Universiteit Utrecht
Departement Informatica

Uitwerking Tussentoets Optimalisering 20 december 2016

Opgave 1.

Beschouw het volgende lineair programmeringsprobleem:

$$
\begin{array}{lllrcrcrcrcl}
\text{(P)} & \text{Minimaliseer} & z = & x_1 & - & 2x_2 & + & x_3 & - & 2x_4 & & \\
& \text{o.v.} & & x_1 & - & 2x_2 & + & 2x_3 & - & x_4 & \geq & 5 \\
& & & x_1 & - & x_2 & - & x_3 & + & x_4 & \leq & 8 \\
& & & x_1 & + & x_2 & - & 2x_3 & + & x_4 & \leq & -4 \\
& & & x_1 & , & x_2 & , & x_3 & , & x_4 & \geq & 0
\end{array}
$$

Laat x_5, x_6, x_7 de spelingsvariabelen zijn. Geef het starttableau van de eerste fase dat gebruikt kan worden om een TBO te vinden. Nadat je het begintableau hebt bepaald **hoef je niet te itereren**.

Voeg eerst de spelingsvariabelen toe en zorg dat de rechterkant ≥ 0 wordt; hiertoe moet je de derde vergelijking met -1 vermenigvuldigen. Dit levert de volgende vergelijkingen op

$$
\begin{aligned}
&x_1 - 2x_2 + 2x_3 - x_4 - x_5 = 5 \\
&x_1 - x_2 - x_3 + x_4 + x_6 = 8 \\
&-x_1 - x_2 + 2x_3 - x_4 - x_7 = 4
\end{aligned}
$$

Wanneer je dit in matrixnotatie $\mathbf{Ax} = \mathbf{b}$ zet, dan zie je dat de eerste en derde eenheidsvector in $\mathbf{A}$ ontbreken. Voeg daarom de kunstmatige variabelen x_8 en x_9 toe aan respectievelijk de eerste en derde vergelijking. Dit geeft de vergelijkingen

$$
\begin{aligned}
&x_1 - 2x_2 + 2x_3 - x_4 - x_5 + x_8 = 5 \\
&x_1 - x_2 - x_3 + x_4 + x_6 = 8 \\
&-x_1 - x_2 + 2x_3 - x_4 - x_7 + x_9 = 4
\end{aligned}
$$

De doelstellingsfunctie is: minimaliseer $z' = x_8 + x_9$; dit levert de doelstellingsfunctie-vergelijking $z' - x_8 - x_9 = 0$ op. Breng dit in het tableau, en zorg ervoor dat het tableau de goede vorm krijgt (omdat x_8 en x_9 basisvariabelen zijn moet er een nul op de nulde rij komen onder die twee); in het onderstaande tableau is de bovenste rij een hulprij. Het begintableau van de eerste fase wordt

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	RHS
0	0	0	0	0	0	0	-1	-1	0
0	-3	4	-2	-1	0	-1	0	0	9
1	-2	2	-1	-1	0	0	1	0	5
1	-1	-1	1	0	1	0	0	0	8
-1	-1	2	-1	0	0	-1	0	1	4

Opgave 2

Beschouw het volgende probleem, dat we het ROOSTERPROBLEEM zullen noemen. Er zijn n werknemers, die allen m van de perioden $1, \ldots, T$ aan het werk gaan. Om de arbeidsvreugde te maximaliseren wordt aan iedere werknemer gevraagd hoeveel werkplezier hij/zij ontleent aan het werken in periode t ($t = 1, \ldots, T$); gebruik w_{jt} als de werkplezierscore van persoon j voor periode t. Gegeven een rooster wordt de hoeveelheid werkplezier van werknemer j berekend als de som van de w_{jt} waarden over de perioden waarin j aan het werk is. In periode t ($t = 1, \ldots, T$) zijn minstens l_t werknemers nodig; het maakt (voorlopig) niet uit hoeveel er werken, zolang het er maar minstens l_t zijn. Om te voorkomen dat werknemers een prutrooster krijgen wordt voor iedere werknemer een ondergrens k_j ($j = 1, \ldots, n$) bepaald; een rooster is alleen toegelaten indien iedere werknemer j ($j = 1, \ldots, n$) een hoeveelheid werkplezier ontleent aan zijn rooster dat $\geq k_j$ is. Het doel is het bepalen van een toegelaten rooster waarin de totale hoeveelheid werkplezier maximaal is.

