# Compte-rendu de TP : Programmation linéaire


WALLACE Ranveig
CATTOËN Céline
4ème année GMM
INSA

# Exercice 1 : Problème des 5 aliments

On cherche à résoudre le problème des 5 aliments, où on veut nourrir un prisonnier en minimisant le coût. On dispose de 5 aliments et on connaît leur contenu en calories et protéines. Il nous est imposé de donner 3000 calories et 100 protéines par jour au prisonnier.

Pour résoudre ce problème avec LINDO, on a d'abord posé le problème :

```
MIN     3 X1 + 10 X2 + X3 + 2 X4 + 3 X5
SUBJECT TO
       2)   25 X1 + 30 X2 + 6 X3 + X4 + 6 X5 =    30
       3)   8 X1 + 15 X2 + 2 X3 + X4 + 4 X5 =    10
END
```

On a ensuite résolu le problème avec la commande `go` et on a obtenu les résultats suivants :

```
LP OPTIMUM FOUND AT STEP      2

        OBJECTIVE FUNCTION VALUE

        1)    4.03846169

  VARIABLE        VALUE          REDUCED COST
        X1         1.153846           .000000
        X2          .000000           .480768
        X3          .000000           .192308
        X4          .000000          1.134615
        X5          .192308           .000000


       ROW   SLACK OR SURPLUS     DUAL PRICES
        2)          .000000           .230769
        3)          .000000         -1.096154

 NO. ITERATIONS=       2

 RANGES IN WHICH THE BASIS IS UNCHANGED:

                           OBJ COEFFICIENT RANGES
 VARIABLE         CURRENT        ALLOWABLE        ALLOWABLE
                   COEF          INCREASE         DECREASE
       X1        3.000000         .833332        29.499994
       X2       10.000000        INFINITY          .480768
       X3        1.000000        INFINITY          .192308
       X4        2.000000        INFINITY         1.134615
       X5        3.000000         .185185         INFINITY

                           RIGHTHAND SIDE RANGES
      ROW         CURRENT        ALLOWABLE        ALLOWABLE
                    RHS          INCREASE         DECREASE
        2       30.000000        1.250000        14.999999
        3       10.000000       10.000000          .400000

THE TABLEAU
ROW  (BASIS)       X1         X2         X3         X4       X5
1 ART            .000       .481       .192      1.135      .000    -4.038
2 X1            1.000       .577       .231      -.038      .000     1.154
3 X5             .000      2.596       .038       .327     1.000      .192
```

On voit que LINDO a introduit deux variables d'écart (une par contrainte), mais leurs coefficients sont nuls puisque les contraintes sont des égalités. Comme on l'a vu en cours, il n'est pas nécessaire d'introduire des variables d'écart quand les contraintes sont des égalités.

La solution optimale trouvée par LINDO est X1 = 1.153846 et X2 = 0.192308 ce qui donne une fonction objectif égale à 4.03846169.

Cette solution optimale est une solution de base car il y a le même nombre de variables non nulles qu'il y a de contraintes (et on a un problème standard) et les coefficients des contraintes correspondant à X1, X5 forment une matrice régulière (c'est la matrice identité $I_2$).

Dans l'analyse de sensitivité, on peut voir les augmentations et diminutions possibles sans changer de base. C'est clair que si l'on impose à X2, X3 ou X4 d'être égal à 0.1, la base ne va pas changer car on peut les augmenter jusqu'à l'infini sans changer de base. Il est donc facile de calculer l'augmentation de Z pour chacune de ces solutions :

```
X2 = 0.1  =>  Z = 4.03846169 + 0.1 * 0.481 = 4.086
X3 = 0.1  =>  Z = 4.03846169 + 0.1 * 0.192 = 4.057
X4 = 0.1  =>  Z = 4.03846169 + 0.1 * 1.135 = 4.15
```

On a également lancé le programme avec LINDO pour vérifier ces résultats et on trouve bien les valeurs ci-dessus.

On cherche maintenant à trouver une solution réalisable quand le problème consiste à maximiser la quantité de viande, c'est-à-dire X2. Les contraintes sont les mêmes que dans le problème précédent, et on rajoute de plus des contraintes du type Xi > 0 pour s'assurer que la solution optimale va bien être logique.

