Aufgabe 1)

Gemäß Definition 1.4 in Wolsey ist P_2 genau dann eine bessere Formulierung, wenn $P_2 \subset P_1$. Dies ist immer dann der Fall, wenn es Punkte gibt, die in P_1 liegen, aber nicht in P_2.

$\exists x_1 \in P_1 : x_1 \notin P_2$

Wir betrachten jetzt die LP-Relaxationen der beiden Formulierungen und wählen für P_1 die folgende gültige Lösung:

$x_1 = 0,2$

$x_2 = 0,9$

$x_3 = 1,0$

Diese Lösung ist für die einzige Restriktion aus P_1 gültig. Diese Lösung gilt aber nicht für die zweite Restriktion aus P_2, denn $x_1 + x_2 = 0,2 + 0,9$ ist nicht ≤ 1

Aufgabe 2)

a) Uncapacitated Lot-Sizing (ULS) Problem

Parameter

f_t: Fixkosten in Periode t, falls produziert wird

p_t:Produktionskosten einer Einheit in Periode t

h_t: Lagerkosten einer Einheit in Periode t

d_t: Nachfrage in Periode t

M: BigM

Entscheidungsvariablen

x_t: Produktionsmenge in Periode t

s_t: Lagermenge am Ende von Periode t

$y_t = 1$ falls in Periode t produziert wird, andernfalls $y_t = 0$

$$\min \sum_{t=1}^{n} p_t x_t + \sum_{t=1}^{n} h_t s_t + \sum_{t=1}^{n} f_t y_t$$

$$\text{s. t.} \quad \begin{aligned} & s_{t-1} + x_t = d_t + s_t \; for \; t = 1,...,n \\ & x_t \leq M y_t \; for \; t = 1,...,n \\ & s_0 = 0, s_t, x_t \geq 0, y_t \in \{0,1\} \; for \; t = 1,...,n \end{aligned}$$

Ein optimaler Produktionsplan für die gegebenen Parameter lautet: {9; 0; 8; 0; 11; 0}.
Dabei wird nur in den Perioden 1, 3 und 5 Produziert und jeweils 3, 4 und 4 Mengeneinheiten eingelagert.

ampl-Modell

```
# -Sets-
set t;

# -Parameter-
param p {t};
param h {t};
param d {t};
param f {t};

# -Variablen-
var y {j in t}, binary;
var x{j in t}, integer ;
var s{j in 0..6}, integer;

# -Zielfunktion-
minimize kosten: sum{j in t} p[j]*x[j] + sum{j in t} h[j]*s[j] + sum{j in t}
f[j]*y[j];

# -Restriktionen-
subject to lagerkosten {j in t} : s[j-1] + x[j] = d[j] + s[j];
subject to prodlimit {j in t}: x[j] <= 28*y[j];
subject to lager {j in t}: s[j] >= 0;
subject to anfangsbedarf: s[0] = 0;
subject to stück {j in t} : x[j] >=0;
```

Data

```
# data für das Produktionsmodell
set t := 1 2 3 4 5 6;
param: f:= 1 60 2 20 3 40 4 40 5 10 6 30 ;
param p:= 1 2 2 3 3 3 4 4 5 7 6 5;
param h:= 1 4 2 4 3 4 4 4 5 4 6 4;
param d:= 1 6 2 3 3 4 4 4 5 7 6 4;
```

Lösungsergebnis mit MOPS

```
MOPS Studio solution summary for model 'ago1.2'
-----------------------------------------------
Solverstatus:                  IPFinished
Solutionstatus:                IPOptimalSolution
LP objective function value:   180.85
IP objective function value:   273

Index   Name                LP-Solution         IP-Solution
-----   ----                -----------         -----------
1       y[1]                          1                   1
2       y[2]                      0.015                   0
3       y[3]                       0.02                   1
4       y[4]                       0.02                   0
5       y[5]                      0.035                   1
6       y[6]                       0.02                   0
7       x[1]                          6                   9
8       x[2]                          3                   0
9       x[3]                          4                   8

10      x[4]                          4                   0
11      x[5]                          7                  11
12      x[6]                          4                   0
13      s[0]                          0                   0
14      s[1]                          0                   3
15      s[2]                          0                   0
16      s[3]                          0                   4
17      s[4]                          0                   0
18      s[5]                          0                   4
19      s[6]                          0                   0
```

b) Extended Formulation for ULS

Variablen

w_{it} : Produktionsmenge in Periode i, um Nachfrage in t zu decken

$y_t = 1$ falls in Periode t produziert wird, andernfalls $y_t = 0$

Nebenbedingungen

$$\sum_{i=1}^{t} w_{it} = d_t \quad \textit{for all } t$$

$$w_{it} \le d_t y_i \quad \textit{for all } i,t, \quad i \le t$$

$$w_{it} \ge 0 \quad \textit{for all } i,t, \quad i \le t \quad y_t \in \{0,1\} \quad \textit{for all } t$$