(a) Formuleer het ROOSTERPROBLEEM als een (I)LP-probleem. **Geef hierbij netjes aan wat de beslissingsvariabelen, beperkingen en doelstellingsfunctie voorstellen.**

Voer beslissingsvariabelen x_{jt} ($j = 1, \ldots, n; t = 1, \ldots, T$)in die aangeven of werknemer j werkt in periode t (dan $x_{jt} = 1$) of niet. Het ILP wordt dan

$$\max \textstyle\sum_{j=1}^{n} \sum_{t=1}^{T} w_{jt} x_{jt} \quad \text{o.d.v.}$$

$$\textstyle\sum_{t=1}^{T} x_{jt} = m \quad \forall j \quad \text{(iedereen werkt } m \text{ perioden)}$$

$$\textstyle\sum_{j=1}^{n} x_{jt} \geq l_t \quad \forall t \quad \text{(voldoende werknemers)}$$

$$\textstyle\sum_{j=1}^{n} w_{jt} x_{jt} \geq k_j \quad \forall j \quad \text{(voldoende werkplezier)}$$

$$x_{jt} \in \{0, 1\} \quad \forall j, t \quad \text{(binaire variabelen)}$$

(b) Het bedrijf heeft bedacht dat, als er minstens q werknemers meer aanwezig zijn in een tijdsperiode t dan de minimaal benodigde l_t (dus minstens $l_t + q$ in totaal), dan is het mogelijk om in die periode t een training te organiseren. Om in aanmerking te komen voor een (verder volstrekt nutteloos) certificaat moet er minstens 5 maal worden getraind. Het doel wordt nu het bepalen van een toegelaten rooster waarin de totale hoeveelheid werkplezier maximaal is met als bijkomende voorwaarde dat er in minstens 5 perioden wordt getraind. Geef aan hoe u de formulering bij (a) moet aanpassen om deze variant van het probleem als (I)LP te modelleren.

Voer de extra beslissingsvariabelen y_t ($t = 1, \ldots, T$) in die aangeven of er in periode t wordt getraind (dan $y_t = 1$) of niet. Je wilt dat er minstens vijf keer wordt getraind, en een training moet alleen mogelijk zijn indien je minstens q mensen extra beschikbaar hebt. Dit kun je bereiken met het volgende ILP:

$$\max \textstyle\sum_{j=1}^n \sum_{t=1}^T w_{jt}x_{jt} \quad \text{o.d.v.}$$
$$\textstyle\sum_{t=1}^T x_{jt} = m \quad \forall j$$
$$\textstyle\sum_{j=1}^n x_{jt} \geq l_t + qy_t \quad \forall t \quad \text{(voldoende mensen voor training)}$$
$$\textstyle\sum_{j=1}^n w_{jt}x_{jt} \geq k_j \quad \forall j$$
$$\textstyle\sum_{t=1}^T y_t \geq 5 \quad \forall t \quad \text{(voldoende training)}$$
$$x_{jt} \in \{0,1\} \quad \forall j,t$$
$$y_t \in \{0,1\} \quad \forall t \quad \text{(binaire variabelen)}$$

Alleen als er minstens $l_t + q$ werknemers aanwezig zijn in periode t, dan mag y_t de waarde 1 aannemen; als het aantal aanwezige werknemers kleiner is (maar tenminste l_t bedraagt), dan moet $y_t = 0$ zijn. De rest spreekt voor zich.

(c) Het blijkt dat het goed mogelijk is om aan die extra eis van 5 trainingen te voldoen, maar de hoeveelheid trainingen is nogal ongelijk verdeeld onder de werknemers. Daarom wordt de eis van minstens 5 trainingsperioden vervangen door de eis dat iedere werknemer werkzaam moet zijn in minstens 3 perioden waarin wordt getraind. De overige eisen blijven gelijk. Het doel is weer het bepalen van een toegelaten rooster waarin de totale hoeveelheid werkplezier maximaal is. Geef aan hoe u de formulering bij (b) moet aanpassen om deze variant van het probleem als (I)LP te modelleren.

