

# Harjoitus 9: Optimointi I (Matlab)

SCI-C0200 Fysiikan ja matematiikan menetelmien studio 2023



# Harjoituksen aiheita

- Optimointimallin muodostaminen
- Optimointitehtävien luokittelu
- Optimointitehtävien ratkaiseminen Matlabilla

## Oppimistavoitteet

- Tiedät mitä on optimointi ja mitkä ovat optimointimallin osat
- Osaat käyttää Matlabin valmiita optimointityökaluja
- Osaat toteuttaa yksinkertaisen optimointialgoritmin
- Osaat analysoida optimoinnin tuloksen herkkyyttä mallin parametrien suhteen

# Mitä on optimointi?

- Optimoinnissa pyritään hakemaan ongelmalle paras (eli optimaalinen) ratkaisu.

## Optimointimallin osat

- Päätösmuuttujat (Engl. Decision variables)
- Mahdolliset rajoitukset (Engl. Constraints) koskien päätösvaihtoehtoja
- Kohdefunktio (Engl. Objective function) jota maksimoidaan tai minimoidaan

## Esimerkki: väännetään rautalangasta

- $L$ :n pituisesta rautalangasta on väännettävä suorakaide, jonka pinta-ala on suurin mahdollinen.
  - päätösmuuttujat:
    - $x$  : leveys [m]
    - $y$  : korkeus [m]
  - rajoitukset:
    - $2(x + y) = L$  : suorakaiteen kehän pituus on  $L$
    - $x, y \geq 0$  : muuttujat positiivisia
  - kohdefunktio:
    - $f(x, y) = xy$  : suorakaiteen pinta-ala

## ... Esimerkki

- Optimointitehtävä:

$$\max_{x,y} f(x,y) = xy$$

$$\text{st. } 2(x+y) = L$$

$$x, y \geq 0$$

st. : "subject to"

- Päättösmuuttujien arvot, jotka toteuttavat tehtävän yhtälö- ja epäyhtälörajoitukset muodostavat tehtävän **käyvän joukon** (Engl. search space tai choice set).
- Tehtävän ratkaisu:  $x = y = L/4$ . Rautalangasta kannattaa siis taivuttaa neliö.

# Optimointitehtävien luokittelu

## Lineaarinen optimointi

- Kohdefunktio ja rajoitukset lineaarisia
- Esim. leipomo maksimoi kinuski- ja suklaakakuista saatavan myyntituoton
- MS-E2121 Linear optimization

## Epälineaarinen optimointi

- Kohdefunktio ja/tai rajoitukset epälineaarisia
- Esim. etsi muoto liikkuvalla kappaleelle siten että ilmanvastus minimoituu
- MS-E2122 Nonlinear optimization

# ... Optimointitehtävien luokittelu

## Monitavoiteoptimointi

- Yhden kohdefunktion sijasta monta kohdefunktiota
- Esim. portfolion optimointi: halutaan sekä maksimoida tuottoa että minimoida riskiä

## Kokonaislukuoptimointi

- Päättömuuttujat ovat kokonaislukuja
- Esim. Halutaan määrittää optimaalinen tilausten lukumäärä, jolla yrityksen voitto maksimoituu
- MS-E2146 Integer programming

# ... Optimointitehtävien luokittelu

## Dynaaminen optimointi

- Päätosmuuttujat ovat funktioita toisista muuttujista, esim. ajasta
- Kohdefunktio on funktionaali, esim. integraali
- Esim. muodosta kulutus-säästöstrategia pääomalle kahden vuoden ajaksi; kulutuksesta seuraa hyöty, säästöstä korko
- MS-E2148 Dynamic optimization

