Department of Mathematics and Computer Science
University of Southern Denmark, Odense

October 2, 2025
Marco Chiarandini

DM545/DM871 – Linear and integer programming

Sheet 4, Autumn 2025

Solution:

Included.
Exercises with the symbol $^+$ are to be done at home before the class. Exercises with the symbol * will be tackled in class and should be at least read at home. The remaining exercises are left for self training after the exercise class.

Exercise 1$^+$

Solve the systems $\mathbf{y}^T E_1 E_2 E_3 E_4 = [1\ 2\ 3]$ and $E_1 E_2 E_3 E_4 \mathbf{d} = [1\ 2\ 3]^T$ with

$$E_1 = \begin{bmatrix} 1 & 3 & 0 \\ 0 & 0.5 & 0 \\ 0 & 4 & 1 \end{bmatrix} \qquad E_2 = \begin{bmatrix} 2 & 0 & 0 \\ 1 & 1 & 0 \\ 4 & 0 & 1 \end{bmatrix} \qquad E_3 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix} \qquad E_4 = \begin{bmatrix} -0.5 & 0 & 0 \\ 3 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

Solution:

This exercise is to show that the two systems can be solved quite easily. Let's take first $\mathbf{y}^T E_1 E_2 E_3 E_4 = [1\ 2\ 3]$, we use the backward transformation and solve the sequence of linear systems:

$$\mathbf{u}^T E_4 = [1\ 2\ 3], \quad \mathbf{v}^T E_3 = \mathbf{u}^T, \quad \mathbf{w}^T E_2 = \mathbf{v}^T, \quad \mathbf{y}^T E_1 = \mathbf{w}^T$$

$$\mathbf{u}^T \begin{bmatrix} -0.5 & 0 & 0 \\ 3 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} = [1, 2, 3]$$

Since the eta matrices have always one 1 in two columns then the solution can be read up easily. From the third column we find $u_3 = 3$. From the second column, we find $u_2 = 2$. Substituting in the first column, we find $-0.5u_1 + 3 * 2 + 1 * 3 = 1$, which yields $u_1 = 16$. The next syestem is:

$$\mathbf{v}^T \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix} = [16, 2, 3]$$

From the first column we get $v_1 = 16$, from the second column $v_2 = 2$ from the last column $v_3 = -19$. The next:

$$\mathbf{w}^T \begin{bmatrix} 2 & 0 & 0 \\ 1 & 1 & 0 \\ 4 & 0 & 1 \end{bmatrix} = [16, 2, -19]$$

which gives $\mathbf{w} = [45, 2, -19]$. The next:

$$\mathbf{y}^T \begin{bmatrix} 1 & 3 & 0 \\ 0 & 0.5 & 0 \\ 0 & 4 & 1 \end{bmatrix} = [45, 2, -19]$$

from we finally find $\mathbf{d} = [119/2, 2, -19]$.
We can carry out the operations in Python for cross checking the results but the function `linalg.solve` would not exploit the advantages of the eta matrices when solving the system of linear equations.

```
import numpy as np

E4=np.array([[-0.5,0,0],[3,1,0],[1,0,1]])
E3=np.array([[1,0,1],[0,1,3],[0,0,1]])
E2=np.array([[2,0,0],[1,1,0],[4,0,1]])
E1=np.array([[1,3,0],[0,0.5,0],[0,4,1]])
u=np.linalg.solve(E4.T,np.array([1,2,3]))
print(u)
v=np.linalg.solve(E3.T,u)
print(v)
w=np.linalg.solve(E2.T,v)
print(w)
d=np.linalg.solve(E2.T,w)
print(d)
```

```
[16.  2.  3.]
[ 16.   2. -19.]
[ 45.   2. -19.]
[ 59.5  2.  -19. ]
```

Exercise 2* Sensitivity Analysis and Revised Simplex

A furniture-manufacturing company can produce four types of products using three resources.

- A bookcase requires three hours of work, one unit of metal, and four units of wood and it brings in a net profit of 19 Euro.
- A desk requires two hours of work, one unit of metal and three units of wood, and it brings in a net profit of 13 Euro.
- A chair requires one hour of work, one unit of metal and three units of wood and it brings in a net profit of 12 Euro.
- A bedframe requires two hours of work, one unit of metal, and four units of wood and it brings in a net profit of 17 Euro.
- Only 225 hours of labor, 117 units of metal and 420 units of wood are available per day.