Voer de extra beslissingsvariabelen u_{jt} ($t = 1, \ldots, T$) in die aangeven of werknemer j traint in periode t (dan $u_{jt} = 1$) of niet. De eis dat iedereen minstens driemaal moet trainen is triviaal, maar je moet er nog voor zorgen dat u_{jt} de juiste waarde krijgt: u_{jt} mag alleen waarde 1 krijgen indien hij/zij aanwezig is (dus $x_{jt} = 1$) en als er in die periode wordt getraind (dus als $y_t = 1$); dit kan dus alleen indien $x_{jt} + y_t = 2$ (en natuurlijk ook als $x_{jt} \cdot y_t = 1$, maar dat is niet lineair). Het geheel kun je afdwingen met de extra beperkingen

$$\textstyle\sum_{t=1}^T u_{jt} = 3 \quad \forall j \quad \text{(voldoende training)}$$
$$u_{jt} \leq (x_{jt} + y_t)/2 \quad \forall j,t \quad \text{(afdwingen goede waarde)}$$
$$x_{jt} \in \{0,1\} \quad \forall j,t \quad \text{(binaire variabelen)}$$

Opgave 3
Beschouw het volgende lineair programmeringsprobleem.

$$\begin{array}{llcrcrcrcl}
\text{(P)} & \text{Minimaliseer} \quad z & = & c_1x_1 & + & c_2x_2 & + & 15x_3 & & \\
 & \text{o.v.} & & a_{11}x_1 & & & + & x_3 & \leq & b_1 \\
 & & & a_{21}x_1 & - & 2x_2 & + & x_3 & \geq & b_2 \\
 & & & a_{31}x_1 & + & x_2 & - & x_3 & \leq & b_3 \\
 & & & x_1 & , & x_2 & , & x_3 & \geq & 0
\end{array}$$

Laat x_4, x_5, x_6 de spelingsvariabelen zijn. Het volgende tableau is het laatste tableau van de tweede fase.

x_1	x_2	x_3	x_4	x_5	x_6	RHS
0	0	p	0	-1	-1	W
0	0	q	1	1	-2	2
0	1	r	0	1	-1	1
1	0	s	0	-1	2	3

(a) Bepaal **met behulp van het tableau** B^{-1}, $c_B B^{-1}$ en de optimale oplossing (punt en waarde); **motiveer uw antwoord**. Bepaal $a_{11}, a_{21}, a_{31}, b_1, b_2, b_3, c_1, c_2, p, q, r, s, W$ en B. **Je hoeft de waarden niet se in deze volgorde te bepalen; het is verstandig om goed naar de volgorde te kijken.** Ga daarbij uit van de gegeven instantie; houd rekening met de $\geq$ tekens. Schrijf na afloop van je berekeningen de antwoorden op het antwoordenblad. Indien je een bepaalde waarde niet hebt kunnen vinden, dan mag je doorrekenen met waarde 1 (waarschijnlijk niet de correcte waarde); **geef dit duidelijk aan**.

In de uitwerking stond een fout in $\mathbf{B}$; deze is nu gecorrigeerd. Voeg eerst de spelingsvariabelen toe en schrijf de vergelijkingen dan als $\mathbf{Ax} = \mathbf{b}$. Dit levert

$$(\mathbf{A}|\mathbf{b}) = \left(\begin{array}{cccccc|c} a_{11} & 0 & 1 & 1 & 0 & 0 & b_1 \\ a_{21} & -2 & 1 & 0 & -1 & 0 & b_2 \\ a_{31} & 1 & -1 & 0 & 0 & 1 & b_3 \end{array}\right)$$

Het gedeelte van het tableau op de rijen 1, 2, 3 is tot stand gekomen door $(\mathbf{A}|\mathbf{b})$ voor te vermenigvuldigen met $\mathbf{B^{-1}}$; er geldt dus

$$(\mathbf{B^{-1}A}|\mathbf{B^{-1}b}) = \left(\begin{array}{cccccc|c} 0 & 0 & q & 1 & 1 & -2 & 2 \\ 0 & 1 & r & 0 & 1 & -1 & 1 \\ 1 & 0 & s & 0 & -1 & 2 & 3 \end{array}\right)$$

