

Graafikud

Matlabis ja Octaves on rohkelt võimalusi mitmesuguste graafikute ja diagrammide joonestamiseks. Vaatleme kõigepealt ühe muutuja funktsioonide graafikute joonestamist käskudega `fplot` ja `plot`.

Kõige lihtsam moodus ühe muutuja funktsiooni graafiku saamiseks on kasutada käsku `fplot`, mille kuju on järgmine:

```
fplot('funkts. valem',[argumendi väikseim väärtus,argumendi suurim väärtus])
```

Näiteks funktsiooni $\sin t$ graafiku saamiseks lõigul $0 \leq t \leq 10$ tuleks sisestada käsk

```
fplot('sin(t)',[0,10])
```

Graafiku paremasse ülemisse nurka tekib automaatselt nn legend. Legendi teksti saab muuta lisades käsu

```
legend('legendi tekst')
```

Graafikule saab lisada võrgu, so vertikaalsed ja horisontaalsed punktiirjooned, mis jooksevad läbi telgede sõlmpunktide. Selleks on käsk

```
grid on
```

Nii Matlab kui ka Octave avavad graafiku eraldi aknas. Matlabis on võimalik graafikut eksportida samast aknast. Selleks tuleb menüüst valida File, Save as ning sobiv faili formaat, milles soovitakse eksportida. Octaves saab graafikut eksportida käsuga `print`, mille kuju on järgmine:

```
print -dformaati faili nimi koos laiendiga
```

Näiteks joonise `figure1` eksportimisel `jpg`, `jpeg`, `ps` või `pdf` formaadis tuleb lisada vastavalt üks järgmistest käskudest:

```
print -djpg figure1.jpg
```

```
print -djpeg figure1.jpeg
```

```
print -dps figure1.ps
```

või

```
print -dpdf figure1.pdf
```

NÄITEÜLESANNE 1. Joonestada funktsiooni $f(x) = x^3 - x$ graafik lõigul $\frac{1}{2} \leq x \leq \frac{3}{2}$ koos võrguga. Lisada sobiv legend ja salvestada joonis nime `joonis1.pdf` all.

Lahendus. Kirjutame skripti järgmised read:

```
fplot('x^3-sqrt(x)',[.5,1.5]);
```

```
legend('x^3-x');
```

```
grid on;
```

Octave kasutamisel lisame ka käsu

```
print -dpdf joonis1.pdf
```

Seejärel salvestame ja käivitame. Graafik on [siin](#)

Ühel joonisel on võimalik kujutada ka mitu graafikut. Selleks tuleb käsus `fplot` funktsioonide valemid eraldada komadega ja asetada nurksulgudesse. Käsus legend tuleb samuti erinevate graafikute kohta käivad tekstid komadega eraldada.

NÄITEÜLESANNE 2. Joonestada funktsioonide $\sin t$, $\cos t$ ja $\sin 4t$ graafikud lõigul $[-2, 7]$. Lisada sobiv legend ja salvestada joonis nime `joonis2.jpg` all.

Lahendus. Kirjutame skripti järgmised read:

```
fplot('[sin(t),cos(t),sin(4*t)]',[-2,7]);
```

```
legend('sin t','cos t','sin 4t');
```

Octave kasutamisel lisame ka käsu

```
print -djpg joonis2.jpg
```

Seejärel salvestame ja käivitame. Graafik on [siin](#)

HARJUTUSÜLESANNE 1. Joonestada funktsioonide $y_1 = 60e^{-\frac{1}{x}}$ ja $y_2 = 7x^2 + 6x$ graafikud lõigul $[1, 3]$ võrguga. Lisada sobiv legend ja salvestada joonis nime joonis3.gif all. Vastav skript salvestada z20.m nime all.

Skript Joonis

Graafikutele on võimalik lisada telgede märgendeid käskudega

```
xlabel('märgendi tekst')
```

```
ylabel('märgendi tekst')
```

Lisaks on võimalik muuta graafikul oleva teksti suurust, stiili ja asukohta, joonte stiili, suurust ja värvi jm. Kõige selle kohta saab täpsemat infot Matlabi Helpist või Octave dokumentatsioonist.

Käsku `plot` on suhteliselt lihtne kasutada, kuid selle võimalused on üsna piiratud. Üks oluline kitsendus on see, et käsus esinevad funktsioonid tohivad sisaldada vaid ühte muutujat. Näiteks käsk

```
fplot('sin(314*t)',[0,0.1])
```

on korrektne, kuid

```
fplot('sin(w*t)',[0,0.1])
```

enam mitte (isegi juhul, kui muutujale w on eelnevalt väärtus antud).

Rohkem võimalusi pakub käsk `plot`, mille kuju on järgmine:

```
plot(xvektor,yvektor)
```

Suurused *xvektor* ja *yvektor* on võrdse pikkusega reaalarvudest koosnevad vektorid:

xvektor = (x_1, x_2, \dots, x_n)

yvektor = (y_1, y_2, \dots, y_n) .

