Wiskunde 13/14

Jan van Hulzen / Liv Harkes
September 3rd 2012 version 1.0

Optimization

Optimalisatie

Organisatie van de cursus

Week	50 min	100 min	100 min	
1	HC	PR	ZS	Opdracht 1
2	HC	PR	ZS	
3	HC	PR	ZS	Opdracht 2
4	HC	PR	ZS	
5		PR	ZS	Opdracht 3
6		PR	ZS	
7		PR	ZS	

Opdrachten

- Opdracht 1
  - Mathematica + kleine optimalisatie opdracht
- Opdracht 2
  - Vertalen probleem naar wiskundige formulering
  - Coderen en oplossen van probleem met mathematica
  - Analyse van oplossing
- Opdracht 3
  - Vertalen probleem naar wiskundige formulering
  - Oplossen met behulp van zelfgeschreven optimalisatie algoritme
  - Analyse van oplossing

Leerdoelen

- De student is in staat om
  - Een optimalisatie probleem in wiskundige termen te formuleren.
  - Het in wiskundige termen geformuleerde probleem te programmeren in een wiskundig pakket (mathematica).
  - De resultaten te analyseren.
  - Het in wiskundige termen geformuleerde antwoord te vertalen naar een in algemene termen geformuleerd advies en dit vast te leggen in een verslag.

Agenda

Maandag

Start	Eind	Activiteit	Locatie	Weken	Klas/Groep
09:20	10:10	Wiskunde 13 (HC)	C1.04	36-39	DT/AVV4 K, L, M, N, O, P, Q, R
16:00	16:50	Wiskunde 13 (PR)	A3.40	36-39	DT/AVV4 A, B, C, D, E, F, G, H, I

Dinsdag

Start	Eind	Activiteit	Locatie	Weken	Klas/Groep
08:30	10:10	Wiskunde 13 (PR)	B8.38	36-42	DT/AVV4 P, Q, R

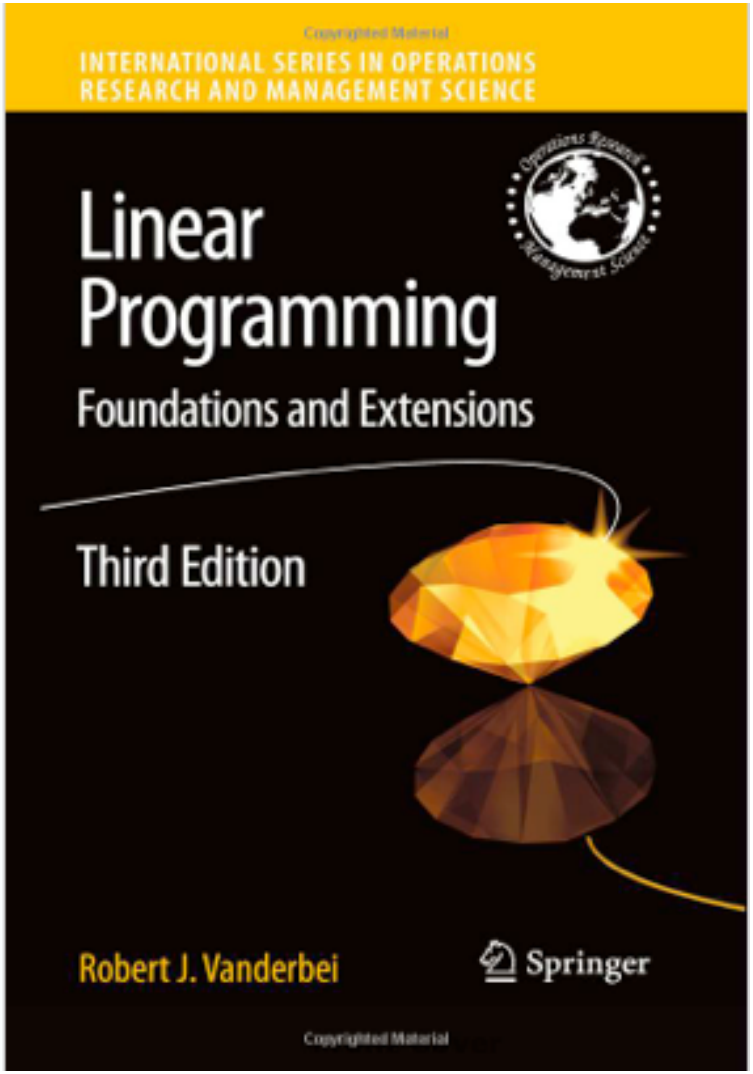
Woensdag

Start	Eind	Activiteit	Locatie	Weken	Klas/Groep
13:30	15:10	Wiskunde 13 (PR)	B8.08	36-42	DT/AVV4 G, H, I

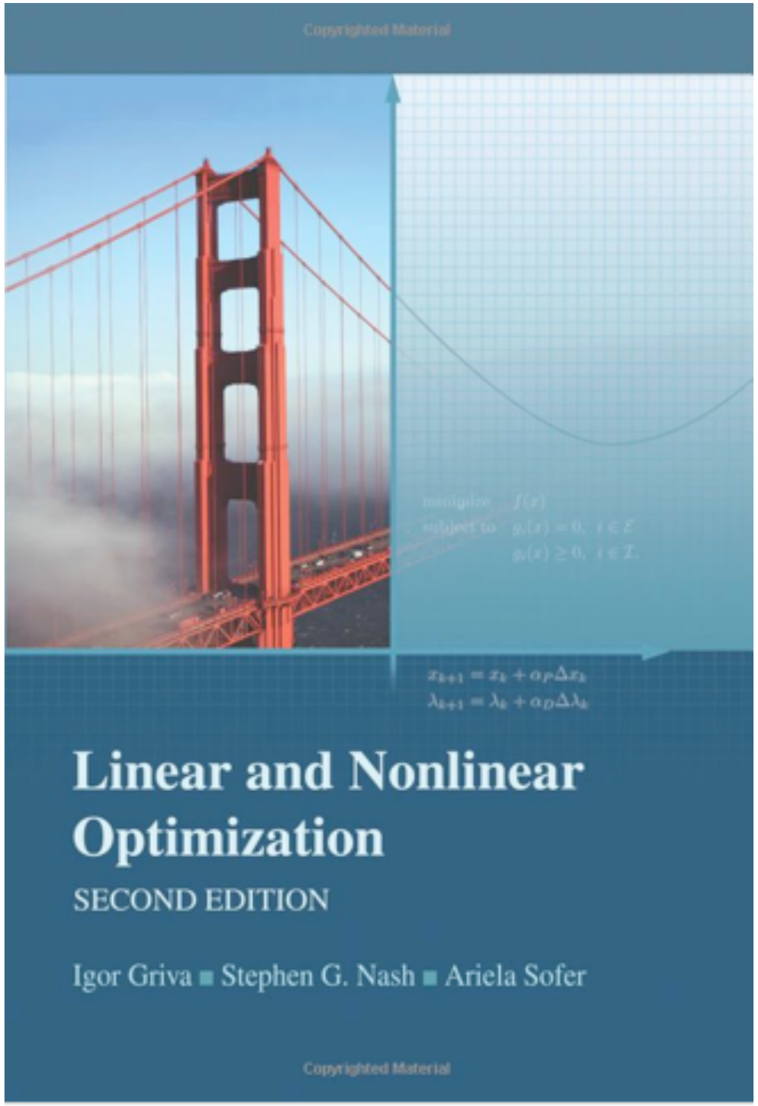
Beoordeling

- Het verslag wordt beoordeeld op :
  - De kwaliteit van het onderzoek,
  - De kwaliteit en formulering van de verslaglegging
    - Vaste elementen bevatten de juiste informatie (samenvatting, inleiding, methode omschrijving, resultaten, conclusie).
    - Redeneringen hebben de observatie-conclusie structuur
    - Nederlands is correct (met name de opbouw van de zinnen)
- Alle verslagen moeten voldoende zijn
- Het cijfer wordt gemiddeld over alle verslagen

Lesmateriaal : Linear programming

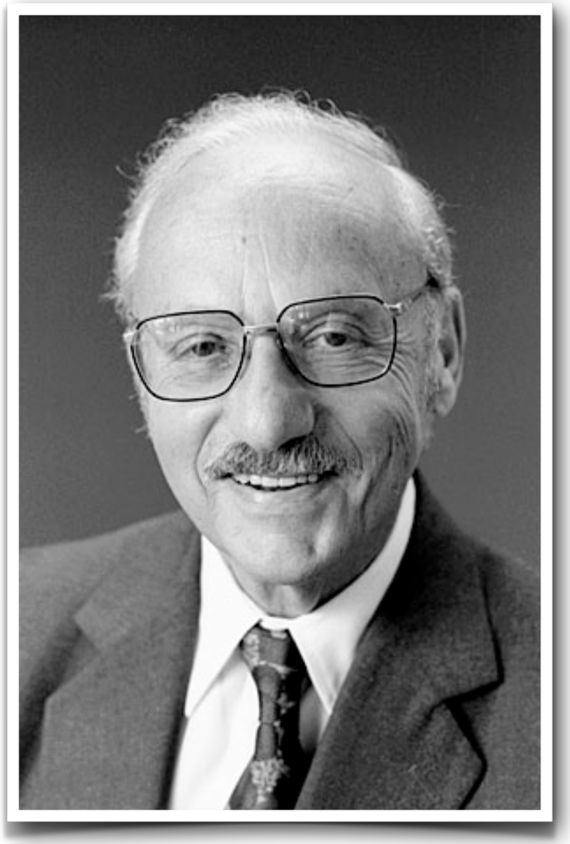


Lesmateriaal



Overzicht

- Lineare Optimalisatie
  - Simplex Algoritme
    - Grafisch
      - Hoogtelijnen
      - Hoekpunten
    - Algebraïsch
      - Inspectie
      - Simplex Tableau
      - Matrix algebra
    - Problemen
      - Degeneracy
      - Sensitivity
      - Cycling
      - Unboundness



George Dantzig

Lineaire optimalisatie : Shader Electronics

- Elektronica fabrikant “Shader Electronics” heeft twee producten:
  - Een portable CD/DVD player, de “Shader Walkman”
  - Een miniatuur internet kleuren-TV, de “Shader Watch-TV”

- De gegevens van het productieproces en de winst-per-unit zijn:

	Hours Required to Produce 1 Unit		
Department	**Walkmans (X_1)**	**Watch-TVs (X_2)**	**Available Hours This Week**
Electronic	4	3	240
Assembly	2	1	100
Profit per unit	\$7	\$5	

- Bereken de productie aantallen die tot de meeste winst leiden

Lineaire optimalisatie

- De variabelen van het probleem zijn:
  - X_1 is het aantal geproduceerde Walkmans
  - X_2 is het aantal geproduceerde TV's

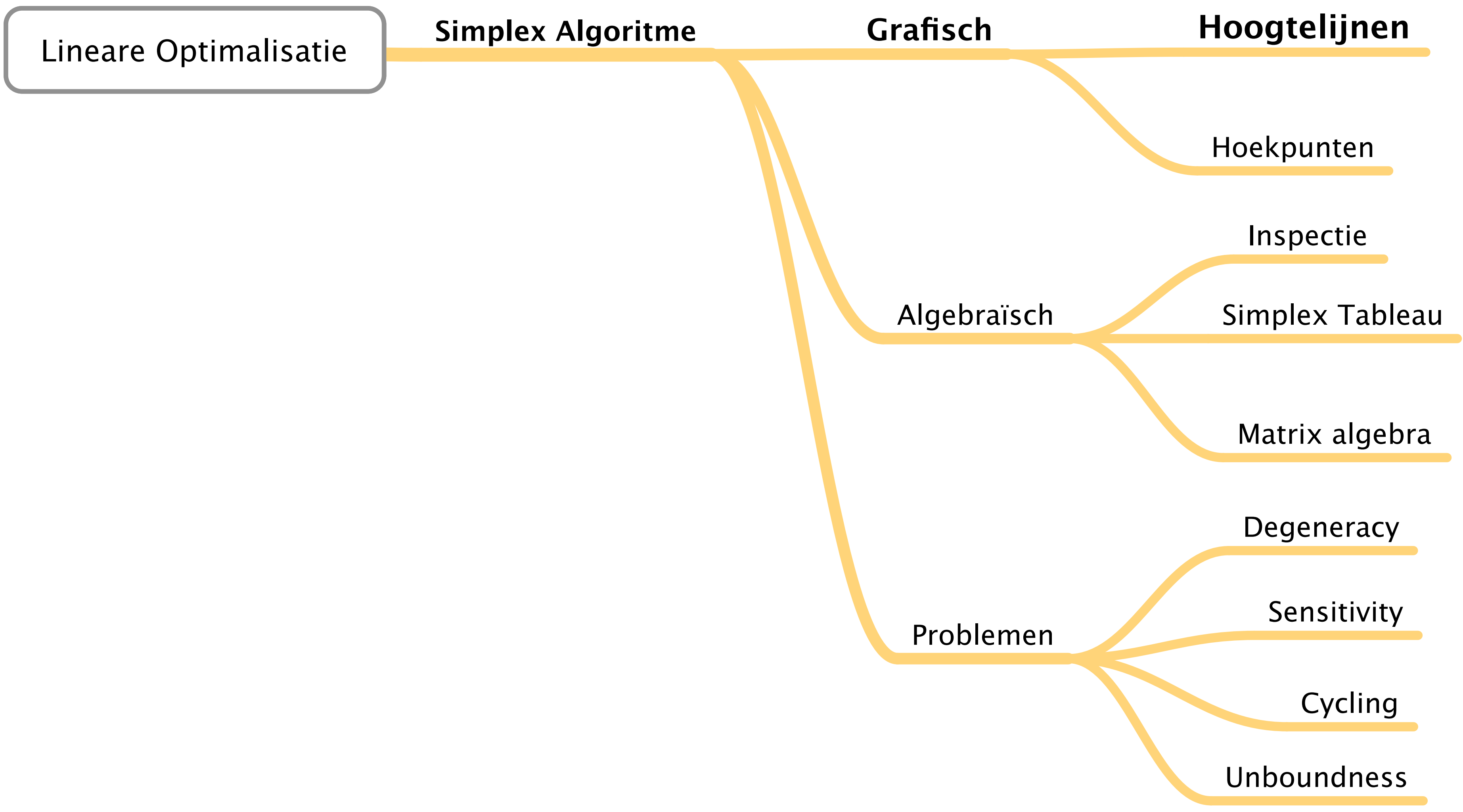
- De objective functie is
  - Winst = $7X_1+5X_2$

- De constraints zijn:

A. Fabricage van elektronica ≤ beschikbare tijd : $4X_1 + 3X_2 \leq 240$

B. Assemblage ≤ beschikbare assemblage tijd: $2X_1 + 1X_2 \leq 100$

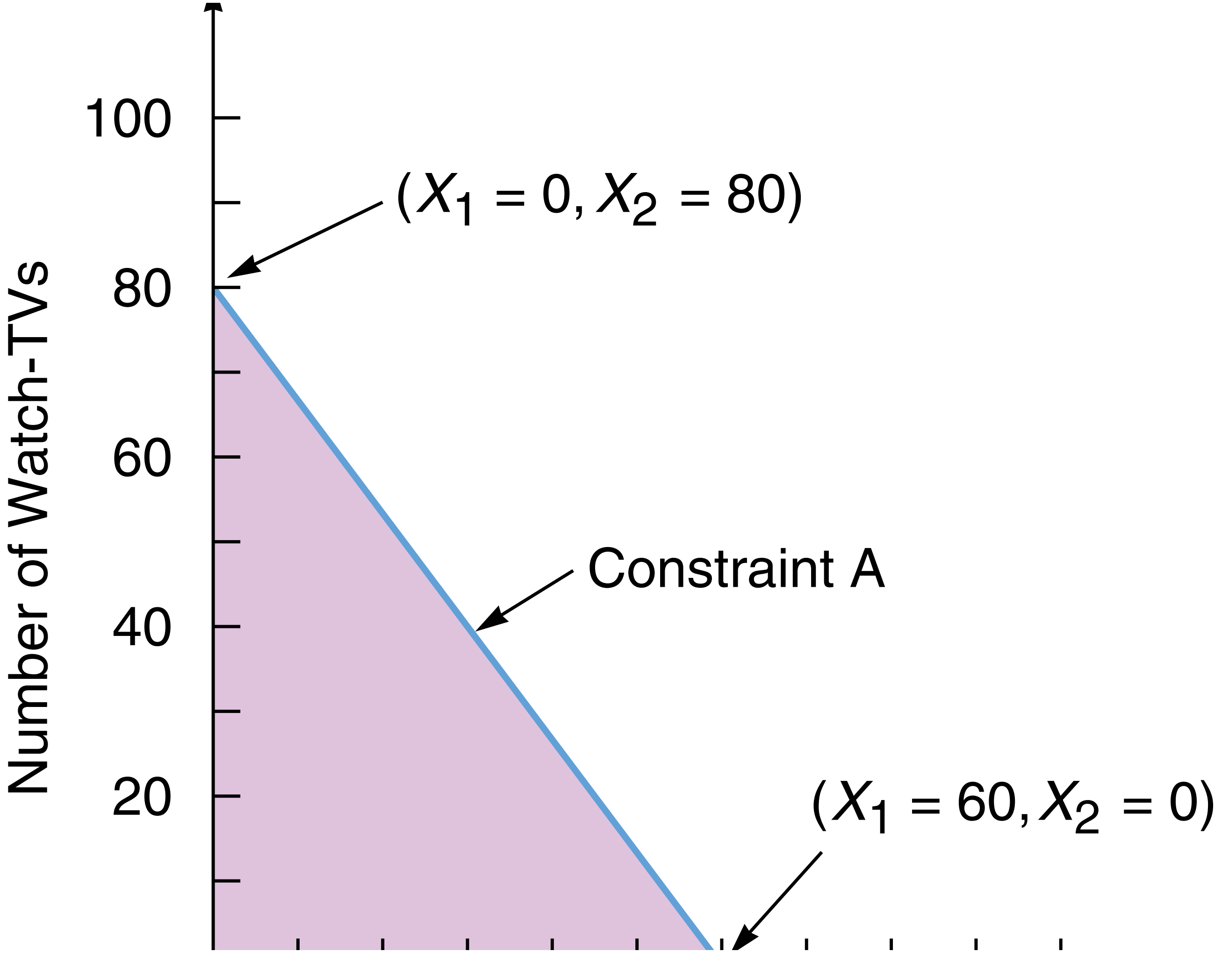
Lineaire optimalisatie : Hoogtelijnen



Lineaire optimalisatie : Hoogtelijnen

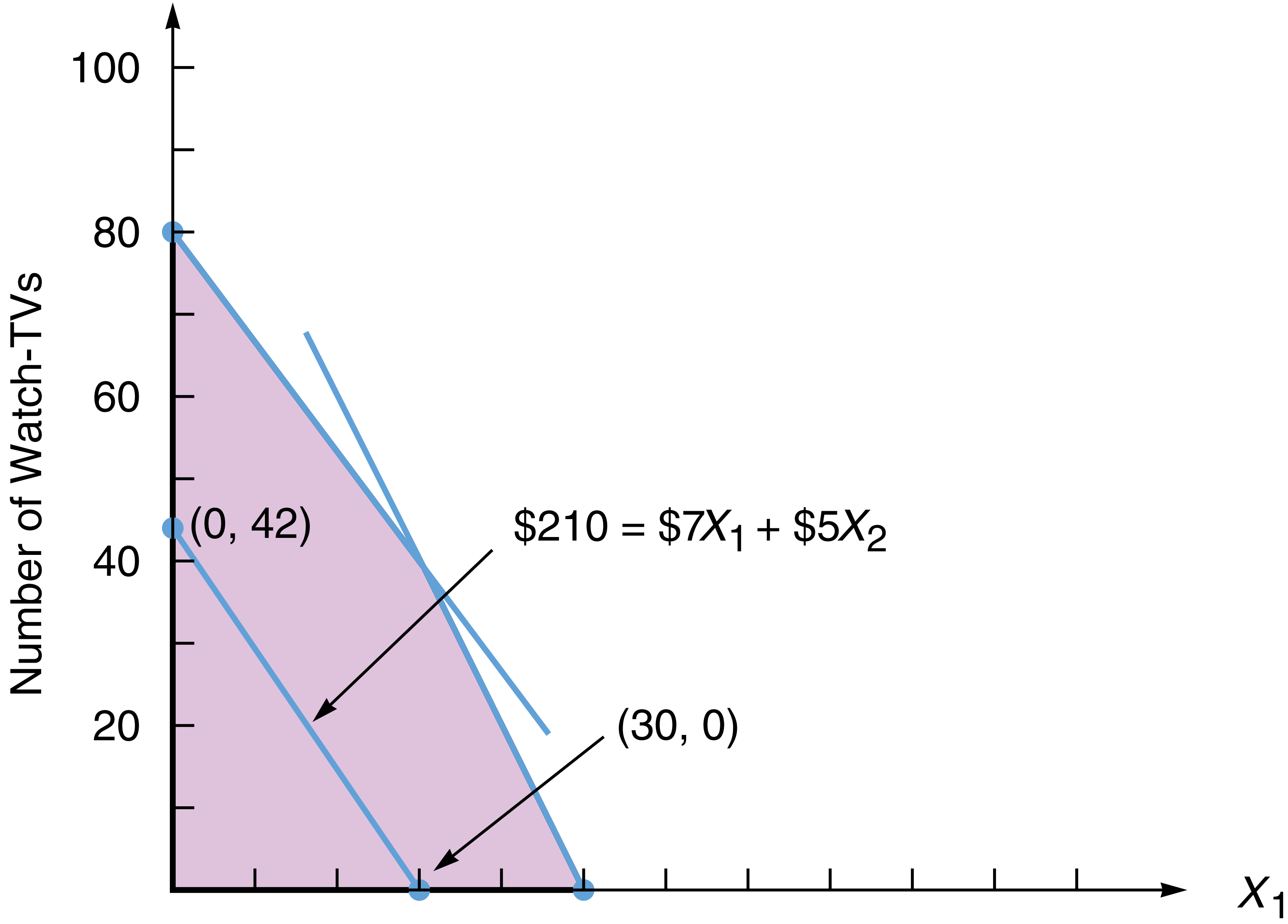
A. Fabricage van elektronica ≤ beschikbare tijd : $4X_1 + 3X_2 \leq 240$