## Stokastinen optimointi

- Tehtävä sisältää satunnaismuuttujia.
- MS-C2111 Stochastic Processes



# Optimointitehtävien ratkaiseminen

- Tehtäviä ratkaistaan analyttisesti tai numeerisesti
- Optimointitehtävien **numeerinen** ratkaisu perustuu **iterointiin**.
  - Piste päätösmuuttujien käyvässä joukossa edustaa ratkaisukandidaattia
  - Lähdetään tietyistä alkupisteistä liikkeelle
  - Lasketaan uusi piste tietyn algoritmin mukaisesti
  - Jatketaan kunnes löydetään optimipiste
- Sopii tietokoneen ratkaisukeinoksi
- Historiallisista syistä tiettyjä numeerisia ratkaisumenetelmiä kutsutaan **ohjelmoinniksi** (esim. lineaarinen ohjelmointi)

# Optimointitehtävien ratkaisu Matlabilla

- Matlab Optimization Toolbox
  - `fminbnd`: Etsii yhden muuttujan funktion minimin annetulta väliltä.
  - `fmincon`: Etsii minimin optimointitehtävälle, jossa kohdefunktio on epälineaarinen monen muuttujan funktio ja rajoitteet mielivaltaisia yhtälö- ja epäyhtälörajoitteita.
  - `fminsearch` ja `fminunc`: Etsii minimin epälineaarille kohdefunktiolle, kun rajoitusehtoja ei ole.
  - `linprog`: Ratkoo lineaarisia optimointitehtäviä.
  - `quadprog`: Ratkoo kvadraattisia (neliöllisiä) optimointitehtäviä.

# fmincon

- fmincon-syntaksi:

$X = \text{fmincon}(@(\mathbf{x}) \text{ FUN}(\mathbf{x}, ? \dots), X0, A, B, Aeq, Beq, LB, UB, @(\mathbf{x}) \text{ mycon}(\mathbf{x}, ? \dots))$

- $\text{FUN}(\mathbf{x}, ? \dots)$  on kohdefunktio, joka määrittää omassa m-tiedostossaan.  $@(\mathbf{x})$  kohdefunktion edessä kertoo, minkä argumentin suhteen optimointi tehdään.  $? \dots$  viittaa kohdefunktion mahdollisiin parametreihin.
- $X0$  on optimointialgoritmille annettava alkuarvaus ratkaisusta
- $A, B, Aeq, Beq$  liittyvät tehtävän lineaarisiin rajoitusehtoihin.  $LB$  ja  $UB$  määrittävät päätösmuuttujille ala- ja ylärajat.
- Funktion  $\text{mycon}(\mathbf{x}, ? \dots)$  avulla voi lisätä kaikenlaisia yhtälö- ja epäyhtälörajoitteita, myös epälineaarisia.  $? \dots$  viittaa mahdollisiin parametreihin.

# Rautalankatehtävä Matlabissa

```
% Omaan m-tiedostoon "pintaala.m"
function A = pintaala(x)
    A = -x(1)*x(2);
% Omaan m-tiedostoon "rajoitus.m"
function [c ceq] = rajoitus(x, L)
c=[];
ceq=[2*(x(1)+x(2))-L];
% Matlabin komentoriviltä
>> L = 29.3;
>> [x,fval] = fmincon(@(x) pintaala(x),[1; L/2-1],[],[],[],[],...,
[0; 0],[],@(x) rajoitus(x,L))
Local minimum found that satisfies the constraints ...
x =    7.3250
      7.3250
fval = -53.6556
```

## ... Rautalankatehtävä Matlabilla

- Kohdefunktiolle oltava oma funktiotiedosto
- rajoitus-funktiolla voisi toteuttaa myös epälineaarisia rajoitteita
- Huomaa @-symbolin käyttö
- Oletusarvoisesti minimoidaan; helppo muuntaa maksimoinniksi merkkimuutoksella:

funktion  $f(x)$  maksimointi  $\Leftrightarrow$  funktion  $-f(x)$  minimointi

# Tehtävä A: Öljyfirman kustannusten minimointi

- EOQ-malli (Economic Order Quantity) kuvaa yrityksen varastointi- ja tilauskustannuksia
- Mallissa kustannukset  $f(x_1)$  tilauskoolla  $x_1$ , tilauskustannuksilla  $a_1$ , vuosikysynnällä  $b_1$  ja varastointikuluilla  $h_1$  muodostuvat seuraavasti

$$f(x_1) = \underbrace{a_1 \times \frac{b_1}{x_1}}_{\text{Kustannukset per tilaus} \times \text{tilauksien lkm}} + \underbrace{h_1 \times \frac{x_1}{2}}_{\text{varastointikulut per yksikkö} \times \text{keskimääräinen varasto}} \quad (1)$$

- Malli pätee, kun tilauskustannukset, toimitusajat sekä kysyntä ovat tasaisia yli vuoden.