Käsk `plot` toimib järgmiselt: tasandile joonestatakse punktid $P(x_i, y_i)$, $i = 1, \dots, n$ ja need ühendatakse järjest sirglõikudega. Seega tekib sirglõikudest koosnev pidev joon. Selleks, et graafik oleks sile (ilma nähtavate murdekohadeta), peavad punktid $P(x_i, y_i)$ paiknema piisavalt tihedalt. See tähendab, et vektorite *xvektor* ja *yvektor* komponendid peavad olema valitud piisavalt lühikese sammuga.

Vektoreid ja maatrikseid käsitlevas peatükis nägime, et ühtlase sammuga vektori loomiseks saab kasutada järgmist käsku:

```
vektori tähis = esimene element : samm : viimane element
```

Näiteks vektori $t_1 = 0, 0.01, 0.02, \dots, 0.5$ saab luua käsuga

```
t1=0:0.01:0.5
```

või

```
t1=0:1e-2:0.5
```

Funktsiooni väärtuste vektori loomiseks piisab funktsiooni rakendamisest argumenti vektorile.

NÄITEÜLESANNE 3. Joonestada funktsiooni $y = 5\sqrt{x} + 17$ graafik lõigul $[0, 3]$, lisada telgede märgendid ja salvestada nime joonis4.pdf all.

Lahendus. Koostame järgmise skripti

```
%Arvutame x vektori piisavalt lühikese sammuga (olgu selleks 0.001)
```

```
x=0:1e-3:3;
```

```
%Arvutame y vektori
```

```
y=5*sqrt(x)+17;
```

```
%Joonestame graafiku
```

```
plot(x,y)
```

```
xlabel('x')
```

```
ylabel('y')
```

Octave korral lisame ka käsu

```
print -dpdf joonis4.pdf
```

Salvestame skripti ja käivitame. Graafik on [siin](#)

Käsuga plot automaatselt legendi ei teki. Legendi saab soovi korral lisada käsuga legend nii nagu fplot korral. Soovitav on skriptis argumendi ja funktsiooni vektorite loomise käskude järgi panna ; Vastasel juhul kuvab Matlab-Octave need pikad vektorid ekraanile.

HARJUTUSÜLESANNE 2. Joonestada funktsiooni $z = \tan(\omega x)$ graafik lõigul $[0, \frac{\pi}{4\omega}]$, kui $\omega = 30$, lisada telgede märgendid, võrk ja salvestada nime tan.pdf all. Vastav skript salvestada nime z21.m all.

[Skript Joonis](#)

HARJUTUSÜLESANNE 3. Harmoonilise ostsillaatori sumbuvad võnked on kirjeldatud järgmise valemiga:

$$x(t) = Ae^{-\zeta\omega t} \sin\left(\sqrt{1-\zeta^2}\omega t + \varphi\right),$$

kus A on amplituud, ω nurksagedus, ζ sumbuviskoefitsient ja φ algfaas. Joonestada võngete graafik intervallil 0 kuni 0.2 sekundit järgmiste andmete korral:

$$A = 0.05\text{m}, \omega = 120 \frac{1}{\text{s}}, \zeta = 0.2, \varphi = \frac{\pi}{4}.$$

Lisada võrk, telgede märgendid ja sobiv legend. Salvestada joonis nime ostsillaator.pdf all. Vastav skript salvestada nime z22.m all.

[Skript Joonis](#)

Sama asi vene keeles:

Формула затухающих колебаний гармонического осциллятора имеет следующий вид:

$$x(t) = Ae^{-\zeta\omega t} \sin\left(\sqrt{1-\zeta^2}\omega t + \varphi\right)$$

где A - амплитуда, ω - угловая частота, ζ - коэффициент затухания и φ - начальная фаза. Нарисовать график колебаний при следующих данных на интервале от 0 до 0.2s:

$$A = 0.05\text{m}, \omega = 120 \frac{1}{\text{s}}, \zeta = 0.2, \varphi = \frac{\pi}{4}.$$

Нанести решетку, ярлыки осей и легенду. Скрипт записать в файл z22.m и график под именем ostsillaator.pdf.

[Скрипт График](#)

Käsuga plot saab joonestada ka mitu graafikut ühte teljestikku. Selleks tuleb käsku kirjutada mitu vektorite *xvektor* ja *yvektor* paari komadega eraldatult.

NÄITEÜLESANNE 4. Joonestada funktsioonide $z = \tan x$ ja $z = e^{2-x}$ graafikud lõigul $[2, 3]$, lisada telgede märgendid, võrk, legend ja salvestada nime joonis6.gif all.