On donne donc le problème suivant à LINDO :

```
 MAX   X2
 SUBJECT TO
      2)   25 X1 + 30 X2 + 6 X3 + X4 + 6 X5 = 30
      3)   8 X1 + 15 X2 + 2 X3 + X4 + 4 X5 = 10
      4)   X1 >= 0
      5)   X2 >= 0
      6)   X3 >= 0
      7)   X4 >= 0
      8)   X5 >= 0
 END

       OBJECTIVE FUNCTION VALUE

       1)    .740740746E-01

 VARIABLE        VALUE          REDUCED COST
       X2          .074074           .000000
       X1         1.111111           .000000
       X3          .000000           .014815
       X4          .000000           .125926
       X5          .000000           .385185

THE TABLEAU
 ROW  (BASIS)          X2        X1        X3        X4        X5     SLK    4
   1 ART             .000      .000      .015      .126      .385      .000

 SLK    5     SLK    6     SLK    7
   .000        .000        .000
```

On voit que la solution du problème est X2 = 0.074. La solution est bien réalisable et les coefficients des Xi qui ne sont pas dans la base sont bien positifs dans la fonction objectif (voir TABLEAU) ce qui nous assure d'avoir trouvé l'optimum.

# Ecriture du problème dual :

On désire écrire le programme linéaire dual du problème des 5 aliments.
On dispose du problème primal P de la forme :

$$MIN\ cX$$

$$SUBJECT\ TO\ \begin{cases} AX = b \\ X = 0 \end{cases} \quad (P)$$

avec : C =3 10 1 2 3
A=25 30 6 1 6
8 15 2 1 4
b=30
10

D'après le théorème 7.1 du cours, le problème dual D est de la forme :

$$MAX\ Yb$$

$$SUBJECT\ TO\ \begin{cases} YA = C \\ Y \text{ de signe quelconque} \end{cases} \quad (D)$$

$(D)$ s'écrit donc :

$$MAX\ z = 30Y1 + 10Y2$$

$$SUBJECT\ TO\ \begin{cases} 25Y1 + 8Y2 \le 3 \\ 30Y1 + 15Y2 \le 10 \\ 6Y1 + 2Y2 \le 1 \\ Y1 + Y2 \le 2 \\ 6Y1 + 4Y2 \le 3 \\ Y1, Y2 \text{ de signe quelconque} \end{cases}$$

Ce problème présente 5 contraintes pour 2 variables: il y a donc des contraintes redondantes.

On résoud le problème avec LINDO:

```
MAX          30Y1+10Y2
SUBJECT TO
             25 Y1 + 8 Y2 <=3
             30 Y1 + 15 Y2 <=10
              6 Y1 +2 Y2 <=1
                Y1 + Y2 <= 2
              6 Y1 + 4 Y2 <= 3
```

On effectue les commandes suivantes :

```
free Y1
free Y2
```

pour indiquer que ces variables sont de signe quelconque, sinon elles sont positives par défaut.

```
LP OPTIMUM FOUND AT STEP      2

       OBJECTIVE FUNCTION VALUE

       1)    4.03846169

 VARIABLE        VALUE          REDUCED COST
       Y1         -.230769           .000000
       Y2         1.096154           .000000

      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)          .000000          1.153846
       3)          .480769           .000000
       4)          .192308           .000000
       5)         1.134615           .000000
       6)          .000000           .192308

NO. ITERATIONS      2
```

La fonction objectif atteint bien la même valeur que dans le problème primal : $z = 4.03846169$.

Les valeurs prises par Y1 et Y2 correspondent aux dual prices des variables d'écart du problème primal, affectées du signe inverse, car on minimise pour le premier problème et on maximise pour le second.

On retrouve de même en dual prices du problème dual les valeurs des variables du problème primal mais celles-ci ont conservent leur signe.

THE TABLEAU

| ROW (BASIS) | Y1 | Y2 | SLK 2 | SLK 3 | SLK 4 | SLK 5 | SLK 6 | |
|---|---|---|---|---|---|---|---|---|
| 1 ART | .000 | .000 | 1.154 | .000 | .000 | .000 | .192 | 4.038 |
| 2 Y1 | 1.000 | .000 | .077 | .000 | .000 | .000 | -.154 | -.231 |
| 3 SLK 3 | .000 | .000 | -.577 | 1.000 | .000 | .000 | -2.596 | .481 |
| 4 SLK 4 | .000 | .000 | -.231 | .000 | 1.000 | .000 | -.038 | .192 |
| 5 SLK 5 | .000 | .000 | .038 | .000 | .000 | 1.000 | -.327 | 1.135 |
| 6 Y2 | .000 | 1.000 | -.115 | .000 | .000 | .000 | .481 | 1.096 |

On peut constater que les coefficients qui se trouvent en colonne dans le tableau du problème primal se retrouvent en ligne dans celui du problème dual, avec leur signe inversé pour certains d'entre eux.
Les prix duaux sont non nuls pour les lignes 2 et 6 de la résolution du problème dual. Ces lignes correspondent aux variables X1 et X5, c'est à dire au pain et au lait.
On en déduit donc, grâce au théorème des écarts complémentaires que l'alimentation de moindre coût ne comporte que du pain et du lait.

