cos(y)$, kun $-5 \leq x \leq 5$, $-3 \leq y \leq 3$
- $f(x,y) = \sin(\sqrt{x^2 + y^2})/\sqrt{x^2 + y^2}$, kun $-10 \leq x \leq 10$, $-10 \leq y \leq 10$

10.1.2 Tasa-arvokäyrästä

Pinnan tasa-arvokäyrästä koostuu tason käyristä

$$f(x,y) = k,$$

missä k saa eri arvoja.

Pinnan tasa-arvokäyrästä on käytettävissä seuraavia komentoja:

Komento	Toiminto
contour(Z)	Tasa-arvokäyrästä matriisin Z arvoista. Matriisin alkio Z_{ij} on funktion arvo xy -tason pisteissä (i,j) .
contour(x,y,Z)	Matriisin Z tasa-arvokäyrästä, missä vektorit x ja y määräävät x- ja y-akselin arvot.
contour(X,Y,Z)	Matriisin Z tasa-arvokäyrästä, missä X ja Y ovat matriiseja. Samaa kertalukua olevat matriisit X, Y ja Z määrittävät pinnan pisteet kuten komennossa surf.
contour(...,n)	Tasa-arvokäyrillä n arvoa.
contour(...,v)	Piirretään tasa-arvokäyrät vektorin v määräämillä arvoilla
contour(...,str)	Merkkijono str määrittää viivatyypin ja värin.
contour3(x,y,Z,v)	Tasa-arvokäyrästä 3-ulotteisesti.
C = contour(...)	Antaa matriisin C, jota esim. clabel voi käyttää
clabel(C)	Arvot tasa-arvokäyrille.
clabel(C,v)	Arvot tasa-arvokäyrille, jotka vektori v määrää.

Pinnan tasa-arvokäyrästä muodostetaan seuraavasti:

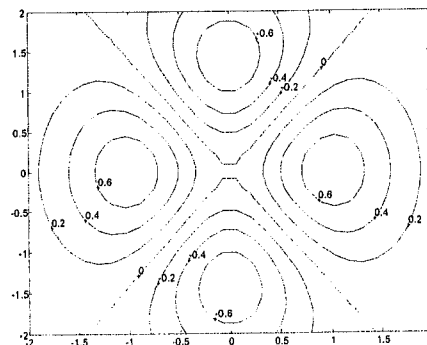
- Piirretään tasa-arvokäyrästä komennolla

```
contour(x,y,Z,...)
```

Esim. Piirretään pinnan

$$z = (2x^2 - y^2)e^{-x^2 - 0.5y^2}, \quad -2 \leq x \leq 2, \quad -2 \leq y \leq 2$$

tasa-arvokäyrästä.



Määritetään x - ja y -akselin jaot

```
>> x = linspace(-2,2,30);
```

Muodostetaan ruudukko xy -tasoon

```
>> [X,Y] = meshgrid(x);
```

Määritetään pinnan z -koordinaatit ja piirretään pinta.

```
>> Z = (2*X.^2-Y.^2).*exp(-X.^2-0.5*Y.^2);
```

```
>> C = contour(x,x,Z);
```

Tasa-arvokäyrien arvot:

```
>> clabel(C)
```

TEHTÄVIÄ

1. Piirrä edellisen kohdan tehtävien funktioiden tasa-arvokäyrät.
2. Tee M -funktio `own_contour`, jolla voit piirtää tasa-arvokäyrästäjä.

10.2 Parametriesitys

Pinnan parametrimuotoinen esitys on

$$\begin{cases} x = f(s,t) \\ y = g(s,t) \\ z = h(s,t) \end{cases},$$

missä $a \leq s \leq b$, $c \leq t \leq d$.

Parametrimuotoisen esityksen piirto suoritetaan seuraavasti:

1. Määritetään parametrien s ja t liikkumisvälit esim. komennoilla


```
s = linspace(a,b,n), t = linspace(c,d,n);
```

 missä n ilmoittaa jakopisteiden lukumäärän.
2. Muodostetaan st -tason suorakulmion ruudukko komennolla


```
[S,T] = meshgrid(s,t)
```
3. Määritetään pinnan pisteiden koordinaatit


```
X = f(S,T)
Y = g(S,T)
Z = h(S,T)
```
4. Piirretään kuva komennolla


```
surf(X,Y,Z,...)
```

Esim. R -säteisellä origokeskisellä ympyrällä on parametriesitys

$$\begin{cases} x = R \cos \theta \cos \varphi \\ y = R \cos \theta \sin \varphi \\ z = R \sin \theta \end{cases},$$

missä $-\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}$, $0 \leq \varphi \leq 2\pi$. Piirrä 2-säteinen pallo.

Parametrien θ ja φ liikkumisväli:

```
>> theta = linspace(-pi/2,pi/2,30);
>> fii = linspace(0,2*pi,30);
```

Muodostetaan ruudukko tasoon.

```
>> [THETA,FII] = meshgrid(theta,fii);
```

Määritetään pallon x -, y - ja z -koordinaatit.