In order to decide how much to make of each product so as to maximize the total profit, the managers solve an LP problem.

1) Write the mathematical programming formulation of the problem.

Solution:

$$\begin{aligned} \max \quad & 19x_1 + 13x_2 + 12x_3 + 17x_4 \\ & 3x_1 + 2x_2 + x_3 + 2x_4 \leq 225 \\ & x_1 + x_2 + x_3 + x_4 \leq 117 \\ & 4x_1 + 3x_2 + 3x_3 + 4x_4 \leq 420 \\ & x_1, x_2, x_3, x_4 \geq 0 \end{aligned}$$

2) With the help of a computational environment such as Python for carrying out linear algebra operations, write the optimal tableau, which has x_1, x_3 and x_4 in basis.

Do this task:

- first, using the original simplex method
- second, using the revised simplex. In this case, start by writing A_B, A_N, then calculate $A_B^{-1}A_N$, and the finally derive the optimal tableau and verify that the solution is indeed optimal.

Solution:

The initial tableau is:

```
|-----+-----+-----+-----+-----+-----+-----+----+-----|
| x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | -z |   b |
|-----+-----+-----+-----+-----+-----+-----+----+-----|
|   3 |   2 |   1 |   2 |   1 |   0 |   0 |  0 | 225 |
|   1 |   1 |   1 |   1 |   0 |   1 |   0 |  0 | 117 |
|   4 |   3 |   3 |   4 |   0 |   0 |   1 |  0 | 420 |
|  19 |  13 |  12 |  17 |   0 |   0 |   0 |  1 |   0 |
|-----+-----+-----+-----+-----+-----+-----+----+-----|
```

We know that there will be 3 variables in basis. The text of the problem tells us which these 3 variables are: 1, 3, 4. Hence,

$$A_B = \begin{bmatrix} 3 & 1 & 2 \\ 1 & 1 & 1 \\ 4 & 3 & 4 \end{bmatrix} \qquad A_N = \begin{bmatrix} 2 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 3 & 0 & 0 & 1 \end{bmatrix}$$

We can calculate $A_B^{-1}A_N$ in Python or in R:

```
> B=matrix(c(3,1,2,1,1,1,4,3,4),byrow=TRUE,ncol=3)
> B1=solve(B)
> B%*%B1  # check to make sure it is correct!
     [,1] [,2] [,3]
[1,]    1    0    0
[2,]    0    1    0
[3,]    0    0    1
> N=matrix(c( 2, 1, 0, 0, 1, 0, 1, 0, 3, 0, 0, 1),ncol=4,byrow=TRUE)
> B1%*%N
     [,1] [,2] [,3] [,4]
[1,]    1    1    2   -1
[2,]    1    0    4   -1
[3,]   -1   -1   -5    2
> cN=c(13,0,0,0)
> cB=c(19,12,17)
> cN-cB%*%B1%*%N
     [,1] [,2] [,3] [,4]
[1,]   -1   -2   -1   -3
> cB%*%B1%*%c(225,117,420)
> 1827
```

This code gives us:

$$\bar{A} = A_B^{-1}A_N = \begin{bmatrix} 1 & 1 & 2 & -1 \\ 1 & 0 & 4 & -1 \\ -1 & -1 & -5 & 2 \end{bmatrix} \qquad x_B^* = \begin{bmatrix} x_1 \\ x_3 \\ x_4 \end{bmatrix} = A_B^{-1}b = \begin{bmatrix} 39 \\ 48 \\ 30 \end{bmatrix}$$

$$\bar{c}_N = \begin{bmatrix} \bar{c}_2 & \bar{c}_5 & \bar{c}_6 & \bar{c}_7 \end{bmatrix} = \begin{bmatrix} -1 & -2 & -1 & -3 \end{bmatrix}$$

and we can write the final tableau as:

```
|-----+-----+-----+-----+-----+-----+-----+----+-------|
| x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | -z |     b |
|-----+-----+-----+-----+-----+-----+-----+----+-------|
|   1 |   1 |   0 |   0 |   1 |   2 |  -1 |  0 |    39 |
|   0 |   1 |   1 |   0 |   0 |   4 |  -1 |  0 |    48 |
|   0 |  -1 |   0 |   1 |  -1 |  -5 |   2 |  0 |    30 |
|   0 |  -1 |   0 |   0 |  -2 |  -1 |  -3 |  1 | -1827 |
|-----+-----+-----+-----+-----+-----+-----+----+-------|
```

Since all reduced costs are negative then the tableau and the corresponding solution are optimal.