# Tehtävä A: Kohdefunktio

- Auta öljyfirmaa minimoimaan öljyn varastointi- ja tilauskustannuksia. Öljyä on kahta tyyppiä, jolloin EOQ malli on

$$f(x_1, x_2) = \left( \frac{a_1 b_1}{x_1} + \frac{h_1 x_1}{2} \right) + \left( \frac{a_2 b_2}{x_2} + \frac{h_2 x_2}{2} \right), \quad (2)$$

missä  $x_i$  = Öljytyypin  $i$  tilauskoko [tonnia / tilauskerta],

$a_i$  = Öljytyypin  $i$  tilauskustannus [10 000 €/ tilauskerta],

$b_i$  = Öljytyypin  $i$  kysyntä vuodessa [tonnia],

$h_i$  = Öljytyypin  $i$  varastointikulut yksikköä kohti vuodessa [10 000 €/ tonni].

1. Toteuta funktio `function f = oljykustannusfunktio(x,a,b,h)` joka ottaa sisäänsä päätösmuuttujat  $x = [x_1, x_2]$ , parametrin  $a = [a_1, a_2]$ ,  $b = [b_1, b_2]$ ,  $h = [h_1, h_2]$  ja palauttaa kustannukset.

## Tehtävä A: Rajoitusehdot

- Varastointitila on rajattu ja tilausmäärien on oltava positiivisia:

$$g(x_1, x_2) = t_1x_1 + t_2x_2 \leq T, \quad (3)$$

$$x_1, x_2 \geq 0, \quad (4)$$

missä  $t_i$  = vaadittava varastointitila per tonni tyyppin  $i$  öljyä ja  $T$  = olemassa oleva varastointitila.

- Varaston koko  $T = 24$ .



## Tehtävä A: Rajoitusehdot

2. Luo function `[c ceq] = oljyrajoitus(x,t,T)`. Funktio ottaa sisäänsä tilauskoot  $x$  sekä parametrit  $t = [t_1, t_2]$  ja  $T$ .

- Funktion palauttaman arvon  $c$  tulee kertoa yhtälörajoitteen (3) vasemman- ja oikeanpuoleisen lausekkeen erotus. Aseta `ceq=[]`.
- `fmincon` hyödyntää tätä funktiota siten, että optimointin edetessä toteutuu  $c \leq 0$ .
- Ei-negatiivisuusrajoitteet (4) toteutetaan suoraan `fmincon:iin`, jota käytetään optimaalisen tilauksen ratkaisemiseen.

# Tehtävä A: Optimointi

3. Ratkaise optimaaliset arvot päätösmuuttujille  $x_1$  ja  $x_2$  käyttäen funktiota `fmincon`. Käytä seuraavia parametrien arvoja:

Öllytyyppi	$a_i$	$b_i$	$h_i$	$t_i$
$i=1$	10	2	0.4	2
$i=2$	6	3	0.15	4

Vinkkejä:

- Katso `!help fmincon` viimeinen esimerkki.
- Käytä apuna kohdissa 1 ja 2 luomiasi funktioita.
- Kirjoittamalla kohdefunktion eteen  $@(\mathbf{x})$  ilmaiset minkä argumentin suhteen kohdefunktiota optimoidaan.
- Käytä optimointialgoritmin alkuarvauksena  $(x_1, x_2) = (4, 4)$

# Tehtävä A: Optimointi

✎ Kuinka suuria tilausmäärien tulee olla, jotta öljyfirman kustannukset minimoituvat? Kuinka suuret kustannukset tällöin ovat?