B. Assemblage tijd ≤ beschikbare assemblage tijd: $2X_1 + 1X_2 \leq 100$

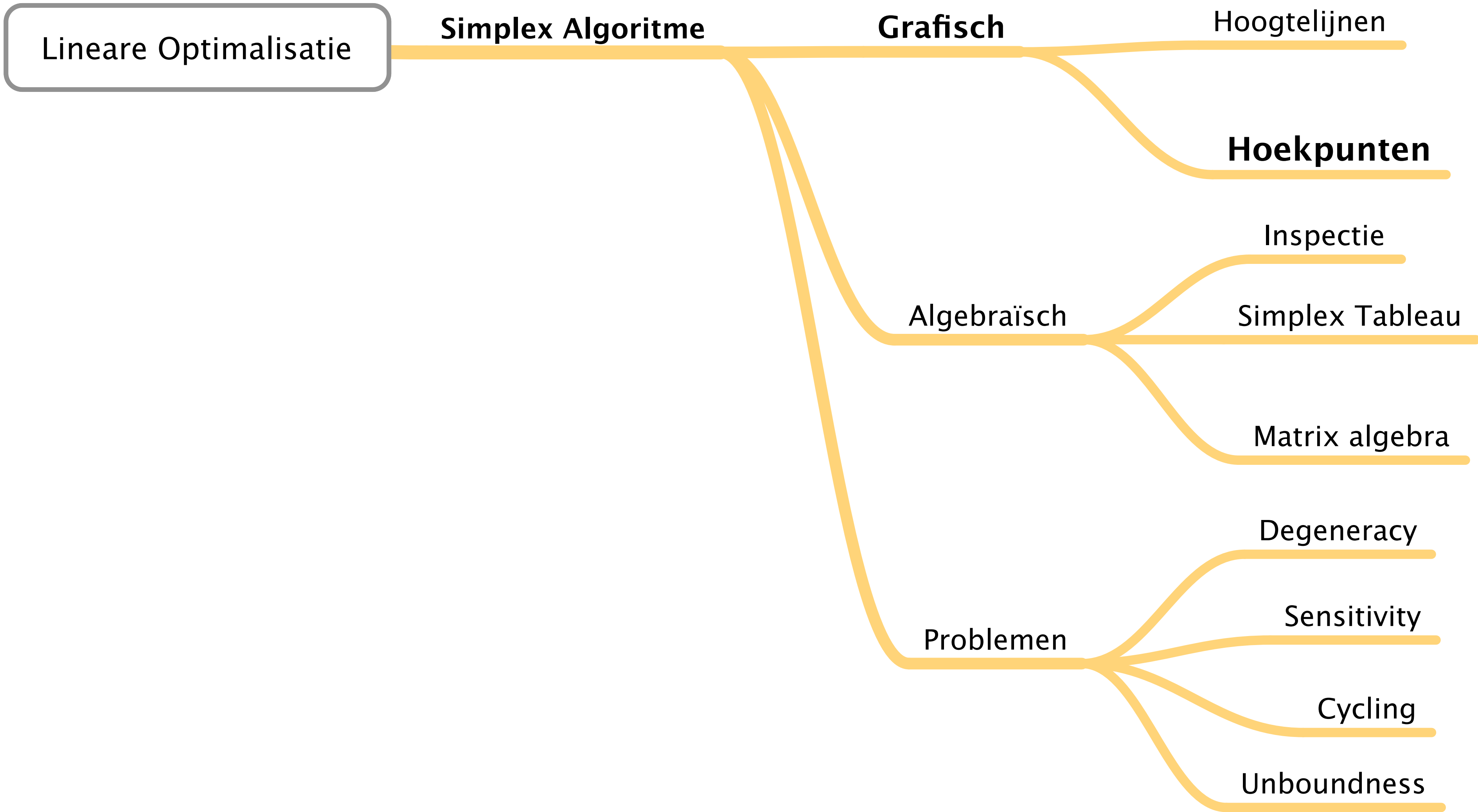


Lineaire optimalisatie : Hoogtelijnen

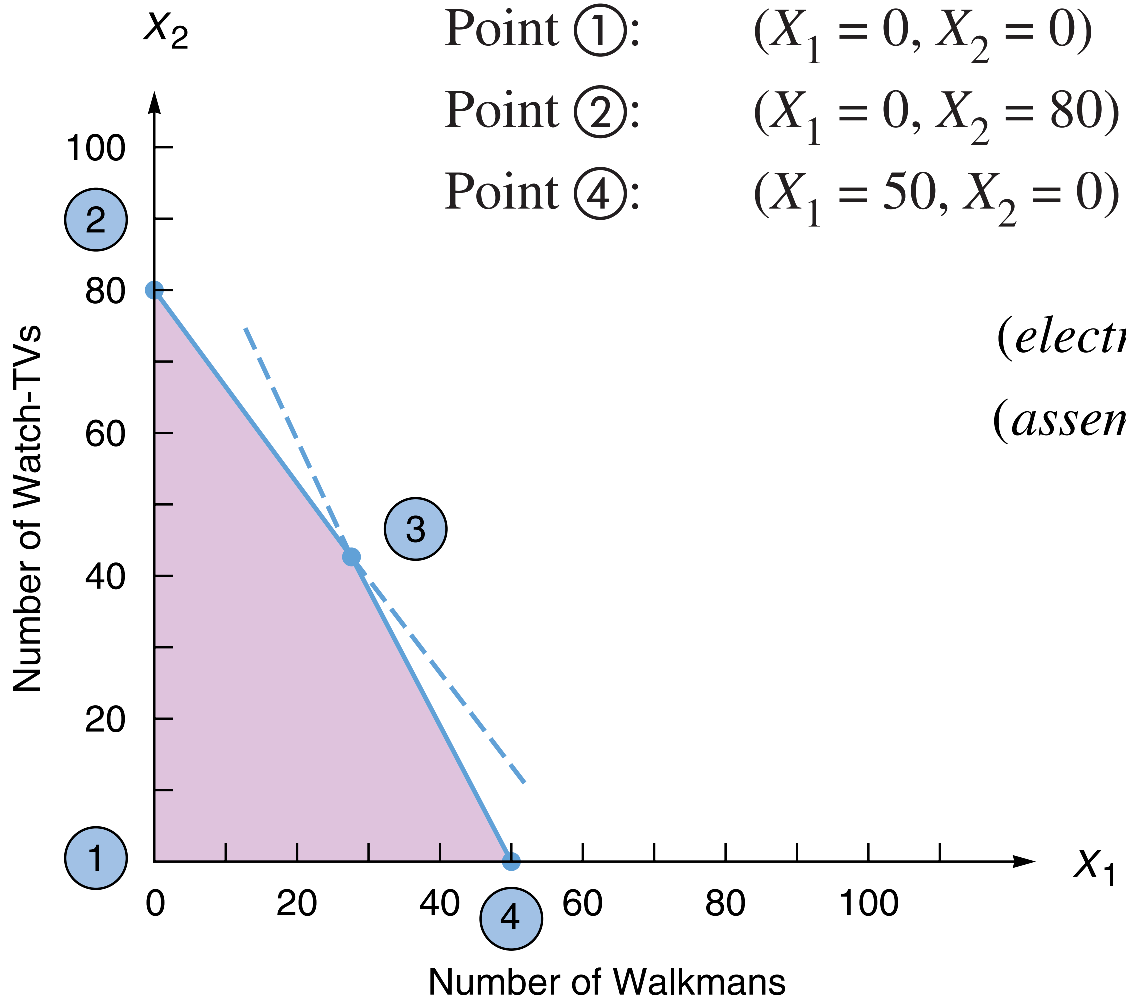
- Objective functie: Winst = $7X_1+5X_2$



Lineaire optimalisatie : Hoekpunten



Lineaire optimalisatie : Hoekpunten



Point ①: $(X_1 = 0, X_2 = 0)$ Profit $\$7(0) + \$5(0) = \$0$

Point ②: $(X_1 = 0, X_2 = 80)$ Profit $\$7(0) + \$5(80) = \$400$

Point ④: $(X_1 = 50, X_2 = 0)$ Profit $\$7(50) + \$5(0) = \$350$

(*electronics time*) $4X_1 + 3X_2 = 240$

(*assembly time*) $2X_1 + 1X_2 = 100$

$$\begin{array}{rrcr} +4X_1 & +3X_2 & = & 240 \\ -4X_1 & -2X_2 & = & -200 \\ \hline & +1X_2 & = & 40 \end{array}$$