3) What is the increase in price (reduced cost) that would make product x_2 worth to be produced?

Solution:

The increase in price of a quantity strictly larger than 1 would make the product 2 worth being produced. Indeed, let $c'_2 = c_2 + \delta$ be the new price. We know that the coefficient in the objective function goes in the reduced cost calculation multiplied by 1. Hence, to have a positive reduced cost we have:

$$-1 + \delta > 0 \qquad \Longrightarrow \qquad \delta > 1$$

We could also recalcuate the reduced cost from scratch using the multipliers π: $c'_2 + \sum_{i=1}^{3} \pi_i a_{i2}$. The value of π_i are read from the final tableau and they correspond to the reduced costs of the slack variables, ie, $(-2, -1, -3)$.

4) What is the marginal value (shadow price) of an extra hour of work or amount of metal and wood?

Solution:

The marginal values are the values of the dual variable y_1, y_2, y_3. From the strong duality theorem, we know that $y_i = -\pi_i = -\bar{c}_{n+i}$, $i = 1..m$. Hence, $\mathbf{y} = (2, 1, 3)$.

An extra hour of work has marginal value of 2, that is, having one unit more of work would improve the revenue by 2. For the other two resources the marginal values are 1 and 3, respectively.

We can cross check these conclusions: by the complementary slackness theorem, the fact that all three dual variables are strictly positive indicates that all three constraints in the primal are active≡tight≡ binding. Hence, it makes sense to have that an increase in the capacity of those constraints implies an increase in the profit. The conclusion that all three constraints are tight can be also reached by the fact that the slack variables are 0 in the final tableau. If some constraint was not tight, then the marginal value of the corresponding resource would be zero since an increase in its capacity does not imply an immediate improvement in total profit.

5) Are all resources totally utilized, i.e. are all constraints "binding", or is there slack capacity in some of them? Answer this question in the light of the complementary slackness theorem.

Solution:

Since all dual variables are strictly larger than zero, then all constraints are binding. Indeed for the complementary slackness theorem, we have that:

$$\left(b_i - \sum_{j=1}^{n} a_{ij} x_j^* \right) y_i^* = 0, \quad i = 1, \ldots, m$$

6) From the economical interpretation of the dual why product x_2 is not worth producing? What is its imputed cost?

Solution:

It is not worth producing 2 because $\sum_i y_i a_{i2} > c_2$, that is, we are better off selling the raw materials to produce the product. Indeed y_i is the price of one unit of resource i and a_{i2} is the amount of i necessary to produce 2.

$$\sum_i y_i a_{i2} = 2 * (2) + 1 * (1) + 3 * (3) = 14 > 13$$

Perform a sensitivity analysis for the following variants:

7) The net profit brought in by each desk increases from 13 Euro to 15 Euro.

Solution:

We saw earlier that if the price of product 2 increases by more than 1 then the reduced cost becomes positive and it enters the basis. We can iterate the revised simplex as follows:

Step 1 and 2 to determine the entering varible are already done in the point a) above.

We need to do Step 3 to determine the leaving variable: we need to find the constraint that limit the increase of x_2, θ. We solve first $A_B d = a$ in d. Here, $\mathbf{a}$ is the column of the matrix A (augmented with the slack variables) from the initial tableau corresponding to the entering variable x_2. We use the inverse of A_B calculated earlier in a) above in R:

```
> B1%*%c(2,1,3)
     [,1]
[1,]    1
[2,]    1
[3,]   -1
```

that is

$$\mathbf{d} = A_B^{-1}\mathbf{a} = A_B^{-1}\begin{bmatrix}2\\1\\3\end{bmatrix} = \begin{bmatrix}1\\1\\-1\end{bmatrix}$$

Then the new solutioon x_B is derived from the old one by means of $\mathbf{d}$ and the increase θ:

$$x_B = \begin{bmatrix}x_1\\x_3\\x_4\end{bmatrix} = \begin{bmatrix}39\\48\\30\end{bmatrix} - \begin{bmatrix}1\\1\\-1\end{bmatrix}\theta \geq 0$$