# Exercice 2 : Exemple de problème duaux

### *1) <u>Problème de l'entreprise agricole :</u>*

Le problème est modélisé par le système suivant :

```
MIN     300 X + 450 Y
SUBJECT TO
    2)   0.8 X + 0.4 Y >=   40
    3)   0.05 X + 0.05 Y >=   4
    4)   0.1 X + 0.2 Y >=   10
    5)   X >=   0
    6)   Y >=   0
END
```

En résolvant le PL avec le logiciel LINDO, on obtient les résultats suivants :

```
LP OPTIMUM FOUND AT STEP      2

        OBJECTIVE FUNCTION VALUE

       27000.0000

  VARIABLE        VALUE          REDUCED COST
         X      60.000000           .000000
         Y      20.000000           .000000

      ROW   SLACK OR SURPLUS     DUAL PRICES
        2)      16.000000           .000000
        3)        .000000      -3000.000000
        4)        .000000      -1500.000000
        5)      60.000000           .000000
        6)      20.000000           .000000
 NO. ITERATIONS=       2
RANGES IN WHICH THE BASIS IS UNCHANGED:
                           OBJ COEFFICIENT RANGES
 VARIABLE         CURRENT        ALLOWABLE       ALLOWABLE
 COEF            INCREASE        DECREASE
 X              300.000000       150.000000      75.000000
 Y              450.000000       150.000000     150.000000


                          RIGHTHAND SIDE RANGES
      ROW         CURRENT        ALLOWABLE        ALLOWABLE
                    RHS          INCREASE         DECREASE
        2       40.000000        16.000000         INFINITY
        3        4.000000         1.000000          .666667
        4       10.000000         4.000000        2.000000
        5         .000000        60.000000         INFINITY
        6         .000000        20.000000         INFINITY

 THE TABLEAU
ROW(BASIS)  X      Y      SLK    2  SLK    3  SLK    4  SLK    5  SLK    6
1 ART     .00E+00 .00E+00 .00E+00   .30E+04   .15E+04   .00E+00   .00E+00  -.27E+05
2 SLK2     .000    .000   1.000      -24.000   4.000     .000      .000    16.000
3  X      1.000    .000    .000      -40.000   10.000    .000      .000    60.000
4  Y       .000   1.000    .000       20.000  -10.000    .000      .000    20.000
5 SLK5     .000    .000    .000      -40.000   10.000   1.000      .000    60.000
6 SLK6     .000    .000    .000       20.000  -10.000    .000     1.000    20.000
```

La solution optimale est 300 X + 450 Y = 27000 pour X=60 et Y=20.

## 2 ***Problème de l'entreprise chimique ;***

Le problème est modélisé par le système suivant :

```
MAX     40 U + 4 V + 10 W
SUBJECT TO
         2)   0.8 U + 0.05 V + 0.1 W <=   300
         3)   0.4 U + 0.05 V + 0.2 W <=   450
         4)   U >=   0
         5)   V >=   0
         6)   W >=   0
END
```

La résolution du problème avec LINDO donne les résultats suivants :

```
  LP OPTIMUM FOUND AT STEP      3

        OBJECTIVE FUNCTION VALUE

        1)    27000.0000

  VARIABLE        VALUE          REDUCED COST
         U         .000000         16.000000
         V     3000.000000           .000000
         W     1500.000000           .000000

      ROW   SLACK OR SURPLUS     DUAL PRICES
        2)        .000000          60.000000
        3)        .000000          20.000000
        4)        .000000            .000000
        5)   3000.000000             .000000
        6)   1500.000000             .000000

 NO. ITERATIONS=       3

 RANGES IN WHICH THE BASIS IS UNCHANGED:

                          OBJ COEFFICIENT RANGES
 VARIABLE         CURRENT        ALLOWABLE        ALLOWABLE
                   COEF          INCREASE         DECREASE
        U        40.000000        16.000000       INFINITY
        V         4.000000         1.000000        .666667
        W        10.000000         4.000000       2.000000

                          RIGHTHAND SIDE RANGES
      ROW         CURRENT        ALLOWABLE        ALLOWABLE
                    RHS          INCREASE           DECREASE
        2      300.000000       150.000000         75.000000
        3      450.000000       150.000000        150.000000
        4         .000000          .000000         INFINITY
        5         .000000      3000.000000         INFINITY
        6         .000000      1500.000000         INFINITY

 THE TABLEAU
ROW(BASIS) U     V      W    SLK  2    SLK  3    SLK  4   SLK  5   SLK  6
1 ART    16.000  .000  .000 60.000    20.000     .000     .000     .000     27000.000
2 V      24.000 1.000  .000 40.000   -20.000     .000     .000     .000      3000.000
3 W      -4.000  .000 1.000 -10.000   10.000     .000     .000     .000      1500.000
4 SLK4   -1.000  .000  .000    .000     .000    1.000     .000     .000          .000
5 SLK 5 24.000  .000  .000  40.000  -20.000     .000    1.000     .000      3000.000
6 SLK 6 -4.000  .000  .000 -10.000   10.000     .000     .000    1.000      1500.000
```

