University of Cape Town

STA5071Z

Simulation and Optimization

Optimization Assignment 1

Author:
Raisa Salie

Student Number:
SLXRAI001

January 4, 2022

Contents

Introduction

A new biofuel product called BC is produced using 35% fuel and 65% corn. The demand for BC is given in Table 1. BC is produced and delivered at the beginning of each quarter. Production and delivery occurs on the same day. In a given year, fuel and corn are purchased at the beginning of two month periods with the prices shown in Table 2. In each of these two month periods, one cannot buy more fuel than triple corn. Furthermore, there are life time constraints on the raw materials. Fuel can only be used up to four months after purchase, and corn can only be used up to six months after purchase.

The following models were built.

1. Base Model
   A model was built to determine a raw materials purchase schedule that minimizes the annual total cost of raw materials while satisfying the demand for each quarter.

2. Corn Discount
   A discount is proposed where if one buys 1 000 tonnes or more of corn in a given two month period, a 25% discount is granted. The base model is adjusted to accommodate this circumstance.

3. Fuel Purchase Constraint
   A constraint is imposed where if more than 520 tonnes of fuel are purchased in any given two month period, none may be purchased in the following two month period. The base model is adjusted to accommodate this circumstance.

Table 1: BC quarterly demand (tonnes)

	Q_1	Q_2	Q_3	Q_4
BC demand	1200	1100	1300	1000

Table 2: Cost of fuel and corn for bimonthly periods (in thousands ZAR)

	B_1	B_2	B_3	B_4	B_5	B_6
Fuel (p_i)	2	2.5	2	1	1.5	3
Corn (q_i)	1.5	1	2	1	2	2.5

1 Base Model

1.1 Model Formulation

We define the decision variables as follows.

$$x_{ij} = \text{fuel purchased in period B}_i \text{ for use in Q}_j$$
$$y_{ij} = \text{corn purchased in period B}_i \text{ for use in Q}_j$$

where

$$i \in \{1, .., 6\},$$
$$j \in \{1, .., 4\}$$

To incorporate the lifetime constraints of fuel and corn into the model we will define the following sets. Let J_{xi} be the set of all indices for the quarters in which fuel can be bought in B_i. Let J_{yi} be likewise for corn. Additionally, let I_{xj} be the set of indices for the periods in which fuel used in Q_j can be bought. Let I_{yj} be likewise for corn. The the lifetime constraints of both fuel and corn can be captured in the following sets.

$$\begin{array}{lll}
J_{x1} = \{1, 2\} & J_{x2} = \{2\} & J_{x3} = \{3\} \\
J_{x4} = \{3, 4\} & J_{x5} = \{4\} & J_{x6} = \{1\} \\
J_{y1} = \{1, 2\} & J_{y2} = \{2, 3\} & J_{y3} = \{3, 4\} \\
J_{y4} = \{3, 4\} & J_{y5} = \{1, 4\} & J_{y6} = \{1, 2\} \\
I_{x1} = \{1, 6\} & I_{x2} = \{1, 2\} & I_{x3} = \{3, 4\} \\
I_{x4} = \{4, 5\} & I_{x5} = \emptyset & I_{x6} = \emptyset \\
I_{y1} = \{1, 5, 6\} & I_{y2} = \{1, 2, 6\} & I_{y3} = \{2, 3, 4\} \\
I_{y4} = \{3, 4, 5\} & I_{x5} = \emptyset & I_{y6} = \emptyset
\end{array}$$

These sets reveal that the decision variables under consideration for this problem (those which may be nonzero) are given by

$$\begin{array}{cccc}
x_{11}, x_{61}, & x_{12}, x_{22}, & x_{33}, x_{43}, & x_{44}, x_{54}, \\
y_{11}, y_{51}, y_{61}, & y_{12}, y_{22}, y_{62}, & y_{23}, y_{33}, y_{43}, & y_{34}, y_{44}, y_{54}.
\end{array}$$

Further, we define D_j as the demand for fuel in Q_j, shown in Table 1. Define p_i and q_i as the price of fuel and corn purchased in period B_i respectively (shown in Table 2). The problem was formulated as follows.

