Written Exam
Linear and Integer Programming
(DM545)

Department of Mathematics and Computer Science
University of Southern Denmark

Tuesday, June 10, 2014, 10:00–14:00, U9 and U27

The exam consists of a number of tasks divided into subtasks. The answers in a PDF document are to be handed in electronically in Blackboard (http://e-learn.sdu.dk).

- The content of the documents "Eksamensvejledning" and "Exam Monitor" published in BlackBoard under the section Course Information is assumed to be known.
- *Remember to justify all your statements!* You may refer to results from the books or the lecture slides listed at the course web page. In particular, it is possible to justify a statement by saying that it derives trivially from a result in the textbook (if this is true). You may use all methods or extensions that have been used in the assignment sheets, published during the course. However, it is not allowed to answer a subtask exclusively by reference to an exercise seen during the course. Reference to other books (outside the course material) or to internet links is not accepted as answer to a task.
- The contribution to the final evaluation of each task, if carried out correctly, is given at the beginning of each task as a list of points for each subtask. The number of points does not necessarily reflect the difficulty of the subtask. The maximum score is 100.

  Exercises are sorted in the order of treatment during the course.
- The exam consists of 8 tasks distributed on 26 pages.
- You may write your answers in Danish or in English.

An HTML version of this document is available at:

http://www.imada.sdu.dk/~marco/DM545/Exam/w8wlOND.html

Contents

Task 1 Network Flows (points: 7)

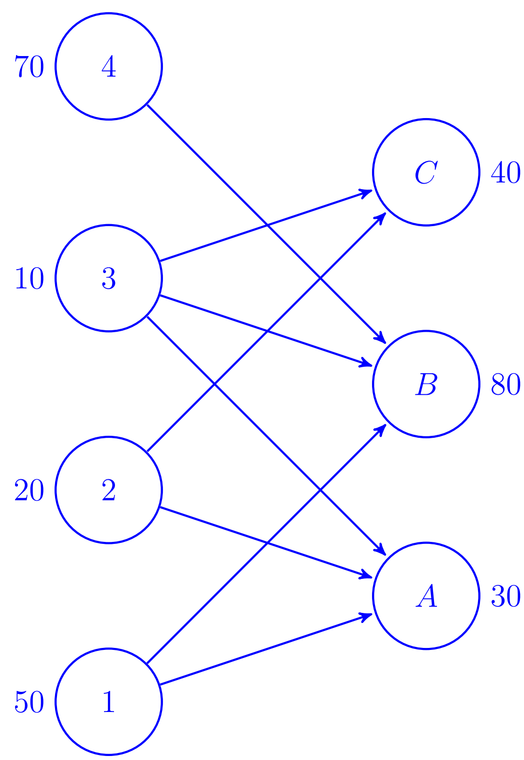
A set of work centers performs operations of various type on a set of mechanical pieces. Each operation $\{1,2,3,4\}$ can be performed on a subset of the work centers $\{A,B,C\}$. The requirement for each type of operation is expressed in minutes:

operation	center	minutes
1	A,B	50
2	A,C	20
3	A,B,C	10
4	B	70

In the next working shift the working centers A, B and C will be available for 30, 80 and 40 minutes, respectively. The problem consists in determining whether it is possible to finish all operations in the next shift. Formulate the problem in terms of network flows.

Solution

The problem can be modelled as a transportation problem.



The network can be transformed in an *st*-network introducing source and target and the problem solved as a max flow problem in the new network. If the flow saturates the arcs going out from s then the it is possible to finish all operations in the next shift.

Task 2 Branch and bound (points: 5, 4, 4)

Consider the Integer Linear Programming problem $P0$: $z = \max\{cx : Ax \leq b, x \text{ integer}\}$ solved by a Branch & Bound algorithm.
In Figure 1 it is represented a situation during the run of the algorithm (the symbol $\nexists$ indicates an infeasible problem).

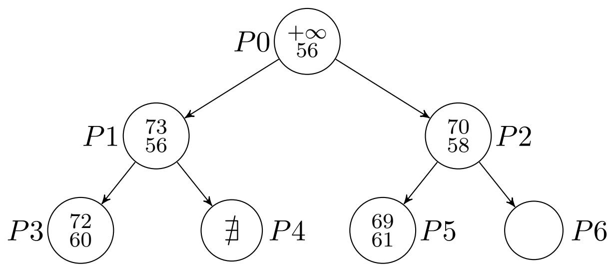


Figure 1:

Subtask 2.a

Give the tightest possible lower and upper bounds on the optimal value z.

Solution Whenever a problem is branched into $k = 1, \ldots, K$ children we have that $\overline{z} = \max_k \overline{z}^k$ is an upper bound on z; $\underline{z} = \max_k \underline{z}^k$ is a lower bound on z
Hence the optimal solution is within $[72; 61]$.

Subtask 2.b

For which values of upper bound and lower bound at node P6 is it possible to prune the tree below that node?

Solution Pruning can occur by infeasibility or by optimality or by bounding. By optimality we prune if $\underline{z}^{P6} = \overline{z}^{P6} \in [70; 58]$ By bounding we prune if $\overline{z}^{P6} \leq \underline{z}$, that is, if $\overline{z}^{P6} \leq 61$.

Subtask 2.c

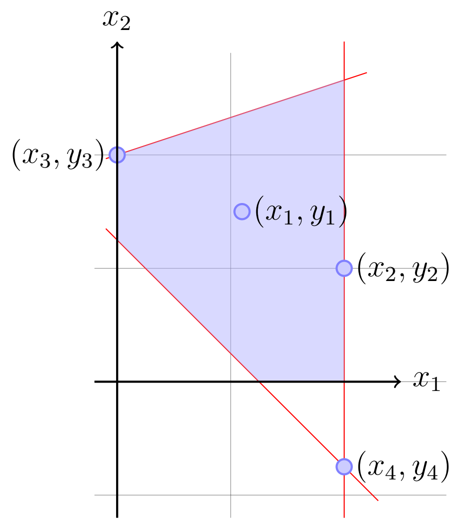
If any, for which values of upper bound and lower bound at node $P6$ is it possible to indicate the optimal solution and close all nodes of the tree?