The increase θ must be such that the value of the variables still remains feasible, ie, $x_i \geq 0$. Hence $\theta \leq 39$ and the leaving variable is x_1, since it is the one that goes to zero. The new solutions is

$$x_B = \begin{bmatrix}x_2\\x_3\\x_4\end{bmatrix} == \begin{bmatrix}39\\48\\30\end{bmatrix} - \begin{bmatrix}1\\1\\-1\end{bmatrix}\theta = \begin{bmatrix}39-39\\48-39\\30+39\end{bmatrix} = \begin{bmatrix}0\\9\\69\end{bmatrix}$$

and the objective value:

```
> c=c(19,15,12,17)
> c%*%c(0,39,9,69)
     [,1]
[1,] 1866
```

8) The availability of metal increases from 117 to 125 units per day

Solution:

This is a change in the RHS term of constraint 2. The optimality of the current solution does not change, since all reduced costs stay negative, but we need to check if it is still feasible. We need to look at the final tableau and recompute the b terms of all constraints. We can do this with $A_B^{-1}b$:

```
> b=c(225,125,420)
> B1%*%b
     [,1]
[1,]   55
[2,]   80
[3,]  -10
```

The last cosntraint becomes negative, hence we need to iterate with the dual simplex.

9) The company may also produce coffee tables, each of which requires three hours of work, one unit of metal, two units of wood and bring in a net profit of 14 Euro.

Solution:

We need to check if the reduced cost of the new variable would become positive by computing $c_0 + \sum_i \pi_i a_{ij}$:

```
> 14-3*2-1*1-2*3
[1] 1
```

which is positive, hence we need to iterate as done in point 1).

10) The number of chairs produced must be at most five times the numbers of desks

Solution:

This corresponds to introduce a new constraint: $x_3 \leq 5x_2$. In the new standard form we have a new slack variable x_8. Adding the constraint in the tableau and bringing back the tableau in canonical standard form we observe that a RHS term becomes negative. Hence, we need to iterate with the dual simplex. After on iteration with the dual simplex, the final tableau becomes:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 4/3 & -5/6 & 1/6 & 0 & 31 \\ 0 & 1 & 0 & 0 & 0 & 2/3 & -1/6 & -1/6 & 0 & 8 \\ 0 & 0 & 0 & 1 & -1 & -13/3 & 11/6 & -1/6 & 0 & 38 \\ 0 & 0 & 1 & 0 & 0 & 10/3 & -5/6 & 1/6 & 0 & 40 \\ 0 & 0 & 0 & 0 & -2 & -1/3 & -19/6 & -1/6 & 1 & -1819 \end{bmatrix}$$

If after the introduction of the constraint the current solution stayed feasible then, we would have needed to check whether it was also optimal. We can either repeat the steps done at part 1 above to compute the new reduced costs or we can include the new row in the final tableau and proceed to put the tableau in canonical form. Then we look at the value of the reduced costs.

Exercise 3* Traffic Light Control

Automobile traffic from three highways, H1, H2, and H3, must stop and wait for a green light before exiting to a toll road. The tolls are 40 kr, 50 kr, and 60 kr for cars exiting from H1, H2, and H3, respectively. The flow rates from H1, H2, and H3 are 550, 650, and 450 cars per hour. The traffic light cycle may not exceed 2.2 minutes, and the green light on any highway must be at least 22 seconds. The yellow light is on for 10 seconds. The toll gate can handle a maximum of 500 cars per hour. Assuming that no cars move on yellow, and that clearly when a highway has the green signal the other two have the red, determine the optimal green time interval for the three highways that will maximize toll gate revenue per traffic cycle.

Solution:

```
|__________ g1__________|y1|_______________________ r1___________________|
|____________ r2___________|_______ g2___________|y2|_________r2_________|
|_________________________ r3_______________________|_______ g3_______|y3|
|____________________________________________ 2.2 minutes________________|
```

Let g_i, y_i and r_i be the durations of green, yellow and red lights for the cars exiting highway i. All time units are in seconds. No cars move on yellow.

$$\begin{aligned} \max \quad & z = 40(550/3600)g_1 + 50(650/3600)g_2 + 60(450/3600)g_3 \\ & (550/3600)g_1 + (650/3600)g_2 + (450/3600)g_3 \leq (500/3600)(g_1 + g_2 + g_3 + 3 \cdot 10) \\ & g_1 + g_2 + g_3 + 3 \cdot 10 \leq 2.2 \cdot 60 \\ & g_1, g_2, g_3 \geq 22 \end{aligned}$$