$$\text{min. } z = \sum_{j \in \{1,..4\}} \sum_{i \in I_{xj}} p_i x_{ij} + \sum_{j \in \{1,..4\}} \sum_{i \in I_{yj}} q_i y_{ij}$$

s.t.

$$\sum_{i \in I_{xj}} x_{ij} \geq 0.35 D_j \qquad \forall j \in \{1, ..., 4\} \tag{1}$$

$$\sum_{i \in I_{yj}} y_{ij} \geq 0.65 D_j \qquad \forall j \in \{1, ..., 4\} \tag{2}$$

$$\sum_{j \in J_{xi}} x_{ij} - 3 \sum_{j \in J_{yi}} y_{ij} \leq 0 \qquad \forall i \in \{1, .., 6\} \tag{3}$$

$$x_{ij}, y_{ij} \geq 0 \qquad \forall\ i \in \{1, ..., 6\},\ \ j \in \{1, ..., 4\} \tag{4}$$

Given the mixing ratios of fuel and corn, we know how much of each is required for use in each Q_j. This is expressed in constraints (1) and (2). Constraint (6) arises from the manager's constraint on the amount of fuel purchased in each period B_i. We get that $\sum_{j \in J_x i} x_{ij} \leq 3 \sum_{j \in J_{yi}}$, and (3) follows.

1.2 Results

The simplex algorithm was used to solve the linear programme. The model yielded the results shown in Table 3. These results imply the purchase schedule shown in Table 4. The total cost for this purchase schedule is **R 5 795 000**.

Table 3: Model 1 results (nonzero decision variables, in tonnes)

dv	value
x_{11}	420
x_{12}	385
x_{43}	455
x_{44}	350
y_{11}	780
y_{22}	715
y_{23}	845
y_{44}	650

Table 4: Model 1 purchase schedule (in tonnes)

Period	Fuel	Corn
B_1	805	780
B_2		1 560
B_3		
B_4	805	650
B_5		
B_6		

2 Corn Discount Model

2.1 Model Formulation

The following decision variables were introduced in addition to those defined in Question 1.

$$\alpha_i = \text{tonnes of corn purchased in B}i \text{ if below 1000 tonnes}$$
$$\beta_i = \text{tonnes of corn purchased in B}i \text{ if above 1000 tonnes}$$
$$d_i = \begin{cases} 1 & \text{if corn purchased in B}i \text{ exceeds 1000 tonnes} \\ 0 & \text{otherwise} \end{cases}$$

The model was formulated as follows.

$$\text{min. } z = \sum_{j\in\{1,..,4\}} \sum_{i\in I_{xj}} p_i x_{ij} + \sum_{i\in\{1,...6\}} q_i\alpha_i + 0.75 \sum_{i\in\{1,...6\}} q_i\beta_i$$

s.t

$$\sum_{i \in I_{xj}} x_{ij} \geq 0.35 D_j \qquad \forall j \in \{1, ..., 4\} \tag{5}$$

$$\sum_{i \in I_{xj}} y_{ij} \geq 0.65 D_j \qquad \forall j \in \{1, ..., 4\} \tag{6}$$

$$\sum_{j \in J_{xi}} x_{ij} - 3 \sum_{j \in J_{yi}} y_{ij} \leq 0 \qquad \forall i \in \{1, .., 6\} \tag{7}$$

$$\alpha_i + \beta_i - \sum_{j \in J_{yi}} y_{ij} = 0 \qquad \forall i \in \{1, ..., 6\} \tag{8}$$

$$\alpha_i + 1000 d_i \leq 1000 \qquad \forall i \in \{1, ..., 6\} \tag{9}$$

$$\beta_i - 1000 d_i \geq 0 \qquad \forall i \in \{1, ..., 6\} \tag{10}$$

$$\beta_i - U d_i \leq 0 \qquad \forall i \in \{1, ..., 6\} \tag{11}$$

$$x_{ij}, y_{ij}, \alpha_i, \beta_i \geq 0 \qquad \forall \ i \in \{1, ..., 6\}, \ j \in \{1, ..., 4\} \tag{12}$$