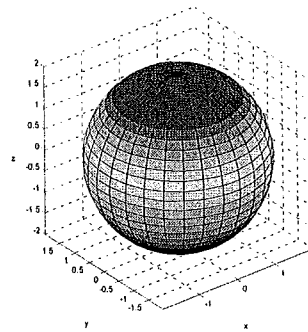
```
>> X = 2*cos(THETA).*cos(FII);
>> Y = 2*cos(THETA).*sin(FII);
>> Z = 2*sin(THETA);
```

Piirretään kuva

```
>> surf(X,Y,Z);
```

Akselimäärittäviä

```
>> axis equal;
>> xlabel('x'); ylabel('y'); zlabel('z');
```



TEHTÄVIÄ

1. Toruspinnalla on seuraava parametriesitys:

$$\begin{cases} x = (R + r \cos \theta) \cos \varphi \\ y = (R + r \cos \theta) \sin \varphi \\ z = r \sin \theta \end{cases}, \text{ missä } 0 \leq \theta \leq 2\pi, 0 \leq \varphi \leq 2\pi.$$

Piirrä toruspinta, kun $R = 5$ ja $r = 2$.

2. Tee M-funktio `own_parsurf`, jolla voit piirtää parametrimuodossa annettuja pintoja.

10.2.1 Pyörähdyspinnat

Tarkastellaan pinnan parametrimuotoisen esityksen erikoistapauksena pyörähdyspintojen määrittämistä.

Olkoon annettu xy -tason käyrä parametrimuodossa

$$\begin{cases} x = x(t) \\ y = y(t) \end{cases}, t \in [a, b].$$

Huom. Funktion $y = f(x)$, $x \in [a, b]$ parametriesitys on

$$\begin{cases} x = t \\ y = f(t) \end{cases}, t \in [a, b]$$

Jos käyrä **pyörähtää x -akselin ympäri**, syntyy pinta, jolla on parametriesitys

$$\begin{cases} x = x(t) \\ y = y(t) \cos \varphi \\ z = y(t) \sin \varphi \end{cases}, \text{ kun } a \leq t \leq b, 0 \leq \varphi \leq 2\pi$$

Jos käyrä **pyörähtää y -akselin ympäri**, syntyy pinta, jolla on parametriesitys

$$\begin{cases} x = x(t) \cos \varphi \\ y = y(t) \\ z = x(t) \sin \varphi \end{cases}, \text{ kun } a \leq t \leq b, 0 \leq \varphi \leq 2\pi$$

Tehdään M-funktio, joka piirtää pintojen perspektiivikuvat. Funktioissa käytetään if-lausetta², jonka rakenne on seuraava:

```

if ehto
    lauseita;
else
    lauseita;
end

function own_rotsurf(fun_handle,tmin,tmax,xy,n)
% OWN_ROTsurf Pyörähdyspinnan piirto
% own_rotsurf(fun_handle,tmin,tmax,xy,n)
% Kutsuparametrit:
% fun_handle    Otin piirrettävään funktioon, joka on muotoa
%               [x,y] = fun(t)
% tmin, tmax    parametrin t rajat
% xy            ilmoittaa pyörähdysakselin:
%               xy = 1: x-akseli
%               xy = 2: y-akseli
% n            jakopisteiden lukumäärä pinnan piirroksessa

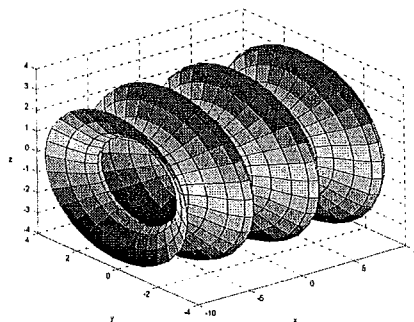
t = linspace(tmin,tmax,n);
fii = linspace(0,2*pi,n);

[T,FII] = meshgrid(t,fii);

if xy == 1
    [X,YY] = feval(fun_handle,T);
    Y = YY.*cos(FII);
    Z = YY.*sin(FII);
else
    [XX,Y] = feval(fun_handle,T);
    X = XX.*cos(FII);
    Z = XX.*sin(FII);
end

surf(X,Y,Z);
xlabel('x'); ylabel('y'); zlabel('z');

```



Esim. Piirretään pinta, joka syntyy, kun käyrä $y = 3 + \sin x$, $x \in [-10, 10]$ pyörähtää x -akselin ympäri.

Muodostetaan M-funktion fun.m, joka sisältää funktion

```

function [x,y] = fun(t)
x = t;
y = 3+sin(t);

```

Piirretään pinta komennolla

```
>> own_rotsurf(@fun,-10,10,1,30)
```

TEHTÄVIÄ

- Piirrä pinta, joka syntyy, kun käyrä $y = 0,3 + \frac{\sin x}{x}$ pyörähtää x -akselin ympäri välillä $[-\pi, 5\pi]$.

² if-lausetta käsitellään ohjelmoinnin yhteydessä.

2. Piirrä pinta, joka syntyy, kun käyrä $y = 0,3 + \frac{\sin x}{x}$ pyörii y -akselin ympäri välillä $[0, 5\pi]$.
3. Piirrä pinta, joka syntyy, kun käyrä $y = e^x$ pyörii y -akselin ympäri välillä $[0, 3]$. Tee kuvaan leikkaus: muokkaa ohjelmaa `own_rotsurf` siten, että kiertokulma `fii` liikkuu vain välillä $\left[-\pi, \frac{2}{3}\pi\right]$. Tallenna muokattu ohjelma eri nimellä.