Solution Since nodes P3 and P5 are open, to close all nodes we must find the optimal solution in P6. But in P6 the optimal solution can be at most $\overline{z}^{P6} = \underline{z}^{P6} \leq 70$. Since $\overline{z} = 72$ then it is not possible to prune all nodes, hence there are no values for the bounds in P6 that would satisfy the requirement.

Task 3 Simplex (points: 4, 5, 10, 5)

Consider the following LP problem:

$$
\begin{array}{llrcrl}
(P) & \max & z = x_1 & + & 5x_2 & \\
 & \text{s.t.} & -x_1 & + & 3x_2 & \leq 6 \\
 & & 4x_1 & + & 4x_2 & \geq 5 \\
 & & 0 & \leq & x_1 & \leq 2 \\
 & & & & x_2 & \geq 0
\end{array}
$$



(Note: the following subtasks can be carried out independently; use fractional mode for numerical calculations; in the online version you find the problem in ASCII format).[1]

Subtask 3.a

The polyhedron representing the feasibility region is depicted in the figure. Indicate for each of the four points represented whether they are feasible and/or basic solutions. Justify your answer.

Solution

- Point 1 is a feasible solution but not basic (no constraint is active in that point).
- Point 2 is a feasible solution but not basic (only one constraint is active while two are needed)
- Point 3 is a basic feasible solution
- Point 4 is a basic solution (combination of two active constraints) but non feasible.

Subtask 3.b

Write the initial tableau or dictionary for the simplex method. Write the corresponding basic solution and its value. State whether the solution is feasible or not and whether it is optimal or not.

[1] Update 26.06.2014: In the figure a point (x_i, y_i) should be intended as (x_1^i, x_2^i).

Solution

$$
\begin{array}{rrcrl}
z = \max & x_1 & + & 5x_2 & \\
\text{s.t.} & -x_1 & + & 3x_2 & \leq 6 \\
 & -4x_1 & - & 4x_2 & \leq -5 \\
 & & & x_1 & \leq 2 \\
 & & & x_1 & \geq 0 \\
 & & & x_2 & \geq 0
\end{array}
$$

```
|-----+----+----+----+----+----+----+----|
|     | x1 | x2 | x3 | x4 | x5 | -z |  b |
|-----+----+----+----+----+----+----+----|
| I   | -1 |  3 |  1 |  0 |  0 |  0 |  6 |
| II  | -4 | -4 |  0 |  1 |  0 |  0 | -5 |
| III |  1 |  0 |  0 |  0 |  1 |  0 |  2 |
| IV  |  1 |  5 |  0 |  0 |  0 |  1 |  0 |
|-----+----+----+----+----+----+----+----|
```

The basic solution is $x_1 = 0$, $x_2 = 0$, $x_3 = 6$, $x_4 = -5$ and $x_5 = 2$. Its value is 0. The solution is not feasible.

Subtask 3.c

Consider the following tableau:

	x1	x2	x3	x4	x5	-z	b
I	0	4	1	-1/4	0	0	29/4
II	1	1	0	-1/4	0	0	5/4
III	0	-1	0	1/4	1	0	3/4
IV	0	4	0	1/4	0	1	-5/4

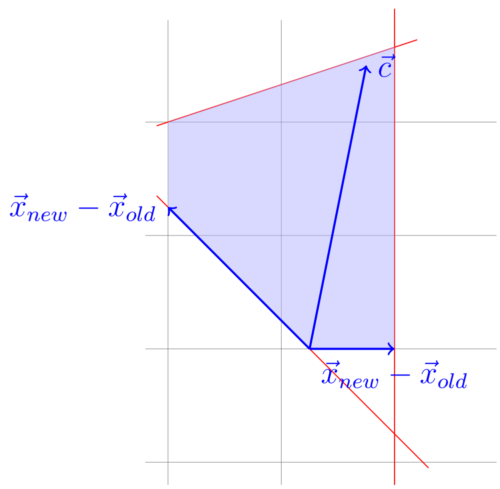
and the following three pivoting rules:

- largest coefficient
- largest increase
- steepest edge.

Which entering and leaving variables would each of them indicate? In this specific case, which rule would be convenient to follow? Report the details of the computations for the first two rules and carry out graphically the application of the third rule using the plot in the figure above (tikz code to reproduce the figure available in the online version.)

Solution

- The two candidate entering variables are $x2$ and $x4$. The reduced cost of $x2$ is larger hence that is the entering variable. The leaving variable is consequently given by the ratio test and is $x1$ since $5/4 < 29/16$.
- The two candidate entering variables are $x2$ and $x4$. The increase possible with $x2$ is $\min\{29/4 \cdot 1/4, 5/4 \cdot 1\} \cdot 4 = 5$ while the increase with $x4$ is $\min\{3/4 \cdot 4\} \cdot 1/4 = 3/4$. Hence $x2$ is the entering variable and the leaving variable is $x1$.
- In the figure we plot the vector $\vec{c}$ which is the perpendicular to the objective function and the two vectors corresponding to the movement we would take by the iteration of the simplex. The angle between $\vec{c}$ and $\vec{x}_{new} - \vec{x}_{old}$ is smaller for the decision x_2 entering x_1 leaving.



None of the three rules is convenient, the best would be to let $x4$ enter and $x5$ leave, we would reach the optimal solution in less iterations.