$$d_i \in \{0, 1\} \qquad \forall i \in \{1, ..., 6\} \tag{13}$$

$$U \text{a large number} \tag{14}$$

Note that constraints (5) to (7) are inherited from the base model. Constraints (8) impose that the sum of the discounted and non-discounted corn purchased in period B_i are equal to the total corn purchased in B_i. Constraint (9) is equivalent to $\alpha_i \leq 1000(1 - d_i)$. This ensures that if $d_i = 1$ (i.e. more than 1 000 tonnes of corn are purchased in period B_i), $\alpha_i = 0$, but if $d_i = 0$ (i.e. less than 1 000 tonnes of corn are purchased in period B_i, $\alpha_i \leq 1000$. Constraint (10) is equivalent to $\beta_i \geq 1000 d_i$. This ensures that $\beta_i \geq 1000$ when $d_i = 1$ (more than 1000 tonnes of corn are purchased in period B_i), and $\beta_i \geq 0$ if $d_i = 0$. Note that this does not force β_i to zero if $d_i = 0$. Hence, we require U to formulate an upper bound for β_i, $\beta_i \leq U d_i$, such that $\beta_i = 0$ when $d_i = 0$, and $\beta_i \leq U$ if $d_i = 1$. In other words, the use of an arbitrarily large number U allows us to force β_i to zero when corn purchased in period B_i does not exceed 1000 tonnes, while maintaining an unconstrained β_i when corn purchased in period B_i is less than 1 000 tonnes.

2.2 Results

The model yielded the results shown in Table 5, corresponding to the purchase schedule shown in Table 6. The total cost for this purchase schedule is **R 5 040 000**. This means that at most **R 791 000** can be saved with the discount on corn.

Table 5: Model 2 results (nonzero decision variables, in tonnes)

dv	value
x_{11}	420
x_{12}	385
x_{43}	455
x_{44}	350
y_{11}	790
y_{12}	210
y_{22}	505
y_{23}	495
y_{43}	350
y_{44}	650
β_1	1000
β_2	1000
β_4	1000
d_1	1
d_2	1
d_4	1

Table 6: Model 2 purchase schedule (in tonnes)

Period	Fuel	Corn
B_1	805	1000
B_2		1000
B_3		
B_4	805	1000
B_5		
B_6		

3 Fuel Purchase Constraint Model

3.1 Model Formulation

We define a new set of decision variables to be combined with those used in the base model in Question 1.

$$\gamma_i = \begin{cases} 1 & \text{if fuel purchased in B}_i \text{ exceeds 520 tonnes} \\ 0 & \text{otherwise} \end{cases}$$

We also introduce the follwing quantity.

$$x_i = \text{total fuel bought in B}_i \quad \forall i \in \{1, .., 6\}$$

The problem is formulated as follows.

$$\text{min. } z = \sum_{j \in \{1,..,4\}} \sum_{i \in I_{xj}} p_i x_{ij} + \sum_{j \in \{1,..,4\}} \sum_{i \in I_{xj}} q_i x_{ij}$$

s.t.

$$\sum_{i \in I_{xj}} x_{ij} \geq 0.35 D_j \qquad \forall j \in \{1, ..., 4\} \tag{15}$$

$$\sum_{i \in I_{yj}} y_{ij} \geq 0.65 D_j \qquad \forall j \in \{1, ..., 4\} \tag{16}$$

$$\sum_{j \in J_{xi}} x_{ij} - 3 \sum_{j \in J_{yi}} y_{ij} \leq 0 \qquad \forall i \in \{1, .., 6\} \tag{17}$$

$$\sum_{j \in J_{xi}} x_{ij} - x_i = 0 \qquad \forall i \in \{1, .., 6\} \tag{18}$$