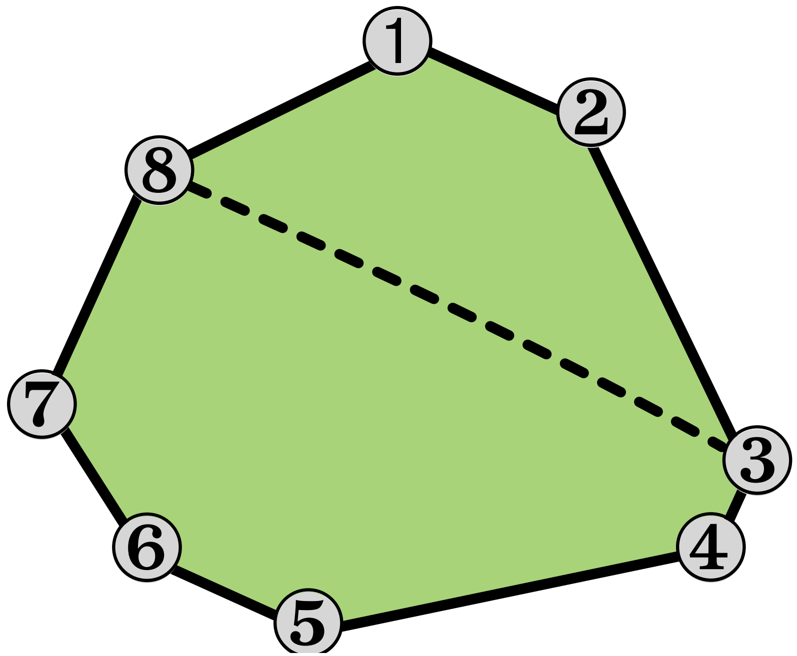
Point ③: $(X_1 = 30, X_2 = 40)$

Profit $= \$7(30) + \$5(40) = \$410$

Lineaire optimalisatie

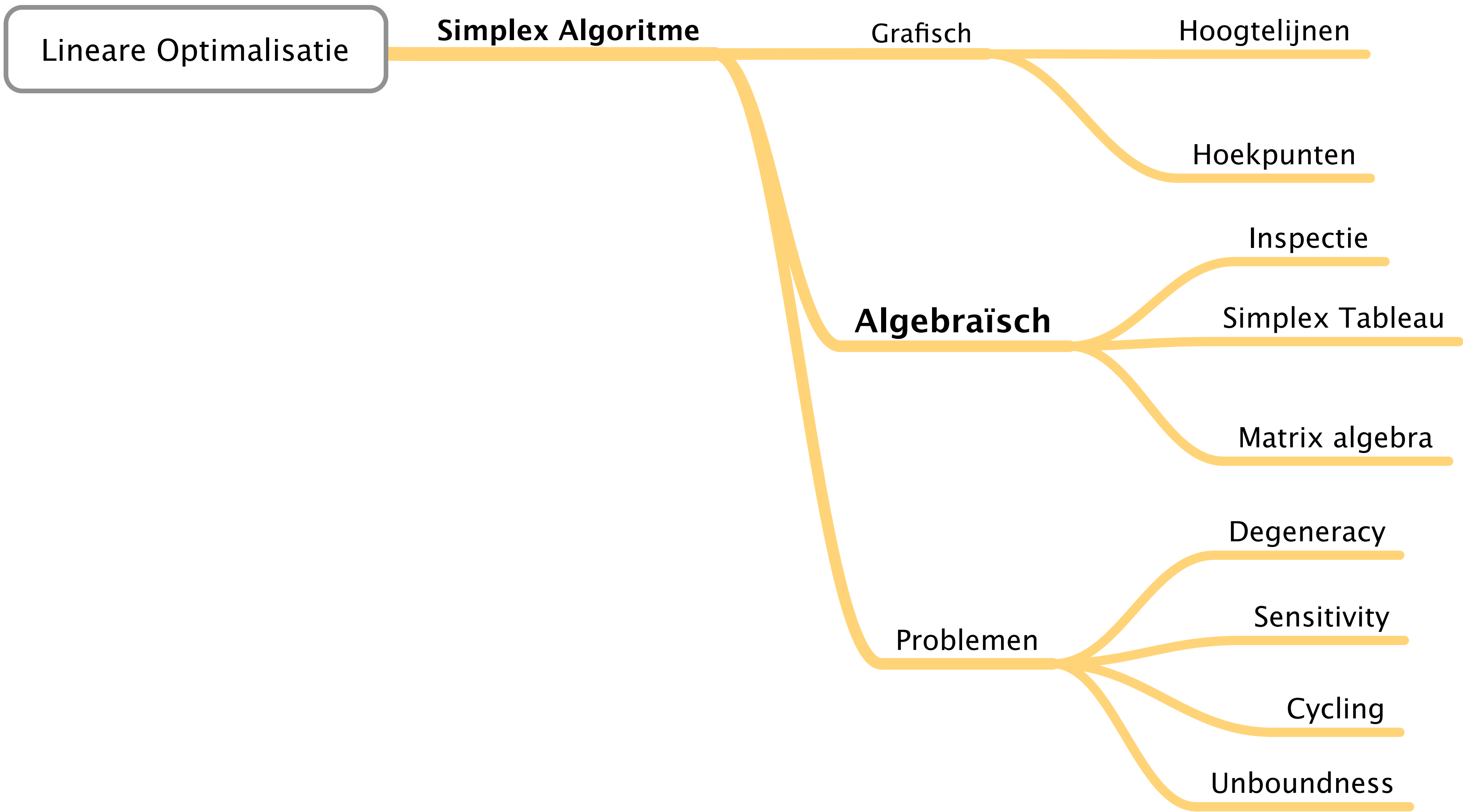
- **Wat valt op ?**
  - Vanaf elk punt in de feasible set is een rechte lijn te trekken naar elk ander punt in de feasible set (de feasible set is convex).
  - De oplossing is aan de rand van de feasible set.
  - De oplossing is op een hoekpunt

    (een hoekpunten is een oplossing die niet te schrijven is als een lineaire combinatie van twee ander punten uit de feasible set)



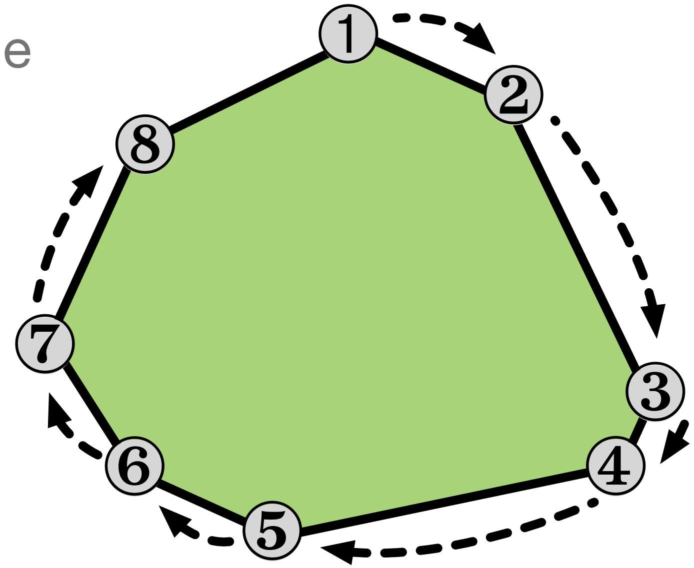
- **Hoe kunnen we dit gebruiken?**

Lineaire optimalisatie : Algebraïsch



Het simplex algoritme

- Het simplex algoritme zoekt de optimale waarde in de feasible set door de randen te onderzoeken.



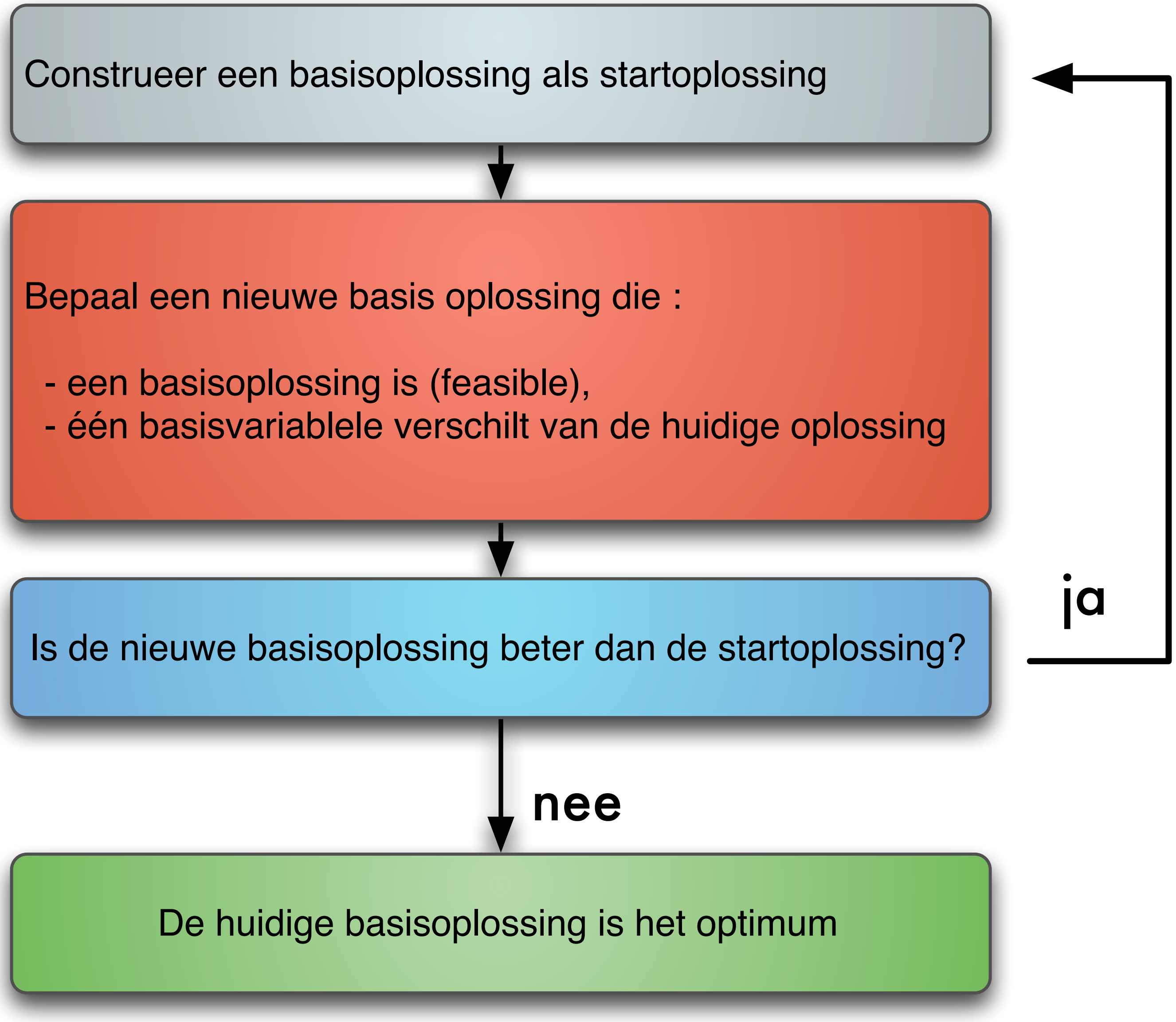
- Op elk hoekpunt wordt de verbinding gezocht naar een nabijgelegen hoekpunt dat de snelste daling van het minimalisatie criterium geeft.