## Tehtävä B: Gradienttimenetelmä

- Rosenbrockin banaanifunktio on esimerkki funktiosta, jonka minimiä monet optimointialgoritmit eivät löydä.
- Tutkitaan banaanifunktion

$$f(x_1, x_2) = 10 \cdot (x_2 - x_1^2)^2 + (1 - x_1)^2$$

minimoimista itse toteutetulla optimointialgoritmilla, gradienttimenetelmällä.

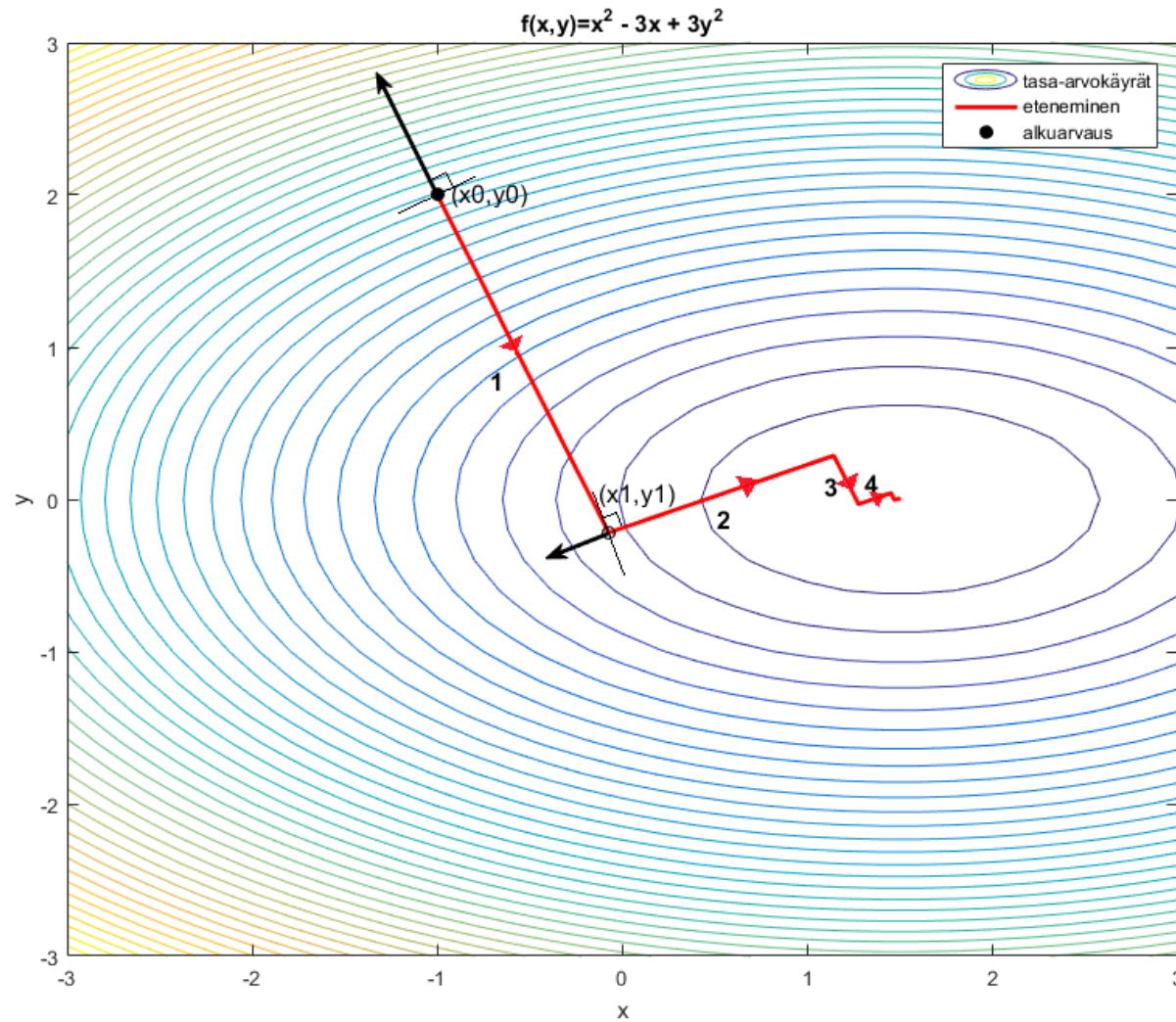
## Tehtävä B: Gradienttimenetelmä

- Ratkaisukandidaattia  $x(i) = [x_1(i), x_2(i)]'$  päivitetään iteratiivisesti käyttäen gradienttia.
- Pisteessä  $x(i)$  lasketaan funktion gradientti  $\nabla f(x_1(i), x_2(i))$  ja siirrytään gradienttia vastakkaisen suunnan mukaisesti pisteeseen  $x(i + 1)$ , s.e.

$$\begin{bmatrix} x_1(i + 1) \\ x_2(i + 1) \end{bmatrix} = \begin{bmatrix} x_1(i) \\ x_2(i) \end{bmatrix} - \alpha \cdot \nabla f(x_1(i), x_2(i)) \quad (5)$$

missä  $\alpha$  on askelpituus. Askelpituus on tässä ratkaiseva, jotta ei mennä liian pitkälle tai jäädä liian lyhyeen gradientin suuntaan edetessä.

# Tehtävä B: Gradienttimenetelmän kulku



## Tehtävä B: Toteutus

1. Muodosta function `[grad] = gradientti(x)`, joka laskee kohdefunktion gradientin `grad=[g1;g2]` funktiolle annetussa pisteessä `x=[x1;x2]`.

Gradientti on muotoa

$$\nabla f(x_1, x_2) = \begin{bmatrix} 10 \cdot 2 \cdot (x_2 - x_1^2) \cdot (-2 \cdot x_1) - 2 \cdot (1 - x_1) \\ 10 \cdot 2 \cdot (x_2 - x_1^2) \end{bmatrix}$$

2. Muodosta function `[f_uusi]=viiva(alpha,grad,x)`, joka palauttaa kohdefunktion arvon uudessa pisteessä, johon päädytään kun pisteestä `x` siirrytään `alpha`-pituinen matka suuntaan `-grad`. Askelpituus  $\alpha$  saadaan yksiulotteiden optimointitehtävän (viivahaun) ratkaisuna. Viivahaussa haetaan  $\alpha$ :n arvo, jolla kohdefunktio minimoituu, kun edetään `-grad` suuntaisesti.