Lahendus. Koostame järgmise skripti

```
%Arvutame x vektori
x=2:1e-4:3;
%Arvutame z vektorid
z1=tan(x);
z2=exp(2-x);
%Joonestame graafikud
plot(x,z1,x,z2)
```

```
xlabel('x')
ylabel('x')
grid('on')
legend('tan x','exp(2-x)')
```

Octave korral lisame ka käsu

```
print -dgif joonis6.gif
```

Salvestame skripti ja käivitame. Graafik on [siin](#)

Kui funktsioon $y = f(x)$ on antud ilmutamata kujul võrrandiga $F(x, y) = 0$, siis tema graafiku joonestamiseks saab kasutada käsku `ezplot`, mille kuju on järgmine:

```
ezplot('funkts. F valem', punktide arv, ristkülik)
```

Seejuures *punktide arv* näitab, kui palju punkte Matlab-Octave graafiku joonestamisel kasutab (mida rohkem punkte, seda siledam joon) ja *ristkülik* on piirkond, mille sisse graafik paigutatakse. *Ristkülik* tuleb ette anda järgmisel kujul:

```
[xmin, xmax, ymin, ymax] .
```

Parameetrid *punktide arv* ja *ristkülik* võib jätta käsus fikseerimata. Sellisel juhul valib Matlab-Octave need automaatselt.

NÄITEÜLESANNE 5. Joonestada ring, mille võrrand on $x^2 + y^2 = 1$ ja salvestada nime `ez1.pdf` all.

Lahendus. Sisestame skripti või käsureale käsu

```
ezplot('x^2+y^2-1')
```

ja käivitame. Saame järgmise [joonise](#). Antud juhul on Matlab moodustanud liiga suure ristküliku ringi ümber. Vähendame seda ristkülikut, nt piiridesse $-1.2 \leq x \leq 1.2, -1.2 \leq y \leq 1.2$. Selleks sisestame käsu

```
ezplot('x^2+y^2-1',[-1.2,1.2,-1.2,1.2])
```

ja käivitame. Tekib järgmine [joonis](#). Juhul, kui mingis Octave või Matlabi versioonis on saadav joonis "kandiline", siis tuleb suurendada joonisel kasutatavate punktide arvu. Näiteks 500 punktiga joonise saamiseks tuleb sisestada ja käivitada järgmine käsk:

```
ezplot('x^2+y^2-1',500,[-1.2,1.2,-1.2,1.2])
```

Käsule `ezplot` saab lisada ka võrku, legendi ja telgede märgendeid defineerivaid lisakäskke `grid on`, `legend`, `xlabel`, `ylabel` nii nagu me tegime seda eespool käskude `fplot` ja `plot` korral.

Automaatselt paigutab `ezplot` horisontaalteljele muutuja, mis esineb funktsiooni F valemis kõige esimesena. Kuna toodud näites esineb valemis $F(x, y) = x^2 + y^2 - 1$ kõige esimesena muutuja x , siis paigutubki horisontaalteljele x ja vertikaalteljele y .

HARJUTUSÜLESANNE 4. Joonestada joon, mille võrrand on $10t^2 = (1-t)(t^2 + x^2)$. Kohendada joonist ümbritseva ristküliku mõõtmed selliseks, et joone kvalitatiivne käitumine oleks paremini esile toodud. Lisada võrk. Salvestada joonis nime `ez2.pdf` all. Vastav skript salvestada nime `z23.m` all.

[Lahendus](#)

Ühe käsuga `ezplot` ei saa mitut graafikut (so joont) ühte teljestikku joonestada. Mitme graafiku joonestamiseks ühte teljestikku tuleb kasutada lisakäskke `hold on` ja `hold off`. Konkreetselt toimub nende käskude kasutamine järgmiselt. Peale esimese graafiku joonestamist käsuga `ezplot` tuleb sisestada käsk `hold on`. Seejärel tuleb lisada ülejäänud graafikutele vastavad käsud `ezplot` ning lõpuks käsk `hold off`.

NÄITEÜLESANNE 6. Joonestada ühte teljestikku ellipsid $x^2 + y^2 = 4$ ja $10x^2 + y^2 = 9$. Kohendada joonist ümbritsev ristkülik sobivaks ja lisada võrk. Joonis

salvestada faili ez3.pdf.

Lahendus. Koostame järgmise skripti

```
ezplot('x^2+y^2-4')
grid on
hold on
ezplot('10* x^2+y^2-9')
hold off
```

ja käivitame. Saadud joonist vaadeldes nähtub, et sobiv ristkülik oleks $-3 \leq x \leq 3$, $-4 \leq y \leq 4$. Seega kohendame skripti järgnevalt:

```
ezplot('x^2+y^2-4',[-3,3,-4,4])
grid on
hold on
ezplot('10* x^2+y^2-9',[-3,3,-4,4])
hold off
```

Octaves lisame ka käsu

```
print -dpdf ez3.pdf
```

Skripti käivitamisel saame järgmise [joonise](#).