Subtask 3.d

For solving P_0 with the Gomory's fractional cutting plane algorithm one needs initially to solve P. After a number of iterations of the simplex algorithm the tableau looks as follows:

	x1	x2	x3	x4	x5	-z	b
I	0	0	4/3	1	16/3	0	41/3
II	0	1	1/3	0	1/3	0	8/3
III	1	0	0	0	1	0	2
IV	0	0	-5/3	0	-8/3	1	-46/3

Derive a Gomory cut and write it in the space of the original variables.
Show that the cut is a valid inequality for (P_0) and that it will make the current optimal solution of (P) infeasible.

Solution During the course we saw that the Gomory cut can be derived as follows:

$$\sum_{j\in N}(\bar{a}_{uj} - \lfloor \bar{a}_{uj} \rfloor)x_j \geq \bar{b}_u - \lfloor \bar{b}_u \rfloor$$

$$\begin{aligned}(1/3-0)x_3 + (1/3-0)x_5 &\geq (8/3-2)\\ 1/3x_3 + 1/3x_5 &\geq 2/3\\ x_3 + x_5 &\geq 2\end{aligned}$$

To express the constraint in the space of the original variables we need to substitute s_1 and s_2 as given from the first tableau:

$$(6 - 3x_2 + x_1) + (2 - x_1) \geq 2 \tag{1}$$
$$x_2 \leq 2 \tag{2}$$
$$\tag{3}$$

The cut leaves out the optimal solution.

Task 4 Revised Simplex and Sensitivity (points: 9, 6)

Consider the following problem

$$\begin{array}{rrcrl}
\min \quad z = & 5y_1 & + & 4y_2 & \\
& 4y_1 & + & 3y_2 & \geq 4 \\
& 2y_1 & + & y_2 & \geq 3 \\
& y_1 & + & 2y_2 & \geq 1 \\
& y_1 & + & y_2 & \geq 2 \\
& & & y_1, y_2 & \geq 0
\end{array}$$

Because this problem has more functional constraints than variables, we will solve it by applying the simplex method to its dual problem. Let x_1, x_2, x_3 and x_4 be the variables of the dual problem that are associated with the constraints of the primal and x_5 and x_6 be the slack variables of the dual problem. The final optimal tableau is the following:

	x1	x2	x3	x4	x5	x6	-z	b
x2	1	1	-1	0	1	-1	0	1
x4	2	0	3	1	-1	2	0	3
	-3	0	-2	0	-1	-1	1	-9

Subtask 4.a

Write:

i. the value of the variables for the optimal solution of the primal problem

ii. the value of the variables for the optimal solution of the dual problem

iii. the value of the shadow prices of the *dual* problem. If we could increase by one unit the RHS term of the dual problem, which one should we choose in order to obtain the largest improvement of the optimal value of the dual problem.

iv. the value of the shadow prices of the primal problem. If we could increase by one unit the RHS term of the primal problem, which one should we choose in order to obtain the largest worsening of the optimal value of z?

Solution Let's start by writing the dual:

$$\begin{array}{rrcrcrcrcrcrl}
\max \quad w = & 4x_1 & + & 3x_2 & + & x_3 & + & 2x_4 & & & & & \\
& 4x_1 & + & 2x_2 & + & x_3 & + & x_4 & + & x_5 & & & \leq 5 \\
& 3x_1 & + & x_2 & + & 2x_3 & + & x_4 & & & + & x_6 & \leq 4 \\
& & & & & & & \multicolumn{5}{r}{x_1, x_2, x_3, x_4, x_5, x_6} & \geq 0
\end{array}$$

i. $y_1 = 1, y_2 = 1$

ii. $x_1 = 0, x_2 = 1, x_3 = 0, x_4 = 3$

iii. The shadow prices tell us the marginal costs of the resources, ie, the RHS terms of the constraints. They tell us the costs of the opportunity missed by having one unit more in the RHS terms. The values can be read from the final tableau from the costs corresponding to the slack variables after a change of sign. They are also the value of the dual variables. Hence they are 1 and 1. Both RHS terms would yield the same improvement of 1.

iv. Shadow prices are the values of the dual variables. Hence for the primal they are: 1 for the second constraint, 3 for the fourth constraint and 0 for the other two constraints. An increase of 1 unit in the RHS term of the fourth constraint yields the largest worsening of the optimal z.

Subtask 4.b

Show how to derive the reduced costs of the final tableau above, hence for the *dual* problem, using only matrix computations, like in the revised simplex method. [Hint to be faster in matrix calculations you can use any program for linear algebra. In the online version of this document you find the R and python code for matrix calculation. In python, import `numpy` and `linalg.solve`. In R, the inverse of a matrix can be calculated by `solve` and matrix multiplications by `%*%`.]

Solution The reduced costs of the nonbasic variables are obtained by $c_N - y^T A_N$, with $y = A_B^{-1} c_B$.
We have:

```
> (A.p <- matrix(c(4, 3, 2, 1, 1, 2, 1, 1), ncol=2, nrow=4, byrow=TRUE))
     [,1] [,2]
[1,]    4    3
[2,]    2    1
[3,]    1    2
[4,]    1    1
> (b.p <- matrix(c(4,3,1,2),ncol=1))
     [,1]
[1,]    4
[2,]    3
[3,]    1
[4,]    2
> (c.p <- matrix(c(5,4),ncol=1))
     [,1]
[1,]    5
[2,]    4
>
```

```
>
> (A <- cbind(t(A.p),matrix(c(1,0,0,1),ncol=2)))
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    4    2    1    1    1    0
[2,]    3    1    2    1    0    1
> (A.N <- A[,c(1,3,5,6)])
     [,1] [,2] [,3] [,4]
[1,]    4    1    1    0
[2,]    3    2    0    1
> (A.B <- A[,c(2,4)])
     [,1] [,2]
[1,]    2    1
[2,]    1    1
> (c <- c(b.p,c(0,0)))
[1] 4 3 1 2 0 0
> (c.N <- c[c(1,3,5,6)])
[1] 4 1 0 0
> (c.B <- c[c(2,4)])
[1] 3 2
> (y <- solve(A.B)%*%c.B)
     [,1]
[1,]    1
[2,]    1
> (c.N-t(y)%*%A.N)
     [,1] [,2] [,3] [,4]
[1,]   -3   -2   -1   -1
```