Hieruit volgt dat $\mathbf{B^{-1}a_4} = \mathbf{B^{-1}e_1} = (1,0,0)^T$; $\mathbf{B^{-1}a_5} = \mathbf{B^{-1}}(-\mathbf{e_2}) = (1,1,-1)^T$; $\mathbf{B^{-1}a_6} = \mathbf{B^{-1}e_3} = (-2,-1,2)^T$. Corrigeren voor het minteken levert dus op dat

$$\mathbf{B^{-1}} = \begin{pmatrix} 1 & -1 & -2 \\ 0 & -1 & -1 \\ 0 & 1 & 2 \end{pmatrix}$$

Uit de positie van de eenheidsvectoren in het tableau kun je afleiden dat $x_{B_1} = x_4$, $x_{B_2} = x_2$, en $x_{B_3} = x_1$. Hieruit volgt dat $\mathbf{B} = (\mathbf{a_{B_1}}, \mathbf{a_{B_2}}, \mathbf{a_{B_3}}) = (\mathbf{a_4}, \mathbf{a_2}, \mathbf{a_1})$. De vector $\mathbf{a_1}$ is onbekend, maar die kun je uitrekenen, aangezien je weet dat $\mathbf{BB^{-1}} = \mathbf{I}$ (qua rekenwerk is dit veel handiger dan gebruiken dat $\mathbf{B^{-1}B} = \mathbf{I}$). Neem de tweede kolom van $\mathbf{B^{-1}}$, en bereken $\mathbf{a_1}$ uit de vergelijkingen

$$\mathbf{B}\begin{pmatrix} -1 \\ -1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & a_{11} \\ 0 & -2 & a_{21} \\ 0 & 1 & a_{31} \end{pmatrix}\begin{pmatrix} -1 \\ -1 \\ 1 \end{pmatrix} = \begin{pmatrix} -1 + a_{11} \\ 2 + a_{21} \\ -1 + a_{31} \end{pmatrix} = \mathbf{e_2} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

Hieruit volgt $a_{11} = 1, a_{21} = -1, a_{31} = 1$, waaruit $\mathbf{B}$ volgt.

Je kunt $\mathbf{c_B B^{-1}}$ bepalen door weer gebruik te maken van de spelingsvariabelen. Er geldt $0 = z_4 - c_4 = z_4 = \mathbf{c_B B^{-1} a_4} = \mathbf{c_B B^{-1} e_1} = (\mathbf{c_B B^{-1}})_\mathbf{1}$ (dit is mijn notatie voor de eerste coördinaat van $\mathbf{c_B B^{-1}}$; $-1 = z_5 - c_5 = z_4 = \mathbf{c_B B^{-1} a_5} = \mathbf{c_B B^{-1}}(-\mathbf{e_2}) = -(\mathbf{c_B B^{-1}})_\mathbf{2}$; en $-1 = z_6 - c_6 = z_4 = \mathbf{c_B B^{-1} a_6} = \mathbf{c_B B^{-1} e_3} = (\mathbf{c_B B^{-1}})_\mathbf{3}$. Hieruit volgt dat $\mathbf{c_B B^{-1}} = (0, 1, -1)$.

De vector $\mathbf{b} = (b_1, b_2, b_3)^T$ kun je bepalen, omdat je $\mathbf{B}$ en $\mathbf{B^{-1} b}$ kent. Uitvermenigvuldigen levert $\mathbf{b} = (5, -5, 4)^T$.

De vector $\mathbf{c_B} = (c_{B_1}, c_{B_2}, c_{B_3}) = (c_4, c_2, c_1)^T$ kun je bepalen, omdat je $\mathbf{c_B B^{-1}}$ en $\mathbf{B}$ kent. Uitvermenigvuldigen levert $\mathbf{c_B} = (c_4, c_2, c_1) = (0, -3, -2)^T$. Hieruit volgt $c_1 = -2$ en $c_2 = -3$.

De vector $(q, r, s)^T$ is gelijk aan $\mathbf{B^{-1} a_3}$, die beide bekend zijn. Uitvermenigvuldigen levert $(q, r, s) = (2, 0, -1)^T$. Verder geldt $p = z_3 - c_3 = \mathbf{c_B B^{-1} a_3} - c_3$; invullen levert $p = -13$.