Funktion

$$\min \sum_{i=1}^{n} \sum_{t=i}^{n} c_i w_{it} + \sum_{t=1}^{n} f_t y_t$$

ampl-Modell

```
# -Sets-
set p;

# -Parameter-
param a {t in p, i in p};
param d {t in p};
param f {t in p};

# -Variablen-
var y{t in p}, binary;
var x{i in p}, >= 0, integer;
var w{i in p, t in p}, >= 0, integer;

# -Zielfunktion-
minimize kosten: sum{t in p, i in p} a[i,t]*w[i,t]  + sum{t in p} f[t]*y[t];

# -Restriktionen-
subject to nachfrage{t in p}: sum{i in 1..t} w[i,t] = d[t];
subject to fixeKosten{t in p, i in 1..t}: w[i,t] <= d[t]*y[i];
subject to Summe{i in p}: x[i] = sum{t in p} w[i,t];
```

Data

```
# data für Produktionsmodell
set p := 1 2 3 4 5 6;   #Perioden
param: f:= 1 60 2 20 3 40 4 40 5 10 6 30 ;
param a:      1    2      3      4      5      6 :=
              1 2    6      10     14     18     22
              2 999 3      7      11     15     19
              3 999 999    3      7      11     15
              4 999 999    999    4      8      12
              5 999 999    999    999    7      11
              6 999 999    999    999    999    5;

param d:= 1 6 2 3 3 4 4 4 5 7 6 4;
```

Vergleich der beiden Formulierungen:

Die Extended ULS hat z. B. zu der Standad ULS deutlich weniger LP-Iterationen. Ebenfalls fällt auf, dass die LU-factorization deutlich geringer ist. Insgesamt ist die Extended ULS besser, weil hier die von MOPS gelöste LP-Relaxation schon ganzzahlig ist, wobei bei der Standard ULS die LP-Lösung noch nicht ganzzahlig ist. Es muss daher danach noch nach einer IP-Lösung gesucht werden.

Lösungsergebnis mit MOPS

```
MOPS Studio solution summary for model 'ago1.2b'
------------------------------------------------
Solverstatus:                  IPFinished
Solutionstatus:                IPOptimalSolution
LP objective function value:   273
IP objective function value:   273
```

Index	Name	LP-Solution	IP-Solution
1	y[1]	1	1
2	y[2]	0	0
3	y[3]	1	1
4	y[4]	0	0
5	y[5]	1	1
6	y[6]	0	0
7	x[1]	9	9
8	x[2]	0	0
9	x[3]	8	8
10	x[4]	0	0
11	x[5]	11	11
12	x[6]	0	0
13	w[1,1]	6	6
14	w[1,2]	3	3
15	w[1,3]	0	0
16	w[1,4]	0	0
17	w[1,5]	0	0
18	w[1,6]	0	0
19	w[2,1]	0	0
20	w[2,2]	0	0
21	w[2,3]	0	0
22	w[2,4]	0	0
23	w[2,5]	0	0
24	w[2,6]	0	0
25	w[3,1]	0	0
26	w[3,2]	0	0
27	w[3,3]	4	4
28	w[3,4]	4	4
29	w[3,5]	0	0
30	w[3,6]	0	0
31	w[4,1]	0	0
32	w[4,2]	0	0
33	w[4,3]	0	0
34	w[4,4]	0	0
35	w[4,5]	0	0
36	w[4,6]	0	0
37	w[5,1]	0	0
38	w[5,2]	0	0
39	w[5,3]	0	0
40	w[5,4]	0	0
41	w[5,5]	7	7
42	w[5,6]	4	4
43	w[6,1]	0	0
44	w[6,2]	0	0
45	w[6,3]	0	0
46	w[6,4]	0	0
47	w[6,5]	0	0
48	w[6,6]	0	0

Aufgabe 3)

a) Matrices TU or not

$$A_1 = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & -1 \\ 0 & -1 & 0 & 1 & 1 & 0 \\ 0 & -1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}$$

A_1 is not TU, because there exists a submatrix where $\det(A_i)=2$

A_2 is TU

A_3 is TU

For A_2 and A_3:

1) $a_{ij} \in \{+1, 0, -1\}$ *for all* i, j

2) Each column contains at most two nonzero coefficients $\left(\sum_{i=1}^{m} |a_{ij}| \le 2 \right)$

3) There exits a partition (M_1, M_2) of the set M of rows such that each column j containing two nonzero coefficients satisfies $\sum_{i \in M_1} a_{ij} - \sum_{i \in M_2} a_{ij} = 0$.

For A_2: $M_1 = \{1,2\}$, $M_2 = \{3,4\}$ and for A_3: $M_1 = \{1,3\}$, $M_2 = \{2\}$

b)
Der Startpunkt ist das folgende Integer Programm
max{cx: $Ax \le b$, $x \ge 0$, integer } (IP)

Annahme:
A ist TU, b integer

Aus der LP wissen wir, dass $x^* = (x_B, x_N) = (B^{-1}b, 0)$, wobei für die Nichtbasisvariablen x_N gilt $x_N = 0$ und für die Basisvariablen x_B gilt $B^{-1}b$.