Sans les contraintes pour u > 0 v>0 w>0, on obtient les résultats suivants:

```
MAX     40 U + 4 V + 10 W
SUBJECT TO
       2)   0.8 U + 0.05 V + 0.1 W <=   300
       3)   0.4 U + 0.05 V + 0.2 W <=   450
END

: GO
LP OPTIMUM FOUND AT STEP      2

       OBJECTIVE FUNCTION VALUE

       1)    27000.0000

 VARIABLE        VALUE          REDUCED COST
        U         .000000         16.000000
        V     3000.000000           .000000
        W     1500.000000           .000000


      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)    .000000               60.000000
       3)    .000000               20.000000

NO. ITERATIONS=       2
```

La solution (v,w)=(3000,1500) est réalisable et optimale car les coefficients des variables hors bases sont tous positifs . On peut remarquer que le problème de maximisation est traité comme un problème de minimisation.
La fonction objective est de 27000.

**CONCLUSION:**
On trouve les mêmes valeurs optimales pour les fonctions objectives des 2 problèmes (c'est-à-dire 27000) ce qui est la conséquence logique de la dualité des 2 deux problèmes, ce problème est en effet un problème dual.

### *3 Analyse marginale du PRIMAL :*

On modifie désormais la troisième contrainte `4)0.1 X + 0.2 Y >= 10` en `4)0.1 X + 0.2Y>=11.`

```
MIN     300 X + 450 Y
SUBJECT TO
       2)   0.8 X + 0.4 Y >=   40
       3)   0.05 X + 0.05 Y >=   4
       4)   0.1 X + 0.2 Y >=   11
END

LP OPTIMUM FOUND AT STEP      2

       OBJECTIVE FUNCTION VALUE

       1)    28500.0000

 VARIABLE        VALUE          REDUCED COST
        X      50.000000           .000000
        Y      30.000000           .000000


      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)       12.000000           .000000
       3)         .000000      -3000.000000
       4)         .000000      -1500.000000

NO. ITERATIONS=       2
```

On obtient bien une augmentation du coût de 1500=28500-27000, valeur de w à l'optimum (w est le « prix dual » relatif à la troisième contrainte du problème de l'entreprise agricole.
On remplace la troisième contrainte par `4)  0.1 X + 0.2 Y >=   5.`

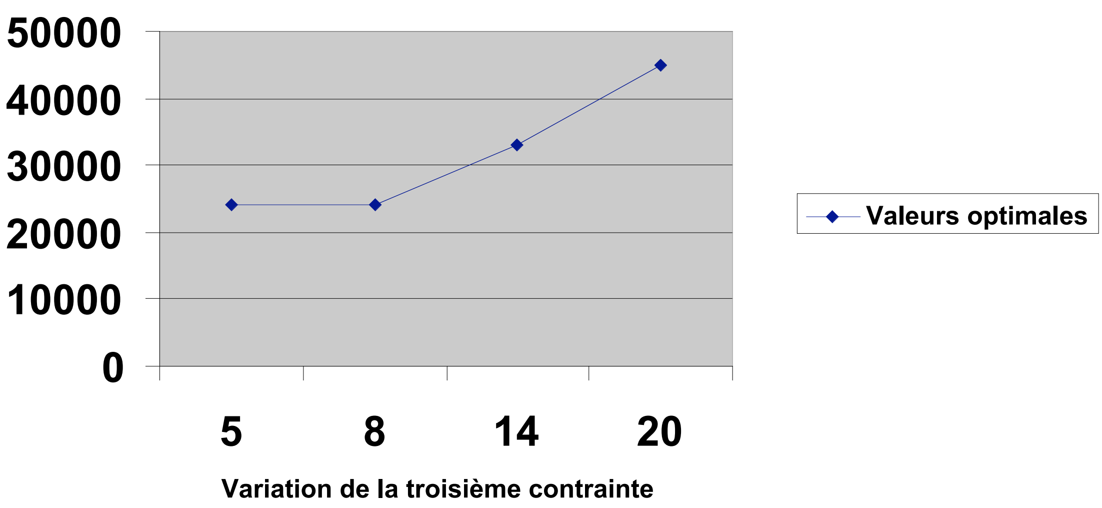
```
 MIN      300 X + 450 Y
 SUBJECT TO
        2)   0.8 X + 0.4 Y >=   40
        3)   0.05 X + 0.05 Y >=   4
        4)   0.1 X + 0.2 Y >=   5
 END

 LP OPTIMUM FOUND AT STEP      0
        OBJECTIVE FUNCTION VALUE
        1)    24000.0000

  VARIABLE        VALUE          REDUCED COST
         X                  80.000000              .000000
         Y                     .000000        150.000000



       ROW   SLACK OR SURPLUS     DUAL PRICES
        2)     24.000000              .000000
        3)       .000000          -6000.000000
        4)      3.000000              .000000

 NO. ITERATIONS=       0
```