Task 5 Duality Theory (points: 7, 10)

Subtask 5.a

Using only the duality theory and without using the simplex method find out if

$$x_1^* = 3, x_2^* = -1, x_3^* = 0, x_4^* = 2$$

is an optimal solution of the problem:

$$\begin{array}{lrcrcrcrl}
\max & 6x_1 & + & x_2 & - & x_3 & - & x_4 & \\
\text{s.t.} & x_1 & + & 2x_2 & + & x_3 & + & x_4 & \leq 5 \\
 & 3x_1 & + & x_2 & - & x_3 & & & \leq 8 \\
 & & & x_2 & + & x_3 & + & x_4 & = 1 \\
 & & & & & & & x_3, x_4 & \geq 0
\end{array}$$

Solution Let us write the dual:

$$\begin{array}{lrcrcrrl}
\min & 5y_1 & + & 8y_2 & + & y_3 & & \\
 & y_1 & + & 3y_2 & & & & = 6 \\
 & 2y_1 & + & y_2 & + & y_3 & & = 1 \\
 & y_1 & - & y_2 & + & y_3 & & \geq -1 \\
 & y_1 & & & + & y_3 & & \geq -1 \\
 & & & & & & y_1, y_2 & \geq 0 \\
 & & & & & & y_3 & \in \mathbb{R}
\end{array}$$

The complementary slackness conditions are:

$$y_1(x_1 + 2x_2 + x_3 + x_4 - 5) = 0 \tag{4}$$
$$y_2(3x_1 + x_2 - x_3 - 8) = 0 \tag{5}$$
$$y_3(x_2 + x_2 + x_4 - 1) = 0 \tag{6}$$

$$x_1(y_1 + 3y_2 - 6) = 0 \tag{7}$$
$$x_2(2y_1 + y_2 + y_3 - 1) = 0 \tag{8}$$
$$x_3(y_1 - y_2 + y_3 + 1) = 0 \tag{9}$$
$$x_4(y_1 + y_3 + 1) = 0 \tag{10}$$

Since the first constraint of the primal holds as strict inequality, then we can deduce that $y_1 = 0$. Further, since x_1^*, x_2^* and x_4^* are $\neq 0$ then from (7), (8) and (10) we have:

$$\begin{cases} y_1 + 3y_2 = 6 \\ 2y_1 + y_2 + y_3 = 1 \\ y_1 + y_3 = -1 \end{cases}$$

whose solution is $y_1 = 0$, $y_2 = 2$ and $y_3 = -1$. Hence, the dual solution is $y = [0; 2; -1]$. These values satisfy the domains of the dual variables. However when we substitute in the dual constraints, some constraints are not satisfied. This indicates that the dual solution is infeasible. An infeasible dual indicates that the primal is unbounded.

Subtask 5.b

Write the dual of the following problem:

$$
\begin{array}{lll}
(P) & \min \sum_{j=1}^{m} c_j y_j & \\
& \sum_{j=1}^{m} a_{ij} x_{ij} = b_i & i = 1, \ldots, n \\
& \sum_{i=1}^{n} x_{ij} \leq y_j & j = 1, \ldots, m \\
& x_{ij} \leq 1 & i = 1, \ldots, n; j = 1, \ldots, m \\
& x_{ij} \geq 0 & i = 1, \ldots, n; j = 1, \ldots, m \\
& y_j \in \mathbb{R} & j = 1, \ldots, m
\end{array}
$$

Solution There are three constraints. We introduce the dual variables $\alpha_i \in \mathbb{R}$, $\beta_j \leq 0$, $\gamma_{ij} \leq 0$
We then write the A matrix augmented with the b vector:
and finally the dual from the columns of the A matrix:

$$
\begin{array}{lll}
(D) & \max \sum_{i=1}^{n} b_i \alpha_i + \sum_{i=1}^{n} \sum_{j=1}^{m} \gamma_{ij} & \\
& a_{ij}\alpha_i + \beta_j + \gamma_{ij} \geq 0 & i = 1, \ldots, n, j = 1, \ldots, m \\
& -\beta_j = c_j & j = 1, \ldots, m \\
& \beta_i \leq 0 & i = 1, \ldots, n; \\
& \gamma_{ij} \leq 0 & i = 1, \ldots, n; j = 1, \ldots, m \\
& \alpha_i \in \mathbb{R} & i = 1, \ldots, n;
\end{array}
$$