$$x_i - 520\gamma_i \geq 0 \qquad \forall i \in \{1, ..., 6\} \tag{19}$$

$$x_i + 520\gamma_{i-1} \leq 520 \qquad \forall i \in \{2, ..., 6\} \tag{20}$$

$$x_1 + U\gamma_6 \leq U \qquad \text{where } i = 1 \tag{21}$$

$$x_{ij}, y_{ij}, x_i \geq 0 \qquad \forall\ i \in \{1..., 6\},\ j \in \{1, ..., 4\} \tag{22}$$

$$\gamma_i \in \{0, 1\} \qquad \forall i \in \{1, ..., 6\} \tag{23}$$

$$U \text{a large number} \tag{24}$$

We retain the constraints pertaining to demand and manager from the base model ((15) - (17)). In constraints (18) we impose that the total fuel purchased in period B_i equal to the sum of all the decision variables x_{ij} purchased in that period. Constraints (19) arise from $x_i \geq 520\gamma_i$ which ensures that if $\gamma_i = 1$, $x_i \geq 520$, but if

$\gamma_i = 0$, $x_i \geq 0$ as desired. Constraints (20) - (21) provide an upper bound for each x_i. These arise from $x_i \leq U(1-\gamma_{i-1})$, which ensures that if $\gamma_{i-1} = 1$, i.e. more than 520 tonnes were purchased in period $i-1$, x_i is fixed to zero. Of course, this needs to be amended where $i = 1$ in constraint (21) to factor in that period 6 from the previous year precedes period one in the current year.

3.2 Results

The third model yielded the results shown in Table 7. Note that this is identical to the solution for the first model, and hence produces the same purchase schedule shown in Table 4. The total cost of the purchase schedule also amounts to **R 5 795 000**. Hence, the constraint imposed by the fuel representative does not affect the purchase schedule or the total cost of raw materials.

Table 7: Model 3 results (nonzero decision variables, in tonnes)

dv	value
x_{11}	420
x_{12}	385
x_{43}	455
x_{44}	350
y_{11}	780
y_{22}	715
y_{23}	845
y_{44}	650
x_1	805
x_4	805