Voor W geldt $W = \mathbf{c_B B^{-1} b}$; omdat je $\mathbf{c_B}$ en $\mathbf{B^{-1} b}$ kent (evenals $\mathbf{c_B B^{-1}}$ en $\mathbf{b}$) kun je W bepalen door uit te vermenigvuldigen; hieruit volgt $W = 4$. Dit is de waarde van de bijbehorende TBO, dit overeenkomt met het punt $(x_1, \ldots, x_6) = (3, 1, 0, 2, 0, 0)$.

(b) Geef het tableau van de **Revised Simplex** methode. Geef de formules indien u bij (a) de correcte waarden niet hebt kunnen vinden; eigenlijk kan het sowieso geen kwaad om de formules erbij te zetten.

Het Revised Simplex tableau ziet er in formule vorm als volgt uit:

	Basic inverse	RHS
z	$\mathbf{c_B B^{-1}}$	$\mathbf{c_B B^{-1} b}$
x_B	$\mathbf{B^{-1}}$	$\mathbf{B^{-1} b}$

Hierin moet je dan de gegevens die je bij (a) hebt gevonden invullen.

(c) Laat de variabele x_3 weg uit het probleem en voeg de nieuwe variable x_0 toe aan het probleem (op het antwoordvel wordt deze overigens weer x_3 genoemd). Deze variabele x_0 heeft c_0 en a_0 zodanig dat in het tableau onder x_0 komt te staan $(1, 1, -1, 1)$ (zie extra blad). Voer 1 (en niet meer) iteratie uit waarbij de oplossing wordt verbeterd. Geef na afloop aan wat de gevonden oplossing is (punt en waarde). Geef ook aan of deze oplossing optimaal is. **Motiveer uw antwoord**.

Het aangepaste probleem luidt nu (met $W = 4$):

x_1	x_2	x_0	x_4	x_5	x_6	RHS
0	0	1	0	-1	-1	4
0	0	1	1	1	-2	2
0	1	-1	0	1	-1	1
1	0	1	0	-1	2	3

Het levert op om x_0 te verhogen (want $z_0 - c_0 = 1$). Volgens de ratio-regel gaat $x_{B_1} = x_4$

dan uit de basis. Pivoteer op element y_{10}. Dit levert

x_1	x_2	x_3	x_4	x_5	x_6	RHS
0	0	0	-1	-2	1	2
0	0	1	1	1	-2	2
0	1	0	1	2	-3	3
1	0	0	-1	-2	4	1

De bijbehorende TBO is $(x_1, x_2, x_0, x_4, x_5, x_6) = (1, 3, 2, 0, 0, 0)$ met waarde 2. Deze is niet optimaal, want $z_6 - c_6 = 1$, dus het loont om x_6 te verhogen.

Opgave 4

Stel dat je **maximaliseringsprobleem** $\{\max\ z = cx\ \text{ s.t. }\ Ax = b,\ \ x \geq 0\}$ wilt oplossen met de simplex methode.

(a) Geef aan hoe je kunt nagaan of het toegelaten gebied leeg is. **Motiveer en bewijs uw antwoord.**

Dit volgt uit het eindresultaat van de eerste fase. Hier minimaliseer je z', die gelijk is aan de som van de kunstmatige variabelen. Wanneer de optimale oplossing een waarde $z' > 0$ heeft, dan is er geen toegelaten oplossing, want die zou $z' = 0$ hebben opgeleverd.

(b) Stel dat er een TBO gegeven is en dat je het bijbehorende tableau kent (dit tableau is hetzelfde qua opzet als bij het minimaliseringsprobleem). Geef aan hoe je in een tableau een oneindig maximum kunt herkennen. **U hoeft geen bewijs te leveren, maar u moet wel een motivatie geven over het hoe en waarom**. Let op dat het om een maximaliseringsprobleem gaat.

In geval van maximaliseren loont het om x_k in de basis te brengen indien $z_k - c_k < 0$ (precies omgekeerd aan minimaliseren). Bij de iteratie gebruik je dezelfde ratio-regel om toegelaten te blijven, want je werkt met hetzelfde toegelaten gebied. Indien $y_{ik} \leq 0$ voor alle $i = 1, \ldots, m$, dan is er sprake van een onbegrensd maximum.