On effectue la commande PARA 4 20 qui va permettre de représenter graphiquement les valeurs optimales de la fonction objective quand la troisième contrainte est :
0.1 X + 0.2 Y >=   b où b varie dans l'intervalle [5,20].
Les résultats sont affichés ci-dessous :

```
VAR        VAR     PIVOT     RHS        DUAL PRICE         OBJ
OUT         IN     ROW        VAL       BEFORE PIVOT       VAL
                           5.00000        .000000E+00     24000.0
SLK4        Y      4        8.00000     .000000E+00      24000.0
SLK2      SLK 3    2       14.0000      -1500.00         33000.0
X         SLK 2    3       20.0000      -2000.00         45000.0
```

Le prix dual est : (coefficient de variation de la  fo)/(delta_2nd_membre  (RHS VAL)

**Valeurs optimales de la fonction objective**



# Exercice 3 : Problème de pollution

Le but de cet exercice est de voir l'influence d'une variation des seuils sur le bénéfice maximum à l'aide des prix duaux et des coûts réduits.

Le problème est celui d'une entreprise qui veut maximiser son bénéfice où la production est contrainte par des limites de pollution. Le problème s'écrit en LINDO sous la forme suivante :

```
MAX     3 P1 + 4 P2 + P3 + 2 P4 + 6 P5
 SUBJECT TO
        2)   8 P1 + 4 P2 + P3 + P4 + 6 P5 <=   12
        3)   2 P1 + 4 P2 + 4 P5 <=   6
        4)   4 P1 + 6 P2 + 2 P3 + P4 + 8 P5 <=   16
        5)   P3 + P4 + P5 <=   4
 END
```

On va d'abord imposer à la variable P5 d'entrer dans la base, suivant le critère que l'on a vu en cours. On choisit la variable qui a le plus grand coefficient dans la fonction objectif. On le fait avec la commande `PIVOT P5` et ensuite on fait `TABL` pour déterminer la nouvelle base (`P1, P5, SLK2, SLK4, SLK5`). On laisse LINDO faire le reste de la résolution avec la commande `GO` et on voit qu'il met 4 itérations à trouver l'optimum.

On va maintenant laisser LINDO choisir les pivots, et on peut observer qu'il ne fait pas les mêmes choix que nous. A l'aide de la commande `PIVOT` on voit qu'il choisit la variable P4 et non la variable P5. Le nombre d'itérations a diminué, LINDO n'utilise plus que 3 itérations pour résoudre le problème. On ne sait pas exactement comment le logiciel fait pour choisir les pivots, mais son choix est différent du notre (critère de Dantzig) parce qu'il essaie de résoudre un problème dual au lieu de primal.

```
                              OBJ COEFFICIENT RANGES
 VARIABLE         CURRENT        ALLOWABLE        ALLOWABLE
                   COEF          INCREASE         DECREASE
       P1        3.000000         5.000000         1.000000
       P2        4.000000         2.000000          .181818
       P3        1.000000         1.000000         INFINITY
       P4        2.000000         INFINITY          .166667
       P5        6.000000          .166667         INFINITY

                         RIGHTHAND SIDE RANGES
      ROW         CURRENT        ALLOWABLE        ALLOWABLE
                    RHS          INCREASE         DECREASE
        2       12.000000        16.000000         2.000000
        3        6.000000         2.000000         4.000000
        4       16.000000         INFINITY         2.666667
        5        4.000000         2.000000         4.000000
```

On cherche maintenant à voir l'effet sur la fonction objectif de changer la valeur de certaines variables. Le bénéfice maximum est obtenu avec P3 = 0 et P5 = 0 et on va voir ce qui se passe si on impose à une de ces variables d'être non nulle. On impose d'abord P3 = 0.2. Par un calcul prévisionnel à partir du tableau ci-dessus, on voit que l'on doit trouver un nouveau bénéfice égal à `bénéfice maximum +(coefficient de P3 dans la fonction objectif)*(variation de P3) = 1*0.2 = 14.333 + 0.2.` Ce calcul marche parce que l'on ne dépasse pas l'augmentation maximale de P3 sans changer de base (allowable increase dans le tableau). On a également relancé le programme avec LINDO en rajoutant la contrainte P3=0.2 avec la commande `EXTENTION` et on trouve la même valeur de la fonction objectif que dans la prévision.