The objective function mesures the income using the flow per hour transformed in flow per second. The first constraint ensures that the limit of 500 cars per hours is not exceeded. The second ensure the duration of the cycle. The last constraints ensure that the duration of the green light satisfies the minimum requirement.

```
#!/usr/bin/python3
import pyomo.environ as po

model = po.ConcreteModel("traffic_light")

infinity = float('inf')
# declare decision variables
model.g1 = po.Var(bounds=(22,infinity),domain=po.NonNegativeReals)
model.g2 = po.Var(bounds=(22,infinity),domain=po.NonNegativeReals)
model.g3 = po.Var(bounds=(22,infinity),domain=po.NonNegativeReals)

# declare objective
model.profit = po.Objective(
    expr = 40*(550/3600)*model.g1+50*(650/3600)*model.g2+60*(450/3600)*model.g3,
    sense = po.maximize)

# declare constraints
model.capacity = po.Constraint(expr = (550/3600)*model.g1+(650/3600)*model.g2+(450/3600)*
     model.g3<= (500/3600)*(model.g1+model.g2+model.g3+3* 10))
model.max_cycle = po.Constraint(expr = model.g1+model.g2+model.g3+3*10<= 2.2 * 60)


model.pprint()
model.write("traffic_light.lp")

# solve
solver = po.SolverFactory('gurobi')
results = solver.solve(model, tee=True)


if results.solver.termination_condition == po.TerminationCondition.optimal:
    for v_data in model.component_data_objects(po.Var, descend_into=True):
        print("Found: "+v_data.name+", value = "+str(po.value(v_data)))
```

```
max
x4: +6.1111111111111116 x1 +9.0277777777777768 x2 +7.5 x3

s.t.

c_u_x5_:
+0.013888888888888895 x1 +0.041666666666666657 x2 -0.013888888888888895 x3 <= 4.166666666666667

c_u_x6_:
+1 x1 +1 x2 +1 x3 <= 102

c_e_ONE_VAR_CONSTANT:
ONE_VAR_CONSTANT = 1.0

bounds
   22 <= x1 <= +inf
   22 <= x2 <= +inf
   22 <= x3 <= +inf
end
```

```
3 Var Declarations
    g1 : Size=1, Index=None
        Key  : Lower : Value : Upper : Fixed : Stale : Domain
        None :    22 :  None :   inf : False :  True : NonNegativeReals
    g2 : Size=1, Index=None
        Key  : Lower : Value : Upper : Fixed : Stale : Domain
        None :    22 :  None :   inf : False :  True : NonNegativeReals
    g3 : Size=1, Index=None
        Key  : Lower : Value : Upper : Fixed : Stale : Domain
        None :    22 :  None :   inf : False :  True : NonNegativeReals

1 Objective Declarations
    profit : Size=1, Index=None, Active=True
        Key  : Active : Sense    : Expression
        None :   True : maximize : 6.111111111111112*g1 + 9.027777777777777*g2 + 7.5*g3

2 Constraint Declarations
    capacity : Size=1, Index=None, Active=True
        Key  : Lower : Body
        None :  -Inf : 0.15277777777777778*g1 + 0.18055555555555555*g2 + 0.125*g3 - 0.13888888888888889
    max_cycle : Size=1, Index=None, Active=True
        Key  : Lower : Body                : Upper : Active
        None :  -Inf : g1 + g2 + g3 + 30 : 132.0 :   True

6 Declarations: g1 g2 g3 profit capacity max_cycle
Using license file /Library/gurobi900/gurobi.lic
Academic license - for non-commercial use only
Reading time = 0.00 seconds
x4: 3 rows, 4 columns, 7 nonzeros
Gurobi Optimizer version 9.0.0 build v9.0.0rc2 (mac64)
Optimize a model with 3 rows, 4 columns and 7 nonzeros
Model fingerprint: 0x5801878b
Coefficient statistics:
  Matrix range     [1e-02, 1e+00]
  Objective range  [6e+00, 9e+00]
  Bounds range     [2e+01, 2e+01]
  RHS range        [1e+00, 1e+02]
Presolve removed 3 rows and 4 columns
Presolve time: 0.01s
Presolve: All rows and columns removed
Iteration    Objective       Primal Inf.    Dual Inf.      Time
       0    8.2305556e+02   0.000000e+00   0.000000e+00      0s

Solved in 0 iterations and 0.01 seconds
Optimal objective  8.230555556e+02
Found: g1, value = 22.0
Found: g2, value = 58.0
Found: g3, value = 22.0
```