$$
\begin{array}{c}
\begin{array}{cccccccccccccccc}
x_{11} & x_{12} & \cdots & x_{1m} & x_{21} & x_{22} & \dots & x_{2m} & \dots & x_{n1} & \dots & x_{nm} & y_1 & y_2 & \dots & y_m
\end{array}\\
\left[\begin{array}{cccccccccccc:cccc|c}
a_{11} & a_{12} & \cdots & a_{1m} & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & 0 & 0 & \cdots & 0 & b_1\\
0 & 0 & \cdots & 0 & a_{21} & a_{22} & \dots & a_{2m} & \dots & 0 & \dots & 0 & 0 & 0 & \dots & 0 & \vdots\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\
0 & 0 & \cdots & 0 & 0 & 0 & \dots & 0 & \dots & a_{n1} & \dots & a_{nm} & 0 & 0 & \dots & 0 & b_n\\
\hdashline
1 & 0 & \cdots & 0 & 1 & 0 & \dots & 0 & \dots & 1 & \dots & 0 & -1 & 0 & \dots & 0 & 0\\
0 & 1 & \cdots & 0 & 0 & 1 & \dots & 0 & \dots & 0 & \dots & 0 & 0 & -1 & \dots & 0 & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
0 & 0 & \cdots & 1 & 0 & 0 & \dots & 1 & \dots & 0 & \dots & 1 & 0 & 0 & \dots & -1 & \vdots\\
\hdashline
1 & 0 & \cdots & 0 & 0 & 0 & \dots & 0 & \dots & 0 & \dots & 0 & 0 & 0 & \dots & 0 & 1\\
\vdots & \cdots & \cdots & \cdots & & & \vdots & & \dots & & \dots & & \vdots & & \dots & \vdots & 1\\
\vdots & \cdots & \cdots & \cdots & & & 1 & & \dots & & \dots & & \vdots & & \dots & \vdots & 1\\
\vdots & \cdots & \cdots & \cdots & & & \vdots & & \dots & & \dots & & \vdots & & \dots & \vdots & 1\\
0 & 0 & \cdots & 0 & 0 & 0 & \dots & 0 & \dots & 0 & \dots & 1 & 0 & 0 & \dots & 0 & 1
\end{array}\right]
\end{array}
$$

Task 6 LP Formulation (points: 12, 5)

A bartender serves usually alcoholic beverages behind the bar. A bartender can generally mix classic cocktails such as a Gin-and-Tonic, Caipirinha and Mojito. In order to achieve this task the bartender has to maintain the supplies and inventory for the bar.
End drinks for the customers are created by directly mixing the raw materials. Restricting our attention to the Gin-and-Tonic drink, the suggested ratios of gin and tonic are 1:1, 2:3, 1:2, and 1:3 (source Wikipedia). The historical data indicate nevertheless that the typical demand on a Saturday evening in the premise of our bartender is as follows:

Alcohol	quantity	price per deciliter
$\geq 20\%$	≤ 12 l	30 dkk
$\geq 16\%$	≤ 12 l	25 dkk
$\geq 13\%$	≤ 12 l	21 dkk
$\geq 10\%$	≤ 12 l	15 dkk

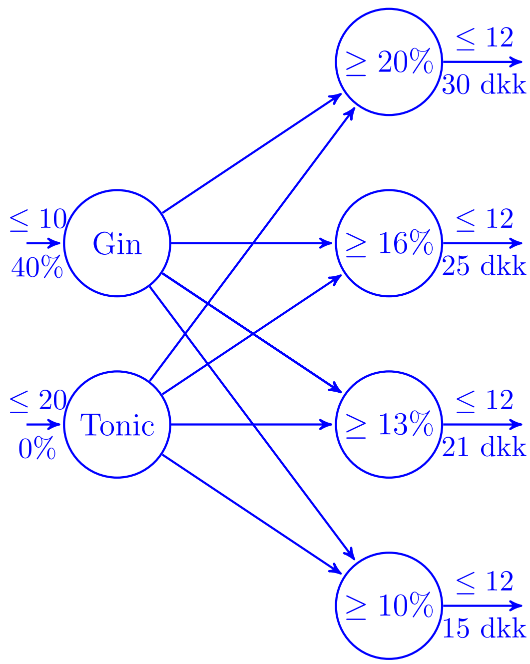
For example, the first row indicates that there are customers that are willing to intake 20% or more alcohol consuming all together up to 12 liters and willing to pay 30 krone per deciliter.
Our bartender for economical and logistic issues can buy up to 10 liters of gin that contain 40% alcohol and up to 20 liters of tonic that contain no alcohol.
The mixing process occurs in a way such that at the end the input products no longer exist in their original forms, but are mixed to form new mixed products with new property values, in this case, the percentage of alcohol content.
The objective is to maximize the total profit from the sell.
The problem is an example of blending problem that arises, beside bar keeping also in the oil refinery industry.

Subtask 6.a

Represent the situation as a network and model the problem as a linear programming problem.

Solution

Let x_{ij} be the amount of product in liters that flows from i to j. Let I be the set of sources of gin and of tonic. Let J be the set of four customer categories.



$$\max \sum_{j \in J} \sum_{i \in I} r_j x_{ij} \tag{11}$$

$$\sum_{j \in J} x_{ij} \leq b_i \qquad \forall i \in I \tag{12}$$

$$\sum_{i \in I} x_{ij} \leq d_j \qquad \forall j \in J \tag{13}$$

$$\sum_{i \in I} p_i x_{ij} \geq p_j \sum_{i \in I} x_{ij} \qquad \forall j \in J \tag{14}$$