- Het algoritme werkt met een standaard formulering van het minimalisatie probleem:

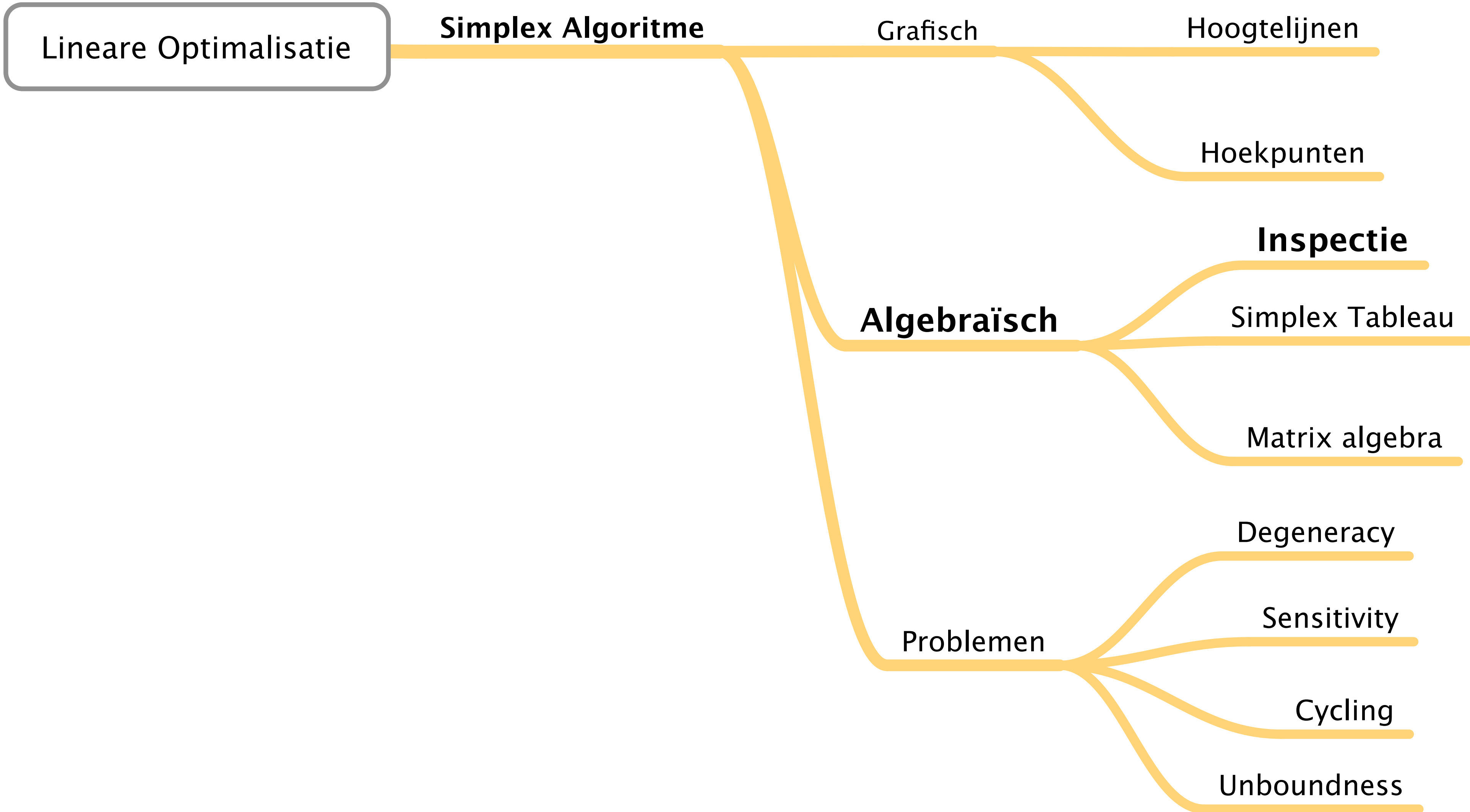
$$
\begin{aligned}
\text{minimize}\; z &= -x_1 - 2x_2 \\
\text{subject to}\quad -2x_1 + x_2 &\leq 2 \\
-x_1 + 2x_2 &\leq 7 \\
x_1 &\leq 3 \\
x_1, x_2 &\geq 0
\end{aligned}
$$

$$
\begin{aligned}
\text{minimize}\quad z &= -x_1 - 2x_2 \\
\text{subject to}\; -2x_1 + x_2 + {\color{blue}x_3} &= 2 \\
-x_1 + 2x_2 + {\color{blue}x_4} &= 7 \\
x_1 + {\color{blue}x_5} &= 3 \\
x_1, x_2, {\color{blue}x_3}, {\color{blue}x_4}, {\color{blue}x_5} &\geq 0
\end{aligned}
$$