4 R Code

```
rm(list = ls(all=TRUE))
library(Rglpk)
library(xtable)
######################
## Question 1
# demand
D <- c(1200,        1100,        1300,        1000)
# dvs
dv <- paste(c(rep("x", 8), rep("y", 12)),
            c(11,61,12,22,33,43,44,54,11,51,61,12,22,62,23,33,43,34,44,54),
            sep="")
# demand constr
demx <- as.matrix(simple_triplet_matrix(i = c(1,1,2,2,3,3,4,4),
                                        j = c(1:8),
                                        v = rep(1, 8), ncol = 20))
demy <- as.matrix(simple_triplet_matrix(i = c(1,1,1,2,2,2,3,3,3,4,4,4),
                                        j = c(9:20),
                                        v = rep(1, 12), ncol = 20))
dem <- rbind(demx, demy)
colnames(dem) <- dv

# dir
dem_sig <- rep(">=", nrow(dem))
# rhs
dem_rhs <- c(0.35*D, 0.65*D)

# manager's constr
# sumj(xij) - 3sumj(yij) <=0
manx <- as.matrix(simple_triplet_matrix(i = c(1,1,2,3,4,4,5,6),
                                        j = c(1,3,4,5,6,7,8,2),
                                        v = rep(1, 8), ncol = 8))
many <- as.matrix(simple_triplet_matrix(i = c(1,1,2,2,3,3,4,4,5,5,6,6),
                                        j = c(1,4,5,7,8,10,9,11,2,12,3,6),
                                        v = rep(-3, 12), ncol = 12))
# combine
man <-cbind(manx, many)
colnames(man) <- dv
# rhs
man_rhs <- rep(0, nrow(man))
```

```
# dir
man_sig <- rep("<=", nrow(man))

##########
# solve
coeff <- rbind(dem, man)
rhs <- c(dem_rhs, man_rhs)
dir <- c(dem_sig, man_sig)

# obj coeff
obj1 <- c(2,3,2,2.5,2,1,1,1.5,
          1.5,2,2.5,1.5,1,2.5,1,2,1,2,1,2)
names(obj1) <- dv

res1 <- Rglpk_solve_LP(obj = obj1,
                       mat = coeff,
                       rhs = rhs,
                       dir = dir,
                       max = FALSE)
# cost
res1$optimum
# sol
sol1 <- cbind(dv, res1$solution)[-which(res1$solution == 0),]
colnames(sol1) <- c("dv", "value (tonnes)")
xtable(sol1)

############################################
## Question 2
# new dv's: alphas, betas, d's
dv2 <- paste(c(rep("x", 8), rep("y", 12), rep("alpha", 6),
    rep("beta",6), rep("d",6)),
             c(11,61,12,22,33,43,44,54,11,
                51,61,12,22,62,23,33,43,34,44,54,rep(1:6,2), 1:6),
             sep="")
#####
# demand constr
# fuel - x
demx <- as.matrix(simple_triplet_matrix(i = c(rep(1,2), rep(2,2),
                                              rep(3,2), rep(4,2)),
                                        j = c(1:8),
```

```
                                        v = rep(1, 8), ncol = 38))
colnames(demx) <- dv2
# corn - y
demy <- as.matrix(simple_triplet_matrix(i = c(rep(1,3), rep(2,3),
                                              rep(3,3), rep(4,3)),
                                        j = c(9:20),
                                        v = rep(1, 12), ncol = 38))
colnames(demy) <- dv2

demytot <- as.matrix(simple_triplet_matrix(i = c(rep(1,4), rep(2,4), rep(3,4),
                                                 rep(4,4), rep(5,4), rep(6,4)),
                                           j = c(9,12,21,27,
                                                 13,15,22,28,
                                                 16,18,23,29,
                                                 17,19,24,30,
                                                 10,20,25,31,
                                                 11,14,26,32),
                                           v = rep(c(-1,-1,1,1),6),
                                                      ncol = 38))
colnames(demytot) <- dv2
# combine all
dem <- rbind(demx, demy, demytot)
colnames(dem) <- dv2
# rhs
dem_rhs <- c(0.35*D, 0.65*D, rep(0, nrow(demytot)))
#signs
dem_sig <- c(rep(">=", 2*nrow(demx)), rep("==", nrow(demytot)))

# managers constr - amended
man2 <- cbind(man, matrix(0, ncol = 18, nrow=nrow(man)))
colnames(man2) <- dv2

# constr on alphas, betas (binary constrS)
alphs1 <- cbind(matrix(0,ncol=20, nrow = 6), diag(1, 6),
                matrix(0,ncol=6, nrow = 6), diag(1000,6))
colnames(alphs1) <- dv2

bets1 <- cbind(matrix(0,ncol=26, nrow = 6), diag(1, 6), diag(-1000,6))
colnames(bets1) <- dv2
```

```
bets2 <- cbind(matrix(0, ncol=26, nrow=6), diag(1,6), diag(-10^6, 6))
colnames(bets2) <- dv2

# combine all binary constr
bin <- rbind(alphs1, bets1, bets2)
colnames(bin) <- dv2