$$x_{ij} \geq 0 \qquad \forall i \in I, j \in J \tag{15}$$

```
set INPUT;
set OUTPUT;

param revenue {OUTPUT} > 0;
param in_max {INPUT} >= 0;
param out_max {OUTPUT} >=0;

param q_in {INPUT} >= 0;
param q_out_min {OUTPUT} >= 0;
```

```
var x {INPUT, OUTPUT} >= 0;

# changed to negative to output mps solvable
maximize profit: sum{i in INPUT, j in OUTPUT} -revenue[j]*x[i,j];

subject to ins{i in INPUT}:
  sum{j in OUTPUT} x[i,j]<=in_max[i];

subject to outs{j in OUTPUT}:
  sum{i in INPUT} x[i,j]<=out_max[j];

subject to quality{j in OUTPUT}:
  sum{i in INPUT} q_in[i]*x[i,j]==q_out_min[j]*sum{i in INPUT} x[i,j];
```

```
set INPUT := a b;
set OUTPUT := c d e f;

param: revenue out_max q_out_min :=
  c 15 12 0.1
  d 21 12 0.13
  e 25 12 0.16
  f 30 12 0.2 ;

param: in_max q_in :=
  a 10 0.4
  b 20 0 ;
```

```
model pooling.mod
data pooling.dat

option solver cplex;
option cplex_options 'timing=1';
option presolve 0;
option show_stats 1;


problem pooling: x, profit, ins, outs, quality;


expand pooling;
write mpooling;

solve;
```

```
display x;
printf{i in INPUT} "%s␣%d\n", i, sum{j in OUTPUT} x[i,j];
printf{j in OUTPUT} "%s␣%d\n", j, sum{i in INPUT} x[i,j];
```

```
maximize profit:
        15*x['a','c'] + 21*x['a','d'] + 25*x['a','e'] + 30*x['a','f'] +
        15*x['b','c'] + 21*x['b','d'] + 25*x['b','e'] + 30*x['b','f'];

subject to ins['a']:
        x['a','c'] + x['a','d'] + x['a','e'] + x['a','f'] <= 10;

subject to ins['b']:
        x['b','c'] + x['b','d'] + x['b','e'] + x['b','f'] <= 20;

subject to outs['c']:
        x['a','c'] + x['b','c'] <= 12;

subject to outs['d']:
        x['a','d'] + x['b','d'] <= 12;

subject to outs['e']:
        x['a','e'] + x['b','e'] <= 12;

subject to outs['f']:
        x['a','f'] + x['b','f'] <= 12;

subject to quality['c']:
        0.3*x['a','c'] - 0.1*x['b','c'] >= 0;

subject to quality['d']:
        0.27*x['a','d'] - 0.13*x['b','d'] >= 0;

subject to quality['e']:
        0.24*x['a','e'] - 0.16*x['b','e'] >= 0;

subject to quality['f']:
        0.2*x['a','f'] - 0.2*x['b','f'] >= 0;


8 variables, all linear
10 constraints, all linear; 24 nonzeros
        10 inequality constraints
1 linear objective; 8 nonzeros.

CPLEX 12.6.0.0: timing=1

Times (seconds):
Input =  0.004
Solve =  0
Output = 0
```

```
CPLEX 12.6.0.0: optimal solution; objective 630
10 dual simplex iterations (2 in phase I)
x :=
a c   0
a d   3.9
a e   4.8
a f   1.3
b c   0
b d   8.1
b e   7.2
b f   1.3
;
a 10
b 16
c 0
d 12
e 12
f 2
```

Subtask 6.b

In order to be faster in serving his clients, our bartender decides to create and maintain intermediate drink containers. Depending on the structure of the network, this may anticipate part of the drink dilution process and delay another part until final drinks have to be blended to specification.
Now the raw materials are input of the pooling container only and customers receive their drinks directly from the pooling container or by mixing the drinks of the pooling containers. It is assumed that the flow of intermediate products into the pooling containers equals the flow required to blend the final products.
The bartender decides to experiment with the use of one only pooling container. The situation is represented in Figure 2.
Unfortunately, writing an LP model for this situation is known to be an NP hard problem. Can you nevertheless write a nonlinear model for this situation? Intuitively, when solved to optimality, will the total profit increase or decrease with respect to Subtask 6.a?

Solution We model the general case with $k > 0$ pools. Let K be the set of pools. Since we do not know the concentration of alcohol at the Pool we need to represent it as a variable y_k.

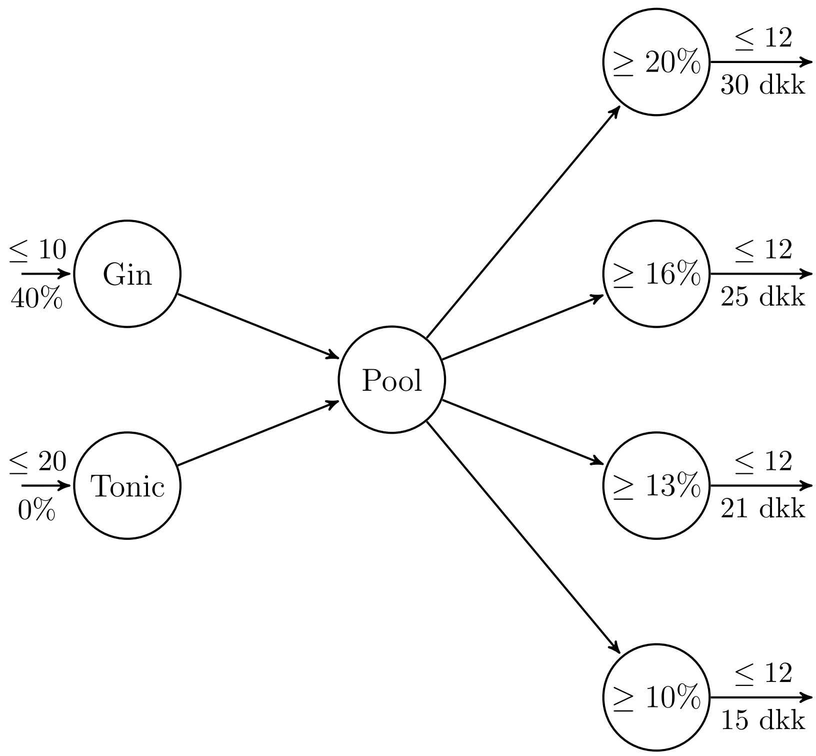


Figure 2: The situation in Subtask 6.b.

$$\max \sum_{j\in J}\sum_{k\in K} r_j x_{kj} \tag{16}$$

$$\sum_{k\in K} x_{ik} \le b_i \qquad \forall i \in I \tag{17}$$

$$\sum_{k\in K} x_{kj} \le d_j \qquad \forall j \in J \tag{18}$$

$$\sum_{i\in I} x_{ik} = \sum_{j\in J} x_{kj} \qquad \forall k \in K \tag{19}$$

$$\sum_{i\in I} p_i x_{ik} = y_k \sum_{i\in I} x_{ik} \qquad \forall k \in K \tag{20}$$

$$\sum_{k\in K} y_k x_{kj} \ge p_j \sum_{k\in K} x_{kj} \qquad \forall j \in J \tag{21}$$

$$x_{ij} \ge 0 \qquad \forall i \in I, j \in J \tag{22}$$

The total profit will decrease.