Exercise 4* Level Terrain for a New Highway

The Highway Department is planning a new 10-km highway on uneven terrain as shown by the profile in Figure 1. The width of the construction terrain is approximately 50 meters and the differences in height measured in meters can be deduced from the y-axis in the figure. To simplify the situation, the terrain profile can be replaced by a step function as shown in the figure. Using heavy machinery, earth removed from high terrain is pulled with effort to fill low areas. There are also two burrow pits, I and II, located at the ends of the 10-km stretch from which additional earth can be pulled, if needed. Pit I is located before km 0 and has a capacity of 15 000 cubic meters, and pit II is located after km 10 and has a capacity of 11 000 cubic meters. The costs of removing earth from pits I and II are, respectively, 15 kr and 19 kr per cubic meter.

The transportation cost per cubic meter per kilometer is 1.5 kr, and the cost of using heavy machinery to load pulling trucks is 2 kr per cubic meter. This means that, for example, three cubic meters extracted from pit I and transported for 1 km will cost a total of $15 \cdot 3 + 1.5 \cdot 3 \cdot 1 + 2 \cdot 3 = 96$ kr and three cubic meters pulled 1 km away from a hill to a fill area will cost $1.5 \cdot 3 \cdot 1 + 2 \cdot 3 = 55.5$ kr.

Write a linear programming model for finding the minimum cost plan for leveling the 10-km stretch.

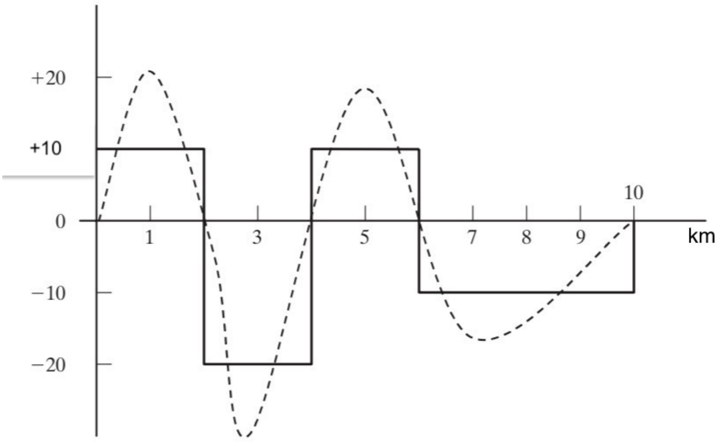

Figure 1:

Solution:

Let $i \in \{1..10\}$ denote the vertical parallelepiped with basis the segment of coordinates $i-1, i$ of length 1000 m. Let 0 indicate the pit I and 11 the pit II. Let a_i be the volume of the parallelepiped i. Let $I^+ \subset \{0..11\}$ be the set of parallelpiped that represent hill areas and $I^- \subset \{0..11\}$ be the set of those repsenting a fill area. The values of a_i can be calculated by calculating the volumes using the measures from the figure after proper tansformation from kilometers to meters and by using the information about the width of 50 meters given in the problem. Let also $a_0 = 15000$ and $a_{11} = 11000$. Let d_{ij} be the distance between the centers of the parallelepiped i and j. The values of d_{ij} are expressed in meters and can be deduced from the figure.

Let $x_{ij} \geq 0, i, j = 1..10$ be the earth moved from parallelepiped i to position j.

$$\min \quad \sum_{i=1}^{10}\sum_{i=1}^{10}(2 + 1.5 \cdot d_{ij})x_{ij} + \sum_{j=1}^{10}(17 + 1.5 d_{0j})x_{0j} + \sum_{j=1}^{10}(21 + 1.5 d_{11,j})x_{11,j}$$

$$\sum_{i=0}^{11} x_{ij} = a_j \qquad \forall j \in I^-$$

$$\sum_{j=1}^{10} x_{ij} \leq a_i \qquad \forall i \in I^+$$

$$x_{ij} \geq 0 \qquad \forall i, j = 1..10$$

The objective function measures the cost of moving earth around. The first component calculates the cost between the areas and the last two the cost of moving earth from the pit I and II, respectively. The first set of constraints ensure that the earth moved to the areas of low terrain is enough to fill the holde. The second set of contraints ensure that the earth moved aways from the high areas does not make them become low areas. Finally, all earth movements are non negative.

Exercise 5 Factory Planning and Machine Maintenance

Tasks 1-3 of Factory Planning and Maintainance Case.