Aus der Cramersche Regel folgern wir: $B^{-1} = \frac{B^*}{\det(B)}$

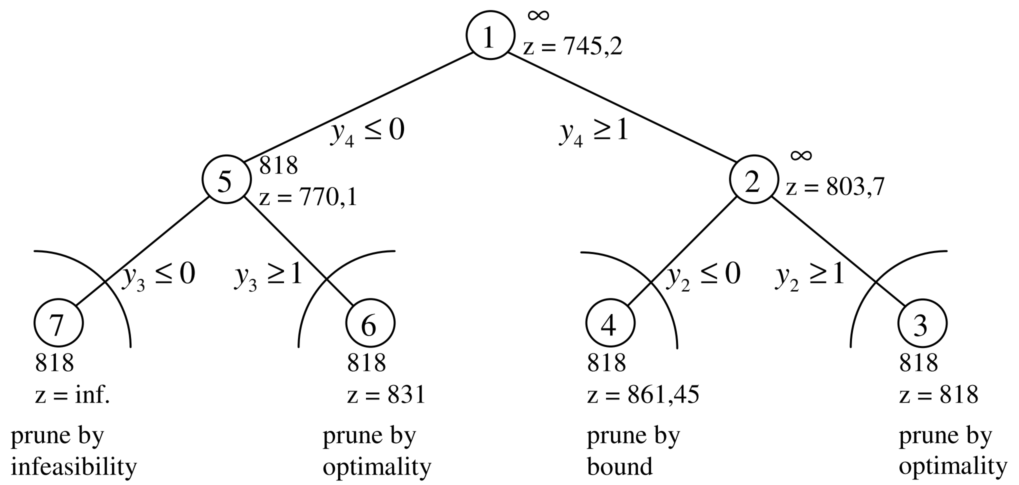
- det(B) kann aufgrund der TU nur $= \pm 1$ betragen
- B* ist die adjunkte Matrix von B ~> alle Elemente aus B sind Produkte aus der Basis B der IP-Lösung
- aus A TU ~> B ganzzahlig ~> B* ganzzahlig

Ergebnis:

B* integer, det(B) integer ~> $B^{-1} = \frac{B^*}{\det(B)}$ integer

Aufgabe 4)

Depth First Node Selection

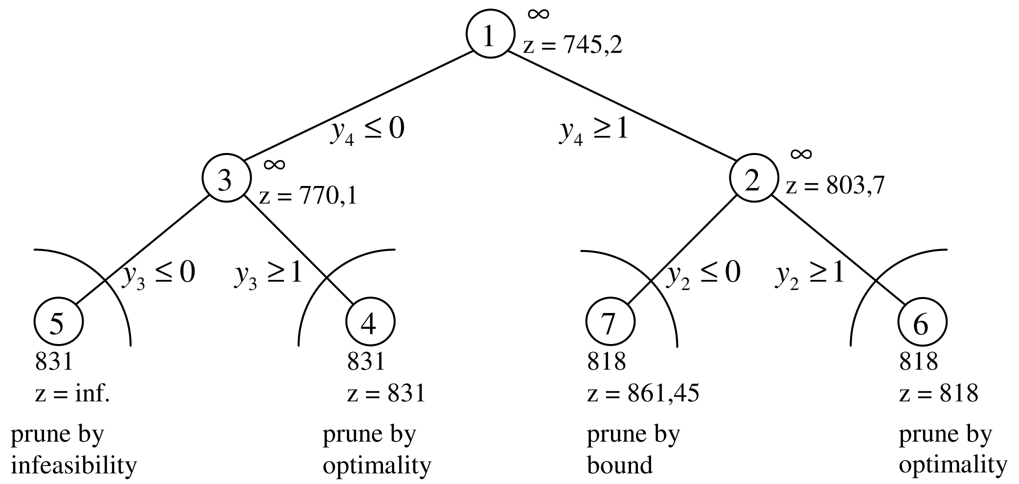


Optimale Lösung

Zielfunktionswert: z = 818

Variablenwert: $s_1 = 16;\ s_2 = 25;\ s_3 = 13;\ s_4 = 0;\ x_1 = 20;\ x_2 = 20;\ x_3 = 0;\ x_4 = 6;\ y_1 = 1;\ y_2 = 1;\ y_3 = 0;\ y_4 = 1;$

Best First Node Selection



Optimale Lösung

Zielfunktionswert: z = 818

Variablenwert: $s_1 = 16;\ s_2 = 25;\ s_3 = 13;\ s_4 = 0;\ x_1 = 20;\ x_2 = 20;\ x_3 = 0;\ x_4 = 6;\ y_1 = 1;\ y_2 = 1;\ y_3 = 0;\ y_4 = 1;$