Task 7 IP Formulation (points: 10)

In two days the world cup in Brazil will finally start! Prandelli, the trainer of the Italian team, had to plan the content of the training sessions for his team. The schedule is two sessions per day, morning and afternoon, for a time horizon of n days; optimistically he planned for the whole duration of the competition, that is, one month. Prandelli has a data base of activities I to propose during the sessions. Associated to each activity i in I there are three numbers that quantify the training effect: *technical effect* a_{i1}, *tactical effect* a_{i2} and *physical effect* a_{i3}. Moreover each activity has a duration t_i expressed in minutes.

Each activity can be proposed more than once throughout the sessions. The following constraints must be taken into account:

i. Each activity can be scheduled only once per day.

ii. Each session lasts 2 hours, that is, 120 minutes.

iii. There is a pre-determined limit on the physical effort b_{3s} for each session s. This value varies through the sessions. For example, sessions in the days just before and just after a match will contain less physical effort.

iv. Considering the technical and tactical aspects as the main strengths of his team, Prandelli wants to ensure that each session has at least a desired amount of effort in those attributes. Let b_{1s} and b_{2s} define these requirements for each session.

v. Some activities are incompatible, that is, they cannot be scheduled in the same day. Let A be a set of pairs of incompatible activities. In each day and for each pair of activities $(i, j) \in A$, if activity i is selected then activity j cannot be prescribed in the same day and viceversa.

vi. Some activities that are physically particularly demanding can be scheduled only with at least 3 days distance. Let $H \subseteq S$[2] be the set that identifies these activities.

vii. For the tactical activities, that is, those with a non zero tactical attribute, that is, $a_{i2} > 0$, the effect must be non-decreasing during the month of training. [Hint: define P to be the set of pairs of activities such that $(i, j) \in P$ if and only if $0 < a_{i2} < a_{j2}$. Thus, if $(i, j) \in P$ then i must be scheduled before j. Further, define a variable to contain for an activity i the last session in which it is scheduled...]

The objective is to maximize the number of activities proposed.

Formulate the problem as an integer linear problem.
[Hint: Let S be the set of session indexed by s. For variables you need at least the variable x_{is} to indicate if activity i is scheduled in session s. You might need more variables to make the IP formulation.]

[2] Update 26.06.2014: $H \subseteq I$

Solution Parameters:

- S set of sessions indexed by s
- D set of days indexed by d
- $S_d \subseteq S$ set of sessions in day d

Variables

- $x_{is} = 1$ if activity i is scheduled in session s else 0
- $y_i = 1$ if activity i is scheduled any where in the set of sessions S else 0
- $\overline{z}_i$ an integer variable indicating the latest session in which activity i is scheduled.
- $\underline{z}_i$ an integer variable indicating the earliest session in which activity i is scheduled.

$$\max \sum_i y_i \tag{23}$$

$$\text{s.t.} \sum_{s \in S_d} x_{is} \leq 1 \qquad \forall i \in I, d \in D \tag{24}$$

$$\sum_i t_i x_{is} \leq 120 \qquad \forall s \in S \tag{25}$$

$$\sum_i a_{3i} x_{is} \leq b_{3s} \qquad \forall s \in S \tag{26}$$

$$\sum_i a_{2i} x_{is} \geq b_{2s} \qquad \forall s \in S \tag{27}$$

$$\sum_i a_{1i} x_{is} \geq b_{1s} \qquad \forall s \in S \tag{28}$$

$$\sum_{s \in S_d} x_{is} + \sum_{s \in S_d} x_{js} \leq 1 \qquad \forall (i,j) \in A, d \in D \tag{29}$$

$$\sum_{s \in S_d} x_{is} + \sum_{d' \in \{d+1, d+2\}} \sum_{s \in S_{d'}} x_{is} \leq 1 \qquad \forall i \in H, d \in \{1..|D|-2\} \tag{30}$$

$$\overline{z}_i \geq s x_{is} \qquad \forall i \in P, s \in S \tag{31}$$

$$\underline{z}_i \leq s x_{is} \qquad \forall i \in P, s \in S \tag{32}$$

$$\overline{z}_i \leq \underline{z}_j \qquad \forall (i,j) \in P \tag{33}$$

$$\sum_{s \in S} x_{is} \leq |S| y_i \qquad \forall i \in I \tag{34}$$

$$x_{is} \in \{0,1\} \qquad \forall i \in I, s \in S \tag{35}$$

$$y_i \in \{0,1\} \qquad \forall i \in I \tag{36}$$

$$\overline{z}_i, \underline{z}_i \in \mathbb{Z}_0^+ \qquad \forall i \in I \tag{37}$$

Task 8 Preprocessing and TUM (points: 7)

Using preprocessing rules only, eliminate as many rows and columns as possible from the following set partitioning problem until you are left with a totally unimodular matrix or it is possible to determine the optimal solution. Remember to justify your claims by showing how a certain result from the course applies. For example, it is not enough to claim that a matrix is TUM but you have to show why.

$$\begin{aligned} \min \quad z &= \begin{bmatrix} 5 & 4 & 5 & 4 & 11 & 6 & 10 & 2 & 5 & 3 & 9 & 5 \end{bmatrix} \mathbf{x} \\ A\mathbf{x} &= \mathbf{1} \\ \mathbf{x} &\in \{0,1\}^{12} \end{aligned}$$

with A given by:

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{array} \begin{array}{c} \begin{array}{cccccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \end{array} \\ \begin{bmatrix} 1 & 1 & & & 1 & & & & 1 & & & \\ & 1 & & & & 1 & & 1 & & & & 1 \\ & 1 & & 1 & 1 & & 1 & & & 1 & 1 & 1 \\ 1 & & 1 & 1 & & & & & & & & \\ & & & & & 1 & & & & & & \\ & 1 & & & 1 & & 1 & & 1 & & 1 & \\ 1 & & & 1 & & & & & & & & \\ & & & & 1 & & & & & 1 & & 1 \end{bmatrix} \end{array}$$

(Note: in the online version you find the matrix in ASCII form.)