## ... Tehtävä B: Toteutus

3. Kirjoita ohjelma, jolla tutkit algoritmin etenemistä, kun lähdetään alkupisteestä  $(x_1, x_2) = (-1, 2)$ . Askelpituus **alpha** määritetään erikseen jokaisella askeleella, käyttäen Matlabin funktiota **fminbnd**. Ohjelman runko:

```
x = zeros(2,100) % Varataan tilaa ratkaisuille
x(:,1) = [-1;2] % 1. sarake vastaa alkupistettä
for i = 2:100 % Suoritetaan 100 askelta (iteraatiota)
    % Määritä gradientin suunta pisteessä x(:,i-1)
    grad = ???
    % Määritä askelpituus alpha minimoimalla funktiota
    % viiva(alpha,grad,x(:,i-1)) välillä 0 <= alpha <= 5
    alpha = fminbnd(@(???) minimoitava funktio?,0,5)
    x(:,i) = x(:,i-1) + ??? % Päivitä piste
end
```



## Tehtävä B: Kysymyksiä

- ✎ Mitä pistettä gradienttimenetelmä lähestyy? Onko tämä piste funktion minimi, ja jos on, mistä tiedät sen?
- ✎ Kuinka monen askeleen kuluttua lähtöarvauksesta on algoritmi päässyt banaanilaakson pohjalle ja kulku merkittävästi hidastunut?
- 🖨 Liitä vastauksiisi kuva josta näkyy iteraation kulku gradienttimenetelmällä (iteroidut pisteet ja niitä yhdistävä murtoviiva) sekä kohdefunktion tasa-arvokäyrät  $(x_1, x_2)$ -tasossa. Nimeä akselit. Käytä tasa-arvokäyrien piirtämisessä komentoja `meshgrid` ja `contour`. Esimerkiksi funktiolle  $x e^{-x^2 - y^2}$  saat piirrettyä tasa-arvokäyrät näin:  

```
[X,Y] = meshgrid(-2:.2:2, -2:.2:2);  
Z = X .* exp(-X.^2 - Y.^2);, contour(X,Y,Z,100);
```