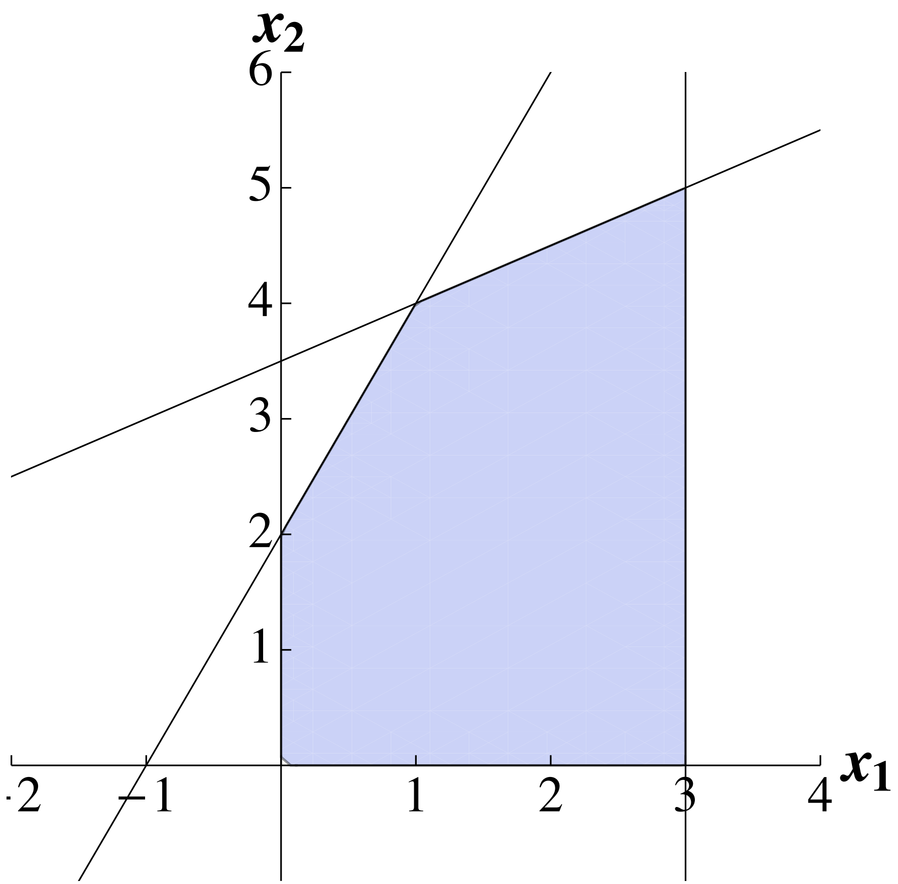
Het simplex algoritme



Het simplex algoritme

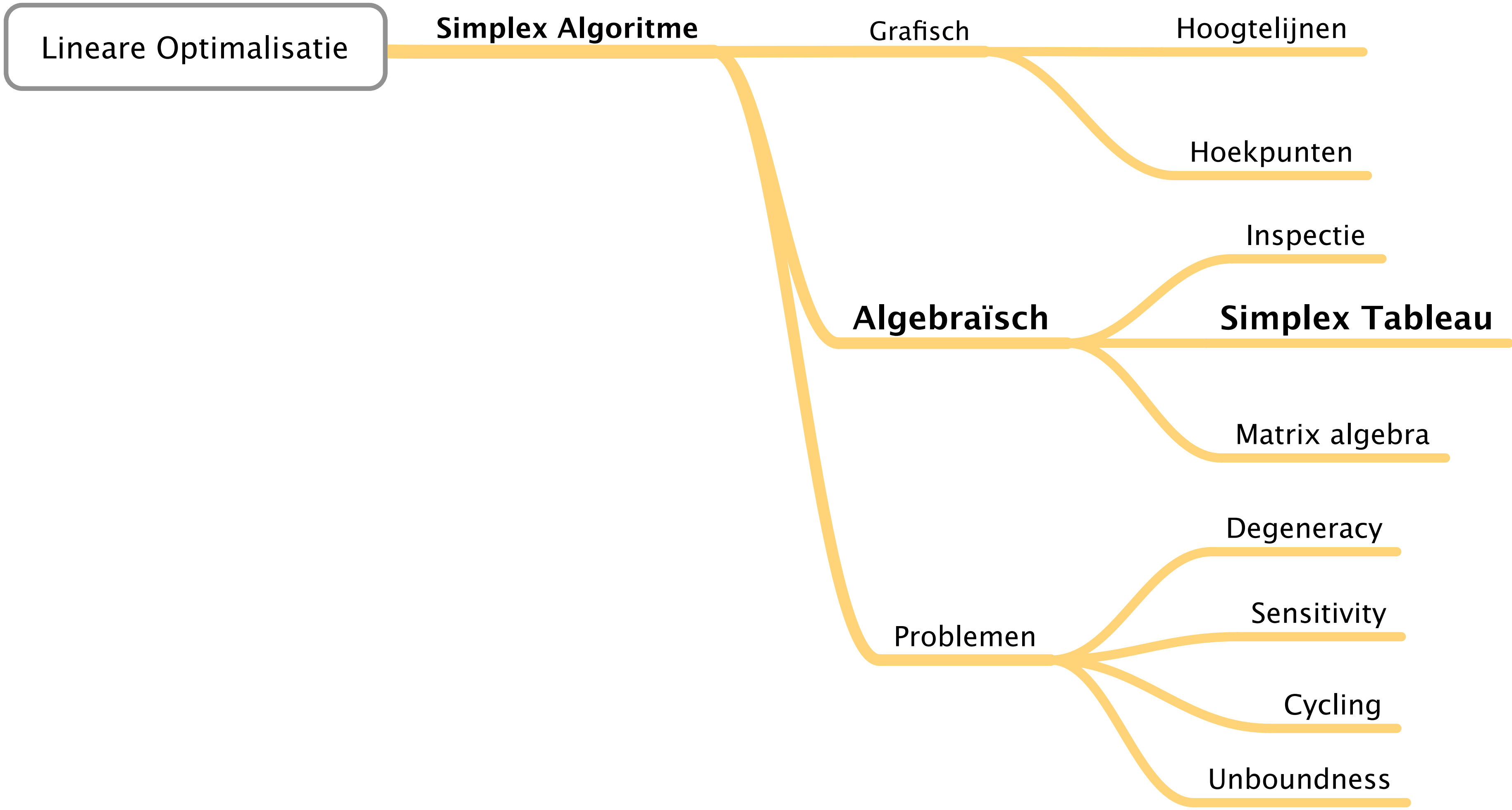


Het simplex algoritme : Inspectie



$$
\begin{aligned}
\text{minimize}\, z = -x_1 - 2x_2 \\
\text{subject to} \quad -2x_1 + x_2 &\leq 2 \\
-x_1 + 2x_2 &\leq 7 \\
x_1 &\leq 3 \\
x_1, x_2 &\geq 0
\end{aligned}
$$

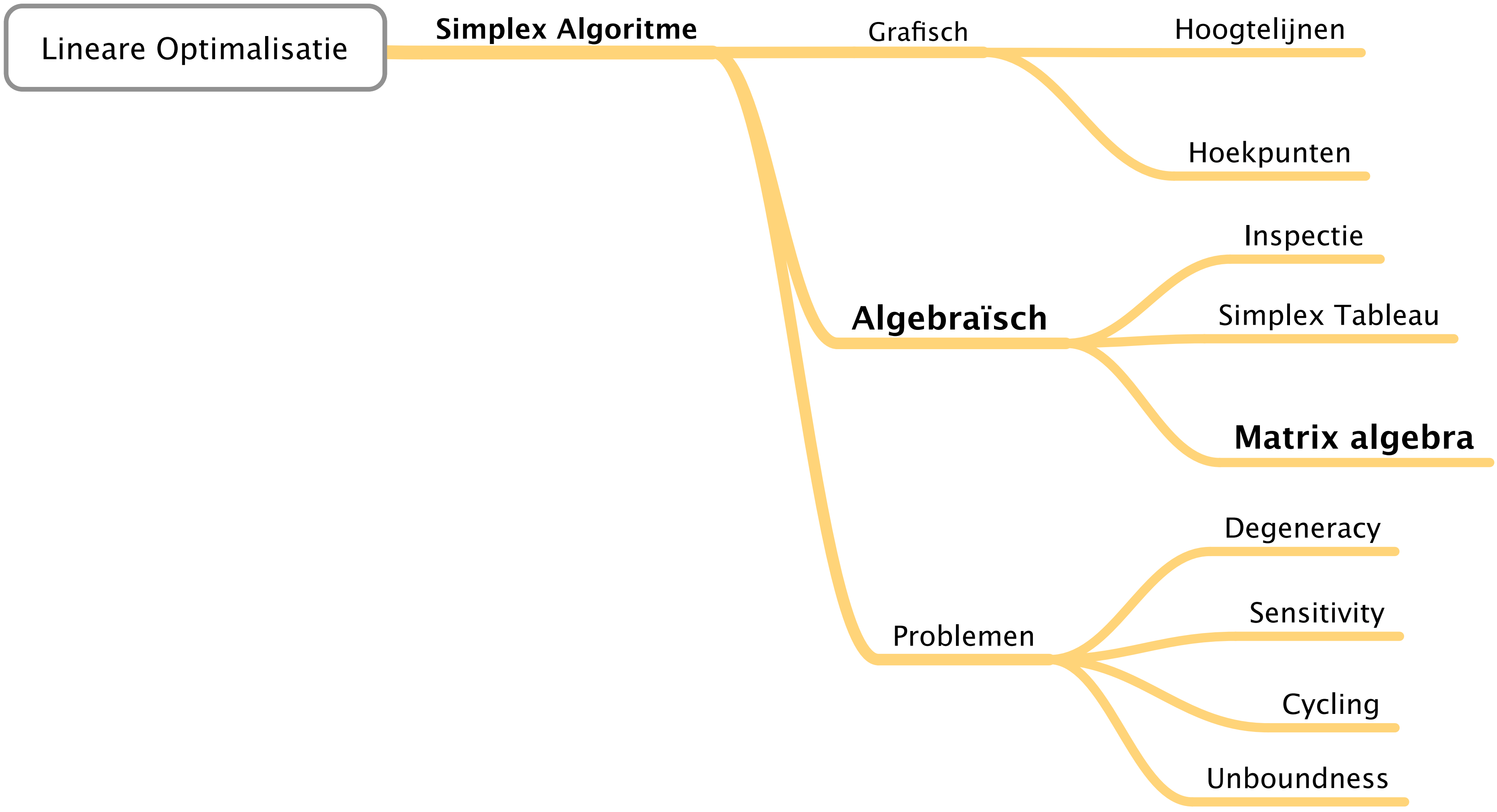
Het simplex algoritme : Simplex Tableau



Het simplex algoritme : Simplex Tableau

$$
\begin{aligned}
\text{minimize}\, z = -x_1 - 2x_2 & \\
\text{subject to} \quad -2x_1 + x_2 &\leq 2 \\
-x_1 + 2x_2 &\leq 7 \\
x_1 &\leq 3 \\
x_1, x_2 &\geq 0
\end{aligned}
$$

Het Simplex Algoritme



Het simplex algoritme

- De algemene vorm

$$\begin{aligned} \text{minimize}\; & z = -x_1 - 2x_2 \\ \text{subject to}\quad & -2x_1 + x_2 \le 2 \\ & -x_1 + 2x_2 \le 7 \\ & x_1 \le 3 \\ & x_1, x_2 \ge 0 \end{aligned}$$

$$\begin{aligned} \text{minimize}\quad & z = -x_1 - 2x_2 \\ \text{subject to}\; & -2x_1 + x_2 + x_3 = 2 \\ & -x_1 + 2x_2 + x_4 = 7 \\ & x_1 + x_5 = 3 \\ & x_1, x_2, x_3, x_4, x_5 \ge 0 \end{aligned}$$

- Kan worden geschreven in vector-matrix notatie

$$\begin{aligned} \text{minimize}\quad & z = \mathbf{c}^{\mathrm{T}}\mathbf{x} \\ \text{subject to}\quad & A\mathbf{x} = \mathbf{b} \\ \text{and}\quad & \mathbf{x} \ge \mathbf{0} \end{aligned}$$

Het simplex algoritme

- In matrix notatie wordt de objective functie

$$z = -x_1 - 2x_2 \Leftrightarrow \mathbf{c}^{\mathrm{T}}\mathbf{x} = \begin{pmatrix} -1 & -2 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix}$$

- De constraint vergelijkingen worden in matrix notatie

$$\begin{aligned} -2x_1 + x_2 + {\color{blue}x_3} &= 2 \\ -x_1 + 2x_2 + {\color{blue}x_4} &= 7 \\ x_1 + {\color{blue}x_5} &= 3 \end{aligned} \Leftrightarrow \mathbf{A}\mathbf{x} = \mathbf{b} \Leftrightarrow \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 2 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 2 \\ 7 \\ 3 \end{pmatrix}$$

- Merk op dat de matrix volle rang heeft

Het simplex algoritme

- Splits de oplossing in twee delen $\mathbf{x} = \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix}$

- Splits de objective functie in een basis deel en een niet-basis deel

$$z = \mathbf{c}_B^{\mathrm{T}}\mathbf{x}_B + \mathbf{c}_N^{\mathrm{T}}\mathbf{x}_N \Leftrightarrow z = \begin{pmatrix} -1 & -2 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 0 & 0 \end{pmatrix} \begin{pmatrix} x_4 \\ x_5 \end{pmatrix}$$

- Splits de constraint matrix A in een basis deel en een niet-basis deel

$$\mathbf{B}\mathbf{x}_b + \mathbf{N}\mathbf{x}_N = \mathbf{b} \Leftrightarrow \begin{pmatrix} -2 & 1 & 1 \\ -1 & 2 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 2 \\ 7 \\ 3 \end{pmatrix}$$

Het simplex algoritme

- De contraint vergelijking kan worden geschreven als

$$\mathbf{B}\mathbf{x}_b + \mathbf{N}\mathbf{x}_N = \mathbf{b}$$

- Waarna xb kan worden bepaald

$$\begin{aligned}\mathbf{B}\mathbf{x}_b + \mathbf{N}\mathbf{x}_N &= \mathbf{b}\\ \mathbf{B}^{-1}\mathbf{B}\mathbf{x}_b + \mathbf{B}^{-1}\mathbf{N}\mathbf{x}_N &= \mathbf{B}^{-1}\mathbf{b}\\ \mathbf{x}_b + \mathbf{B}^{-1}\mathbf{N}\mathbf{x}_N &= \mathbf{B}^{-1}\mathbf{b}\\ \mathbf{x}_b &= \mathbf{B}^{-1}\mathbf{b} - \mathbf{B}^{-1}\mathbf{N}\mathbf{x}_N\end{aligned}$$