Solution

```
|---+---+---+---+---+---+-----+---+-----+---+---+----+----+----|
|   |   | 1 | 2 | 3 | 4 |   5 | 6 |   7 | 8 | 9 | 10 | 11 | 12 |
|   |   | 5 | 4 | 5 | 4 |  11 | 6 |  10 | 2 | 5 |  3 |  9 | 5  |
|---+---+---+---+---+---+-----+---+-----+---+---+----+----+----|
|   |   |   | x | x |   |   x | + |   x | x |   |    |    |    |
|---+---+---+---+---+---+-----+---+-----+---+---+----+----+----|
| 1 |   | 1 | 1 |   |   |   1 |   |     |   | 1 |    |    |    |
| 2 | x |   | 1 |   |   |     | 1 |     | 1 |   |    |    |  1 |
| 3 |   |   | 1 |   | 1 |   1 |   |   1 |   |   |  1 |  1 |  1 |
| 4 | x | 1 |   | 1 | 1 |     |   |     |   |   |    |    |    |
| 5 | x |   |   |   |   |     | 1 |     |   |   |    |    |    |
| 6 |   |   | 1 |   |   |   1 |   |   1 |   | 1 |    |  1 |    |
| 7 |   | 1 |   |   | 1 |     |   |     |   |   |    |    |    |
| 8 |   |   |   |   |   |   1 |   |     |   |   |  1 |    |  1 |
|---+---+---+---+---+---+-----+---+-----+---+---+----+----+----|
```

Fix col 6 because only one that can cover row 5. Hence remove row 5 and row 2. Since row 2 is covered neither col 2 nor col 8 can be chosen. Hence we remove them too.

```
|---+---+---+---+---+----+----+---+----+----+----|
|   |   | 1 | 3 | 4 |  5 |  7 | 9 | 10 | 11 | 12 |
|   |   | 5 | 5 | 4 | 11 | 10 | 5 |  3 |  9 | 5  |
|---+---+---+---+---+----+----+---+----+----+----|
|   |   |   | x |   |  x |  x |   |    |    |    |
|---+---+---+---+---+----+----+---+----+----+----|
| 1 |   | 1 |   |   |  1 |    | 1 |    |    |    |
| 3 |   |   |   | 1 |  1 |  1 |   |  1 |  1 |  1 |
| 4 | x | 1 | 1 | 1 |    |    |   |    |    |    |
| 6 |   |   |   |   |  1 |  1 | 1 |    |  1 |    |
| 7 |   | 1 |   | 1 |    |    |   |    |    |    |
| 8 |   |   |   |   |  1 |    |   |  1 |    |  1 |
|---+---+---+---+---+----+----+---+----+----+----|
```

Remove row 4 because dominated by row 7. Consequently remove col 3 because it will be covered by col 1 or 4.

```
|---+---+---+---+----+----+---+----+----+----|
|   |   | 1 | 4 |  5 |  7 | 9 | 10 | 11 | 12 |
|   |   | 5 | 4 | 11 | 10 | 5 |  3 |  9 | 5  |
|---+---+---+---+----+----+---+----+----+----|
|   |   |   |   |  x |  x |   |    |    |    |
|---+---+---+---+----+----+---+----+----+----|
| 1 |   | 1 |   |  1 |    | 1 |    |    |    |
| 3 |   |   | 1 |  1 |  1 |   |  1 |  1 |  1 |
| 6 |   |   |   |  1 |  1 | 1 |    |  1 |    |
| 7 |   | 1 | 1 |    |    |   |    |    |    |
| 8 |   |   |   |  1 |    |   |  1 |    |  1 |
|---+---+---+---+----+----+---+----+----+----|
```

Remove col 5 because it covers the same as the set of cols 9 and 10 which, if chosen together, cost less than col 5

```
|---+---+---+---+----+---+----+----+----|
|   |   | 1 | 4 |  7 | 9 | 10 | 11 | 12 |
|   |   | 5 | 4 | 10 | 5 |  3 |  9 | 5  |
|---+---+---+---+----+---+----+----+----|
|   |   |   |   |  x |   |    |    |    |
|---+---+---+---+----+---+----+----+----|
| 1 |   | 1 |   |    | 1 |    |    |    |
| 3 |   |   | 1 |  1 |   |  1 |  1 |  1 |
| 6 |   |   |   |  1 | 1 |    |  1 |    |
| 7 |   | 1 | 1 |    |   |    |    |    |
| 8 |   |   |   |    |   |  1 |    |  1 |
|---+---+---+---+----+---+----+----+----|
```

Remove col 7 because equal to col 11 but 11 costs less.

```
|---+---+---+---+---+----+----+----|
|   |   | 1 | 4 | 9 | 10 | 11 | 12 |
```

```
|   |   | 5 | 4 | 5 |  3 |  9 | 5  |
|---+---+---+---+---+----+----+----|
|   |   |   |   |   |    |    |    |
|---+---+---+---+---+----+----+----|
| 1 |   | 1 |   | 1 |    |    |    |
| 3 |   |   | 1 |   |  1 |  1 |  1 |
|---+---+---+---+---+----+----+----|
| 6 |   |   |   | 1 |    |  1 |    |
| 7 |   | 1 | 1 |   |    |    |    |
| 8 |   |   |   |   |  1 |    |  1 |
|---+---+---+---+---+----+----+----|
```

We are left with a TUM. The partition is indicated by the line.