## Kotitehtävä: Jatkoa EOQ-tehtävälle

Esitit EOQ-mallin tulokset öljy-yrityksen johdolle. Esittämäsi tulos ei kuitenkaan riitä johtoportaallesi, joka vaatii sinulta lisää.

- Johtoporras esittää seuraavan kysymyksen:



*Mitä kustannuksille käy, jos yritys muuttaa varaston kokoa, s.e.*

$T \in \{14, 15, \dots, 34\}$ ?

1. Tutki miten optimoidut kustannukset käyttäytyvät  $T$ :n funktiona.

- Luo function `fval = optimi_kustannus(a,b,h,t,T)`, joka palauttaa optimoidut kustannukset annetuilla parametreilla. (Vinkki: Käytä tunnilla tehtyjä funktioita ja `fmincon`:illa toteutettua optimointia.)

# Kotitehtävä: Jatkoa EOQ-tehtävälle

- Piirrä kuvaaja optimoiduista kustannuksista, kun  $T=14:1:34$ .
-  Liitä kuva kustannuksista varastokoon funktiona. Nimeä akselit.  
Kommentoi kuvaajaa.
-  Kuinka paljon  $T$ :n kasvattaminen 24:stä 30:een saa korkeintaan maksaa, jotta investointi tuottaisi itsensä takaisin vuoden aikana?


# Kotitehtävä: Kustannusten kehitys tilauskustannusten funktiona


Yrityksen johtoporrasta kiinnostaa lisäksi, miten optimoidut kustannukset muuttuisivat, jos tilauskustannukset  $a_1, a_2$  muuttuvat, mutta varastokoko ei muutu, eli  $T = 24$ .

2. Tehtävänä on visualisoida optimoidut kustannukset, kun  $a_1 \in \{8, 8.25, \dots, 12\}$  ja  $a_2 \in \{4, 4.25, \dots, 8\}$ . Toteutus täydentämällä seuraavan kalvon koodinpätkää (sisemmän for-silmukan sisään).

```
a1_grid=[8:0.25:12]; a2_grid=[4:0.25:8];
F=zeros([length(a2_grid),length(a1_grid)]);
for i=1:length(a1_grid)
    for j=1:length(a2_grid)
        a(1)=a1_grid(i);
        a(2)=a2_grid(j);
        F(j,i)= %Tallenna F(j,i):hin optimi näillä a(1)
                %ja a(2) arvoilla
    end
end
end
%Pinnan piirto
surf(a1_grid,a2_grid,F)
colorbar %väriselite
%Nykytilanteen merkkkaus 3D-kuvaan
text(10, 6.05, 10.3966,'\leftarrow Cost now','FontSize', 16)
xlabel('a1')
ylabel('a2')
```

# Kotitehtävä: Kysymyksiä

 Selitä mitä edellisen kalvon matlab-komennon `text` parametrit tekevät.

 Liitä piirtämäsi kuva. Käännä sitä sopivaan asentoon matlabin figure-ikkunan työkalulla Rotate 3D (Tools → Rotate 3D). Nimeä akselit sopivasti. Kommentoi kuvaa.

 Miten optimoidut kustannukset riippuvat tilauskustannuksista?

Yritykselle tarjoutuu mahdollisuus vaihtaa toimittajaan, jolta tilattaessa kustannukset olisivat  $a_1 = 9.5$ ,  $a_2 = 5$ . Lue kuvasta mitkä kustannukset olisivat tällöin. Vinkki: Tools → Data Cursor

 Kuinka paljon yritys säästäisi vuosittain, jos se vaihtaisi toimittajaa?