- Een van de variabelen in xn wordt nu verhoogt

Het simplex algoritme

- De matrix B moet vierkant zijn en volle rang hebben

$$\mathbf{B}^{-1}\mathbf{B}\mathbf{x}_b = \begin{pmatrix} 0 & 0 & 1 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ 1 & -\frac{1}{2} & \frac{3}{2} \end{pmatrix} \begin{pmatrix} -2 & 1 & 1 \\ -1 & 2 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

- De vector xb wordt dan

$$\mathbf{x}_b = \mathbf{B}^{-1}\mathbf{b} - \mathbf{B}^{-1}\mathbf{N}\mathbf{x}_N$$

$$= \begin{pmatrix} 0 & 0 & 1 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ 1 & -\frac{1}{2} & \frac{3}{2} \end{pmatrix} \begin{pmatrix} 2 \\ 7 \\ 3 \end{pmatrix} - \begin{pmatrix} 0 & 0 & 1 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ 1 & -\frac{1}{2} & \frac{3}{2} \end{pmatrix} \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_4 \\ x_5 \end{pmatrix}$$

- Verhoog of x4 of x5 om bij het volgende hoekpunt te komen

Het simplex algoritme

- De objective functie is gegeven door

$$z = \mathbf{c}_B^{\mathrm{T}} \mathbf{x}_B + \mathbf{c}_N^{\mathrm{T}} \mathbf{x}_N$$

- Invullen van $\mathbf{x}_b = \mathbf{B}^{-1}\mathbf{b} - \mathbf{B}^{-1}\mathbf{N}\mathbf{x}_N$ levert

$$z = \mathbf{c}_B^{\mathrm{T}} \left(\mathbf{B}^{-1}\mathbf{b} - \mathbf{B}^{-1}\mathbf{N}\mathbf{x}_N\right) + \mathbf{c}_N^{\mathrm{T}} \mathbf{x}_N$$

- hetgeen kan worden geschreven als

$$z = \mathbf{c}_B^{\mathrm{T}} \mathbf{B}^{-1}\mathbf{b} + \left(\mathbf{c}_N^{\mathrm{T}} - \mathbf{c}_B^{\mathrm{T}} \mathbf{B}^{-1}\mathbf{N}\right) \mathbf{x}_N$$

- waarna de vector

$$\mathbf{y} = \left(\mathbf{c}_B^{\mathrm{T}} \mathbf{B}^{-1}\right)^{\mathrm{T}} = \mathbf{B}^{-\mathrm{T}} \mathbf{c}_B$$

- wordt gedefinieerd als de vector van simplex multipliers

$$z = \mathbf{y}^{\mathrm{T}}\mathbf{b} + \left(\mathbf{c}_N^{\mathrm{T}} - \mathbf{y}^{\mathrm{T}}\mathbf{N}\right) \mathbf{x}_N$$

Het simplex algoritme : Voorbeeld 1

- Met A,b en c gedefinieerd als

$$A = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 2 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix} \; b = \begin{pmatrix} 2 \\ 7 \\ 3 \end{pmatrix} \; c = \begin{pmatrix} -1 \\ -2 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

- Kiezen we $x_B = (x_1, x_2, x_3)^T$ en $x_N = (x_4, x_5)^T$.

- Dan volgt

$$B = \begin{pmatrix} -2 & 1 & 1 \\ -1 & 2 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad B^{-1} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ 1 & -\frac{1}{2} & \frac{3}{2} \end{pmatrix}, \quad N = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix},$$

- en

$$c_B^T = (-1, -2, 0), \; c_N^T = (0, 0).$$

Het simplex algoritme : Voorbeeld 1

- De waarden van de variabelen zijn

$$x_B = \hat{b} = B^{-1}b = \begin{pmatrix} 3 \\ 5 \\ 3 \end{pmatrix}$$

- met $x_N = (0, 0)^T$. volgt dat

$$y^T = c_B^T B^{-1} = (\,0 \quad -1 \quad -2\,),$$

- en

$$\hat{z} = y^T b = -13.$$

Het simplex algoritme : Voorbeeld 1

- Als $x_N \neq 0,$ dan

- Waarna volgt

$$x_B = B^{-1}b - B^{-1}Nx_N = \begin{pmatrix} 3 \\ 5 \\ 3 \end{pmatrix} - \begin{pmatrix} 0 & 1 \\ \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & \frac{3}{2} \end{pmatrix} \begin{pmatrix} x_4 \\ x_5 \end{pmatrix},$$

- Invullen levert

$$z = y^T b + (c_N^T - y^T N)x_N = -13 + \begin{pmatrix} 1 & 2 \end{pmatrix} \begin{pmatrix} x_4 \\ x_5 \end{pmatrix}.$$

- De basis is optimaal!

Het simplex algoritme in drie stappen

- Bereken de simplex multipliers $y^T = \mathbf{c}_B^{\mathrm{T}}\mathbf{B}^{-1}$
- Bereken de coëfficiënten $\hat{\mathbf{c}}_N^{\mathrm{T}} = \mathbf{c}_N^{\mathrm{T}} - \mathbf{y}^{\mathrm{T}} N$
- Als $\hat{\mathbf{c}}_N^{\mathrm{T}} \geq 0$ dan kan de objective functie niet verder afnemen
- Als niet geld dat $\hat{\mathbf{c}}_N^{\mathrm{T}} \geq 0$ zoek dan de kleinste $\hat{c}_t < 0$

- Bereken de constraint coëfficiënten van de entering variabele $\hat{\mathbf{A}}_t^{\mathrm{T}} = \mathbf{B}^{-1}\mathbf{A}_t^{\mathrm{T}}$
- vind de index s die voldoet aan $$\frac{\hat{b}_s}{\hat{a}_{s,t}} = \min_{l \leq i \leq m} \left\{ \frac{\hat{b}_s}{\hat{a}_{s,t}} : \hat{a}_{i,t} > 0 \right\}$$

- Update matrices B en N en basic variabelen $\hat{\mathbf{x}}_b^{\mathrm{T}}$

Het simplex algoritme : Voorbeeld 2 / iteratie 1

- De matrix A en de vectoren b en c zijn

$$A = \begin{pmatrix} -2 & 1 & 1 & 0 & 0 \\ -1 & 2 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 2 \\ 7 \\ 3 \end{pmatrix} \quad c = \begin{pmatrix} -1 \\ -2 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

- De keuze voor de set basis variabelen is

$$x_B = (x_3, x_4, x_5)^T, \; x_N = (x_1, x_2)^T,$$

- en dus $B = I = B^{-1} \quad c_B^T = (0, 0, 0), c_N^T = (-1, -2),$

$$N = \begin{pmatrix} -2 & 1 \\ -1 & 2 \\ 1 & 0 \end{pmatrix}.$$

Het simplex algoritme : Voorbeeld 2 / iteratie 1

- Het volgt dat

$$x_B = \hat{b} = B^{-1}b = (2, 7, 3)^T.$$

- met deze basis zijn

$$y^T = c_B^T B^{-1} = \begin{pmatrix} 0 & 0 & 0 \end{pmatrix} \qquad \hat{c}_N^T = c_N^T - y^T N = \begin{pmatrix} -1 & -2 \end{pmatrix}$$

- De basis is niet optimaal, voor de ratiotest selecteren we de kolom

$$\hat{A}_2 = B^{-1}A_2 = \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix}, \quad \frac{\hat{b}_1}{\hat{a}_{1,2}} = 2 \qquad \frac{\hat{b}_2}{\hat{a}_{2,2}} = \frac{7}{2}.$$

- zodat de eerste basis variabele vertrekt x2 en x3 wisselen

Het simplex algoritme : Voorbeeld 2 / iteratie 2

- In de volgende iteratie krijgen we $x_B = (x_2, x_4, x_5)^T,\ x_N = (x_1, x_3)^T$,
- met

$$B = \begin{pmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad B^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad N = \begin{pmatrix} -2 & 1 \\ -1 & 0 \\ 1 & 0 \end{pmatrix},$$

$$c_B^T = (-2, 0, 0),\ c_N^T = (-1, 0).$$

- en volgt

$$\begin{aligned} x_B &= \hat{b} = B^{-1}b = (\,2 \quad 3 \quad 3\,)^T \\ y^T &= c_B^T B^{-1} = (\,-2 \quad 0 \quad 0\,) \\ \hat{c}_N^T &= c_N^T - y^T N = (\,-5 \quad 2\,)\,. \end{aligned}$$

- Deze basis is niet optimaal

Het simplex algoritme : Voorbeeld 2 / iteratie 2

- Het eerste element van de gereduceerde kostenfunctie is negatief

$$\hat{c}_N^T = c_N^T - y^T N = \begin{pmatrix} -5 & 2 \end{pmatrix}.$$

- Het optimum is dus nog niet gevonden, uit de ratio test volgt

$$\hat{A}_1 = B^{-1} A_1 = \begin{pmatrix} -2 \\ 3 \\ 1 \end{pmatrix} \quad \begin{aligned} \hat{b}_2/\hat{a}_{2,1} &= 1 \\ \hat{b}_3/\hat{a}_{3,1} &= 3 \end{aligned}$$

- Zodat de tweede basis variabele vertrekt (x4)

$$x_B = (x_2, x_1, x_5)^T, x_N = (x_3, x_4)^T,$$

Het simplex algoritme : Voorbeeld 2 / iteratie 3

- Bij iteratie 3 hebben we

$$x_B = (x_2, x_1, x_5)^T, x_N = (x_3, x_4)^T,$$

- met

$$B = \begin{pmatrix} 1 & -2 & 0 \\ 2 & -1 & 0 \\ 0 & 1 & 1 \end{pmatrix}, \quad B^{-1} = \begin{pmatrix} -\frac{1}{3} & \frac{2}{3} & 0 \\ -\frac{2}{3} & \frac{1}{3} & 0 \\ \frac{2}{3} & -\frac{1}{3} & 1 \end{pmatrix}, \quad N = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix},$$

- en

$$c_B^T = (-2, -1, 0) \quad c_N^T = (0, 0).$$

- zodat

$$x_B = \hat{b} = B^{-1}b = (\,4 \quad 1 \quad 2\,)^T$$
$$y^T = c_B^T B^{-1} = (\,\tfrac{4}{3} \quad -\tfrac{5}{3} \quad 0\,)$$
$$\hat{c}_N^T = c_N^T - y^T N = (\,-\tfrac{4}{3} \quad \tfrac{5}{3}\,).$$