# rhs
bin_rhs <- c(rep(1000, 6), rep(0,6), rep(0,6))
# dir
bin_sig <- c(rep("<=",6), rep(">=",6), rep("<=",6))
###########
# solve
coeff2 <- rbind(dem, man2, bin)
rhs2 <- c(dem_rhs, man_rhs, bin_rhs)
dir2 <- c(dem_sig, man_sig, bin_sig)

# obj coeff
ycost <- c(1.5,1,2,1,2,2.5)
obj2 <-c(2,3,2,2.5,2,1,1,1.5, # xij
         rep(0,12) , #yij
         ycost, #alphas
         0.75*ycost, #betas
         rep(0,6) #b's
)
names(obj2) <- dv2

res2 <- Rglpk_solve_LP(obj = obj2,
                       mat = coeff2,
                       rhs = rhs2,
                       dir = dir2,
                       max = FALSE,
                       types = c(rep("C",32), rep("B",6)))
# cost
res2$optimum
sol2 <- cbind(dv2, res2$solution)[-which(res2$solution==0),]
colnames(sol2) <- c("dv", "value")
xtable(sol2)

##############################################
## Question 3
```

```
# dvs
dv3 <- paste(c(rep("x", 8), rep("y", 12), rep("g", 6), rep("x",6)),
             c(11,61,12,22,33,43,44,54,11,51,61,12,22,62,23,33,43,
               34,44,54,rep(1:6,2)),
             sep="")
# demand constr
demx <- as.matrix(simple_triplet_matrix(i = c(1,1,2,2,3,3,4,4),
                                        j = c(1:8),
                                        v = rep(1, 8), ncol = 32))
demy <- as.matrix(simple_triplet_matrix(i = c(1,1,1,2,2,2,3,3,3,4,4,4),
                                        j = c(9:20),
                                        v = rep(1, 12), ncol = 32))
dem <- rbind(demx, demy)
colnames(dem) <- dv3

# dir
dem_sig <- rep(">=", 8)
# rhs
dem_rhs <- c(0.35*D, 0.65*D)

# manager's constr
# sumj(xij) - 3sumj(yij) <=0
manx <- as.matrix(simple_triplet_matrix(i = c(1,1,2,2,3,3,4,4),
                                        j = c(1:8),
                                        v = rep(1,8), ncol = 8))
many <- as.matrix(simple_triplet_matrix(i = c(1,1,1,2,2,2,3,3,3,4,4,4),
                                        j = c(1:12),
                                        v = rep(-3,12), ncol = 12))
man <- cbind(manx, many, matrix(0, ncol=12, nrow = nrow(manx)))
colnames(man) <- dv3
# rhs
man_rhs <- rep(0, nrow(man))
# dir
man_sig <- rep("<=", nrow(man))

# total fuel constraint
# sum(xij) - xi = 0
totfuel <- as.matrix(simple_triplet_matrix(i = c(rep(1,3), rep(2,2), rep(3,2),
                                                 rep(4,3), rep(5,2), rep(6,2)),
                                           j = c(1,3,27,
```

```
                                                        4,28,
                                                        5,29,
                                                        6,7,30,
                                                        8,31,
                                                        2,32),
                                                  v = c(1,1,-1,
                                                        1,-1,
                                                        1,-1,
                                                        1,1,-1,
                                                        1,-1,
                                                        1,-1), ncol = 32))
colnames(totfuel) <- dv3
# rhs
totfuel_rhs <- rep(0, nrow(totfuel))
# dir
totfuel_sig <- rep("==", nrow(totfuel))

# binary constraint
bin <- cbind(matrix(0, ncol = 20, nrow=6),
             diag(-520, nrow = 6),
             diag(1, nrow = 6))
# rhs
bin_rhs <- rep(0, nrow(bin))
# dir
bin_sig <- rep(">=", nrow(bin))

# previous month constraint
# xi - 520gammai-1 <=520
prev <- cbind(matrix(0, ncol = 20, nrow=6),
              cbind(rbind(rep(0,5),diag(520, 5)), c(520, rep(0,5))),
              diag(1, nrow = 6))
colnames(prev) <- dv3
# rhs
prev_rhs <- rep(520, nrow(prev))
# dir
prev_sig <- rep("<=", nrow(prev))
###############
# solve
coeff <- rbind(dem, man, totfuel, bin)
rhs <- c(dem_rhs, man_rhs, totfuel_rhs, bin_rhs)
```

```
dir <- c(dem_sig, man_sig, totfuel_sig, bin_sig)

# obj coeff
obj3 <- c(2,3,2,2.5,2,1,1,1.5,
          1.5,2,2.5,1.5,1,2.5,1,2,1,2,1,2,
          rep(0, 12))
names(obj3) <- dv3

res3 <- Rglpk_solve_LP(obj = obj3,
                       mat = coeff,
                       rhs = rhs,
                       dir = dir,
                       max = FALSE)
# cost
res3$optimum
sol3 <- cbind(dv3,res3$solution)[-which(res3$solution < 0.01),]
colnames(sol3) <- c("dv", "value")
xtable(sol3)
```