De l'analyse de sensitivité on peut voir que pour la première contrainte, on peut l'augmenter de 16 sans changer de base. On peut donc changer le deuxième membre de la contrainte de 12 à 14 sans changement de base et le nouveau bénéfice se calcule par `bénéfice maximum +(prix dual contrainte S1)*(variation du second membre de la contrainte) = 0.1667*2 = 14.3333 + 0.3334 = 14.6667.` On vérifie aisément avec qu'on obtient la même réponse avec LINDO.

```
ROW   SLACK OR SURPLUS     DUAL PRICES
 2)         .000000          .166667
```

En faisant le même changement à la contrainte S2, on ne dépasse toujours pas l'augmentation autorisée sans changer de base et donc le calcul du nouveau bénéfice se fait de la même façon, on trouve BM = 15.999666. Avec LINDO on trouve le même bénéfice, mais il y a quand même eu un changement de base. C'est parce que l'on est sur la limite acceptable (« allowable increase »), donc il y a deux bases possibles et LINDO a choisit l'autre.

# Exercice 4 : Utilisation d'un fichier de commande.

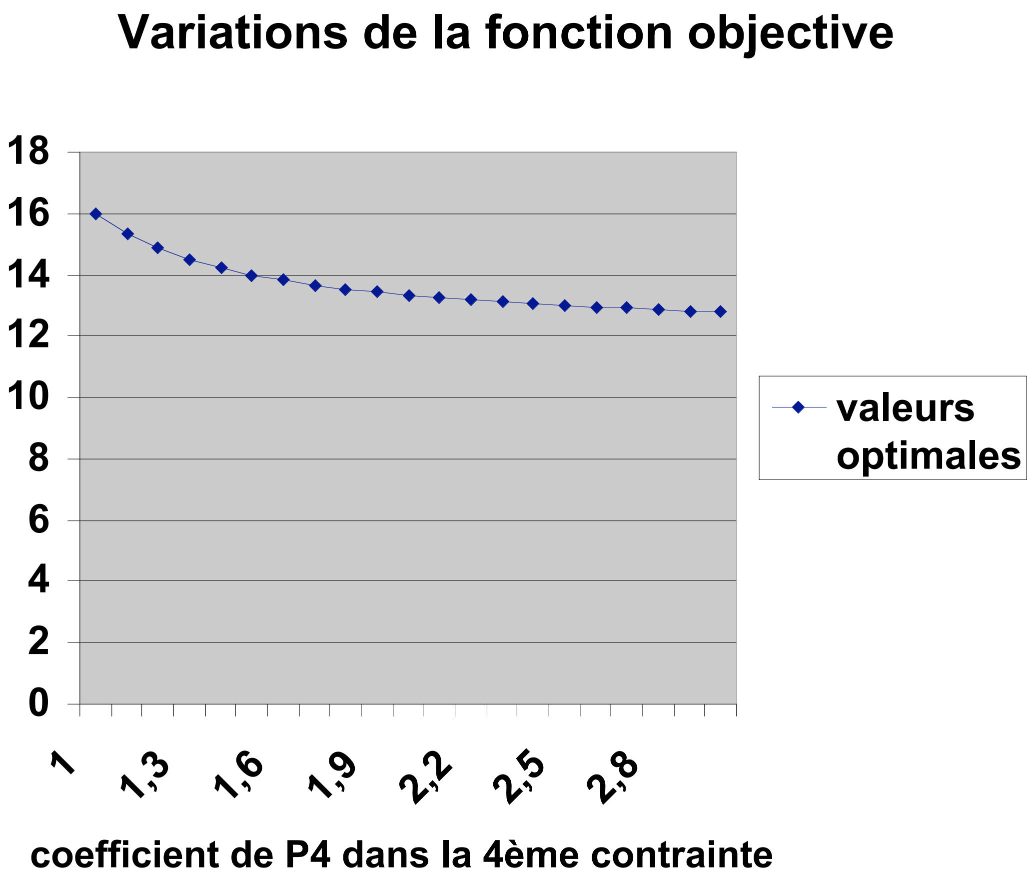
Dans le problème de pollution de l'usine pétrochimique, on veut savoir comment varie la fonction objectif si les valeurs prises par le coefficient 1 (représentant la quantité de substance S4 rejetée par P4) est en augmentation. Pour observer ces variations on écrit un fichier de commande de la forme :

```
Exo4.txt :
BATCHFILE
DIVERSION resultat.txt
RETROUVE pollutio.bin
LOOK ALL
TERSE
ALTER 5
P4
1
GO
……
GO
RVRT
LEAVE
```