Het simplex algoritme : Voorbeeld 2 / iteratie 3

- De basis is niet optimaal omdat $\hat{c}_N^T = c_N^T - y^T N = \left(-\frac{4}{3} \quad \frac{5}{3} \right).$
- Met x3 als enetering variabele volgt

$$\hat{A}_3 = B^{-1} A_3 = \begin{pmatrix} -\frac{1}{3} \\ -\frac{2}{3} \\ \frac{2}{3} \end{pmatrix} \qquad \hat{b}_3/\hat{a}_{3,1} = 3.$$

- x5 is de vertrekkende variabele

$$x_B = (x_2, x_1, x_3)^T, x_N = (x_4, x_5)^T,$$

Het simplex algoritme : Voorbeeld 2 / iteratie 4

- Iteratie 4 : $x_B = (x_2, x_1, x_3)^T, x_N = (x_4, x_5)^T,$

$$B = \begin{pmatrix} 1 & -2 & 1 \\ 2 & -1 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \quad B^{-1} = \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 1 \\ 1 & -\frac{1}{2} & \frac{3}{2} \end{pmatrix}, \quad N = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix},$$

$$c_B^T = (-2, -1, 0), \quad c_N^T = (0, 0).$$

$$x_B = \hat{b} = B^{-1}b = (\,5 \quad 3 \quad 3\,)^T$$

$$y^T = c_B^T B^{-1} = (\,0 \quad -1 \quad -2\,)$$

$$\hat{c}_N^T = c_N^T - y^T N = (\,1 \quad 2\,)\,.$$

- Deze basis is optimaal

Implementatie Mathematica

```
mA = {{-2, 1, 1, 0, 0}, {-1, 2, 0, 1, 0}, {1, 0, 0, 0, 1}};
b =  {{2}, {7}, {3}};
c = {{-1}, {-2}, {0}, {0}, {0}};

mB = IdentityMatrix[3];
mN = {{-2, 1}, {-1, 2}, {1, 0}};

cb = {{0}, {0}, {0}};
cn = {{-1}, {-2}};
```

```
tableau = MatrixForm[Join[
    Join[Join[Transpose[cb], Transpose[cn], 2], {{0}} , 2] ,
    Join[Join[mB             , mN              , 2],   b   , 2]]];
```

Stap 1 : De optimum test

- Bereken de simplex multipliers $y^T = \mathbf{c}_B^{\mathrm{T}}\mathbf{B}^{-1}$
- Bereken de coëfficiënten $\hat{\mathbf{c}}_N^{\mathrm{T}} = \mathbf{c}_N^{\mathrm{T}} - \mathbf{y}^{\mathrm{T}}N$
  - Als $\hat{\mathbf{c}}_N^{\mathrm{T}} \geq 0$ dan kan de objective functie niet verder afnemen
  - Als niet geld dat $\hat{\mathbf{c}}_N^{\mathrm{T}} \geq 0$ zoek dan de kleinste $\hat{c}_t < 0$

```
y = Transpose[cb].Inverse[mB];
cnh = Transpose[cn] - Transpose[cb].Inverse[mB].mN;
If[
      Length[Cases[Flatten[cnh], x_ /; x < 0]] != 0,
      optimal = False;
      indexenteringvariable = Part[Position[Flatten[cnh], x_ /; x == Min[cnh]], 1, 1];
  ,
      optimal = True;
 ];
```

Stap 2 : Bepaal de vertrekkende variabele

- Bereken de constraint coëfficiënten van de entering variabele $\hat{\mathbf{A}}_t^{\mathrm{T}} = \mathbf{B}^{-1}\mathbf{A}_t^{\mathrm{T}}$
- vind de index s die voldoet aan $$\frac{\hat{b}_s}{\hat{a}_{s,t}} = \min_{l \leq i \leq m} \left\{ \frac{\hat{b}_s}{\hat{a}_{s,t}} \ : \ \hat{a}_{i,t} > 0 \right\}$$

```
ath = Inverse[mB].mA[[All, indexenteringvariable]];
indexlist = Position[Flatten[ath], x_ /; x > 0];
indexleavingvariable = 1;
Do[
    If[
         Part[b, indexlist[[i]]] / Part[ath, indexlist[[i]]] <
         Part[b, indexlist[[1]]] / Part[ath, indexlist[[1]]]
       , indexleavingvariable = indexlist[[i]];
        ];
   , {i, 1, Length[indexlist]}
  ]
```

Stap 3 : Update Matrices

- Update matrices B en N en basic variabelen $\hat{\mathbf{x}}_b^{\mathrm{T}}$

```
mBnew = mB; mNnew = mN; cbnew = cb; cnnew = cn;
mBnew[[All, indexleavingvariable]] = mN[[All, indexenteringvariable]];
mNnew[[All, indexenteringvariable]] = mB[[All, indexleavingvariable]];
cbnew[[indexleavingvariable]] = cn[[indexenteringvariable]];
cnnew[[indexenteringvariable]] = cb[[indexleavingvariable]];
mB = mBnew; mN  = mNnew;
cb = cbnew; cn = cnnew;
mA = Join[mB, mN, 2];
```

Inverse van een matrix

- De inverse van een (vierkante) matrix $\mathbf{A}$ wordt gegeven als

$$\mathbf{A}^{-1} = \frac{1}{|\mathbf{A}|}\left(\mathbf{C}^{\mathrm{T}}\right)_{ij} = \frac{1}{|\mathbf{A}|}\left(\mathbf{C}_{ji}\right) = \frac{1}{|\mathbf{A}|}\begin{pmatrix} \mathbf{C}_{11} & \mathbf{C}_{21} & \cdots & \mathbf{C}_{n1} \\ \mathbf{C}_{12} & \mathbf{C}_{22} & \cdots & \mathbf{C}_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_{1n} & \mathbf{C}_{2n} & \cdots & \mathbf{C}_{nn} \end{pmatrix}$$

- Waarbij de cofactoren gegeven worden door

$$\mathbf{C}_{ij} = (-1)^{i+j}\mathbf{M}_{ij}$$

- Waarin $\mathbf{M}_{ij}$ is gegeven als de minor van het element a_{ij}

Determinant van een matrix

- Bereken de determinant van matrix A

$$A = \begin{bmatrix} -2 & 2 & -3 \\ -1 & 1 & 3 \\ 2 & 0 & -1 \end{bmatrix}.$$

- Kies de rij of kolom met het hoogste aantal nullen

$$A = \begin{bmatrix} -2 & 2 & -3 \\ -1 & 1 & 3 \\ 2 & 0 & -1 \end{bmatrix}.$$

- Bereken de determinant door het optellen van cofactors

$$A = \begin{bmatrix} -2 & 2 & -3 \\ -1 & 1 & 3 \\ 2 & 0 & -1 \end{bmatrix}. \quad A = \begin{bmatrix} -2 & 2 & -3 \\ -1 & 1 & 3 \\ 2 & 0 & -1 \end{bmatrix}.$$

$$\begin{aligned} \det(A) &= (-1)^{1+2} \cdot 2 \cdot \det \begin{bmatrix} -1 & 3 \\ 2 & -1 \end{bmatrix} + (-1)^{2+2} \cdot 1 \cdot \det \begin{bmatrix} -2 & -3 \\ 2 & -1 \end{bmatrix} \\ &= (-2) \cdot ((-1) \cdot (-1) - 2 \cdot 3) + 1 \cdot ((-2) \cdot (-1) - 2 \cdot (-3)) \\ &= (-2) \cdot (-5) + 8 = 18 \end{aligned}$$

Inverse van een matrix : Minor / Cofactor

- Van de 3x3 matrix $\mathbf{A}$

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

- wordt de minor bepaald door de determinant van de matrix die overblijft nadat de rij en de kolom van het element a_{ij} verwijderd zijn

$$M_{23} = \begin{vmatrix} a_{11} & a_{12} & \\ & & \\ a_{31} & a_{32} & \end{vmatrix} \Longrightarrow M_{23} = \begin{vmatrix} a_{11} & a_{12} \\ a_{31} & a_{32} \end{vmatrix} = a_{11}a_{32} - a_{31}a_{12}$$

- De cofactor van het element 2,3 wordt dan

$$C_{23} = (-1)^{2+3}(M_{23}) \Longrightarrow \; C_{23} = (-1)^5(a_{11}a_{32} - a_{31}a_{12})$$
$$= a_{31}a_{12} - a_{11}a_{32}.$$

Matrix inverse : 2x2 matrix

- Voor een matrix met dimensies 2x2 wordt de inverse gegeven door

$$\mathbf{A}^{-1} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{\det(\mathbf{A})} \begin{bmatrix} d & -c \\ -b & a \end{bmatrix}^{T} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}.$$

- Voorvermenigvuldigen van $\mathbf{A}$ met de inverse van $\mathbf{A}$ geeft

$$\begin{aligned} \mathbf{A}^{-1}\mathbf{A} &= \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} \begin{bmatrix} a & b \\ c & d \end{bmatrix} \\ &= \frac{1}{ad - bc} \begin{bmatrix} da - bc & db - bd \\ -ca + ac & -cb + ad \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{aligned}$$

Matrix inverse : 3x3 matrix

- Voor een 3x3 matrix wordt de inverse

$$\mathbf{A}^{-1} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & k \end{bmatrix}^{-1} = \frac{1}{\det(\mathbf{A})} \begin{bmatrix} A & B & C \\ D & E & F \\ G & H & K \end{bmatrix}^T = \frac{1}{\det(\mathbf{A})} \begin{bmatrix} A & D & G \\ B & E & H \\ C & F & K \end{bmatrix}$$

- Met als determinant

$$\det(\mathbf{A}) = a(ek - fh) - b(kd - fg) + c(dh - eg).$$

- De cofactors worden

$$\begin{aligned} A &= (ek - fh) & D &= (ch - bk) & G &= (bf - ce) \\ B &= (fg - dk) & E &= (ak - cg) & H &= (cd - af) \\ C &= (dh - eg) & F &= (gb - ah) & K &= (ae - bd). \end{aligned}$$

- In Mathematica wordt de inverse gegeven door het commando

$$\text{Inverse}[\text{A}]$$

Vragen?

- Jan van Hulzen : j.r.van.hulzen@hva.nl
- Liv Harkes : l.harkes@hva.nl