En tapant la commande : `Take 'exo4.txt'` on obtient les résultats sur le fichier suivant :

```
Resultat.txt :

 MAX     3 P1 + 4 P2 + P3 + 2 P4 + 6 P5
 SUBJECT TO
        2)   8 P1 + 4 P2 + P3 + P4 + 6 P5 <=   12
        3)   2 P1 + 4 P2 + 4 P5 <=   8
        4)   4 P1 + 6 P2 + 2 P3 + P4 + 8 P5 <=   16
        5)   P3 + P4 + P5 <=   4
 END


 LP OPTIMUM FOUND AT STEP      3
 OBJECTIVE VALUE =   16.0000000
 …………
 LP OPTIMUM FOUND AT STEP      0
 OBJECTIVE VALUE =   12.8333330
 LP OPTIMUM FOUND AT STEP      0
 OBJECTIVE VALUE =   12.8000002
```

On représente graphiquement les variations de la valeur optimale de la fonction objective :




<u>Interprétation des résultats :</u>
L'augmentation du coefficient de la variable `P4` dans la 4$^{ème}$ contrainte `P3 + P4 + P5 <=    4` entraîne une diminution nette du coût de la fonction objectif, ce qui signifie que l'augmentation de la quantité de produit `P4` pour une même quantité de substance polluante `S4` n'améliore pas le bénéfice de l'entreprise.

# Exercice 5 : PL où le bénéfice à maximiser est « linéaire par morceaux »

On cherche toujours à résoudre le problème de pollution de l'usine pétrochimique en supposant qu'il y a saturation de marché, c'est-à-dire que le prix de vente sera plus petit si la production est augmentée. On le modélise de la façon suivante : le bénéfice relatif au produit P4 est de

4 par centaine d'unités si P4 < 1 ;
3 par centaine d'unités si 1 < P4 < 3 ;
2 par centaine d'unités si P4 > 3.

On remplace ensuite P4 par A + B + C dans les contraintes et par 4A + 3B + 2C dans la fonction objectif, où chacune des variables correspond à l'une des tranches de centaines d'unités considérés. On rajoute également les contraintes A < 1 et A + B <3. Puisque l'on gagne le plus en remplissant A d'abord au lieu de B ou C et que ensuite on gagne plus en remplissant B avant C, on va effectivement avoir une modélisation du problème que l'on considère ici (dans les contraintes A, B, C jouent le meme role :P4 = A + B + C).

On va maintenant imposer une nouvelle contrainte de la forme A + B + C < m et on va faire varier m de 0.5 à 4.5 par pas de 0.5. On constate pour ces valeurs que l'on a toujours l'une des trois situations suivantes :

(s1) A < 1, B =0, C = 0
(s2) A = 1, B < 2, C = 0
(s3) A = 1, B = 2, C > 0

ce qui montre que l'on a bien modélisé le problème, car logiquement, on ne peut remplir B avant d'avoir atteint A = 1 et on ne peut pas remplir C avant d'avoir saturé la contrainte sur B.

On cherche ensuite à modéliser le même problème, mais avec un marché qui augmente avec la production, donc on va maintenant remplacer P4 par 4C + 3B + 2A ce qui veut dire que c'est la variable C qui est la plus intéressante. La modélisation du problème précédent n'est plus valable, car il remplira d'abord C, ensuite B et finalement A, alors que le modèle impose un remplissage de A d'abord, ensuite B et finalement C. On sépare alors le problème en trois sous problèmes où l'on impose des contraintes sur deux des variables A, B et C. Par exemple, pour le problème (s1) (c'est-à-dire que m < 1), on impose B = 0 et C = 0. Dans le cas 1 < m < 3 on doit imposer A = 1 et C = 0. Pour m > 3, on doit avoir A = 1 et B = 2. Avec cette modélisation on retrouve bien un modèle du problème physique que l'on considère.

# Exercice 6 : Problème d'affectation de machines.

On affecte 4 machines A,B,C,D aux tâches T1,T2,T3,T4 et on souhaite minimiser la somme des coûts. On modélise le problème en PL sachant qu l'on aune seule machine par tâche et une seule tâche par machine.

```
MIN     8 A1 + 3 A2 + A3 + 5 A4 + 11 B1 + 7 B2 + B3 + 6 B4 + 7 C1 + 8 C2 + 6 C3 + 8
C4                                                            + 11 D1 + 6 D2 + 4 D3+
9 D4
 SUBJECT TO
        2)   A1 + A2 + A3 + A4 =    1
        3)   B1 + B2 + B3 + B4 =    1
        4)   C1 + C2 + C3 + C4 =    1
        5)   D1 + D2 + D3 + D4 =    1
        6)   A1 + B1 + C1 + D1 =    1
        7)   A2 + B2 + C2 + D2 =    1
        8)   A3 + B3 + C3 + D3 =    1
        9)   A4 + B4 + C4 + D4 =    1
 END
INTE 16
```

On utilise un fichier de commandes batchfile pour résoudre ce problème où les 12 variables sont astreintes à être bivalentes (spécification : `INT 16` après `END`) c'est-à-dire valent 0 ou 1. Les résultats obtenus sont :

```
OBJECTIVE FUNCTION VALUE
        1)    19.0000000
```

et les variables non nulles pour la valeur optimale de la fonction sont : `A4,B3,C1,D2` .

Si l'on supprime la restriction qui impose aux 16 variables d'être bivalentes (`INTE 0`), on obtient la même valeur optimale de la fonction objectif et les même variables non nulles que la résolution précédente. Cependant les valeurs des « reduced costs » sont différentes si l'on supprime la restriction.

Cette équivalence des solutions s'explique par le fait que chaque sous matrice des contraintes a pour déterminant + ou - 1: en effet, si on a le problème RX=B on trouve X=1/(detR)[ ...]. Les valeurs entières dans la partie [...] sont divisées par detR=1 ou -1, ainsi on obtient bien des valeurs entières pour les variables du problème.

Si l'on supprime l'une quelconque des contraintes la solution est encore identique à celle de la question précédente sauf les « reduced costs » qui sont modifiés.
Cette particularité s'explique par le fait qu'une contrainte peut s'écrire comme une combinaison linéaire des autres contraintes. Ainsi, même si une contrainte disparaît elle est toujours active par le biais des autres contraintes.

On observe les coefficients de la matrice des contraintes: il n'y a que des 1,-1, ou 0.

# Exercice 7 : Analyse paramétrique d'un problème de production

On considère un problème de production où l'on a trois machines qui peuvent fabriquer des pièces d'une certaine épaisseur à une certaine vitesse avec un certain coût. Les trois machines ne peuvent travailler que 35 heures par semaine. Sur une semaine on doit produire au moins 21800 m de barres de 0.5 cm d'épaisseur (d'où la contrainte X3A >= 218 car seulement la première machine peut produire ces pièces), 11400 barres de 1 cm d'épaisseur (d'où la contrainte X3B + X4B + X5B >= 114 car les trois machines peuvent produire ces pièces) et 11100 barres de 1.5 cm d'épaisseur (d'où la contrainte X4C + X5C >= 111 car seulement les deux dernières machines peuvent produire ces pièces). Les trois contraintes

```
0.1111 X3A + 0.1111 X3B <=   35
0.1667 X4B + 0.1667 X4C <=   35
0.2222 X5B + 0.2222 X5C <=   35
```

sont dues aux limitation en temps des machines (35 heures par semaine par machine) et la dernière contrainte (X3A + X3B + X4B + X4C + X5B + X5C <=  600) vient de la capacité de stockage par semaine.

```
:go
LP OPTIMUM FOUND AT STEP      9

       OBJECTIVE FUNCTION VALUE

       1)     10073.8799

 VARIABLE        VALUE          REDUCED COST
      X3A       218.000000          .000000

      X3B        97.031494          .000000
      X4B          .000000          .000000
      X4C       209.958008          .000000
      X5B        16.968506          .000000
      X5C        58.042000          .000000


      ROW   SLACK OR SURPLUS     DUAL PRICES
       2)          .000000        -2.999999
       3)          .000000         -.999999
       4)       157.000000          .000000
       5)          .000000        24.032404
       6)          .000000         7.678468
       7)        18.332666          .000000
       8)          .000000        16.219999

NO. ITERATIONS=       9
```

Les variables qui sont dans la base sont celles qui sont non nulles dans la première colonne. On voit qu'ils ont aussi la deuxième colonne nulle. On peut rentrer X4B dans la base car le coût réduit est nul donc cela ne changera pas la valeur de la fonction objectif. On va constater ce fait en faisant un petit changement dans la fonction objectif : on met le coefficient de X4B égal à 16.50001. Maintenant, LINDO va préférer choisir X4B dans la base, puisqu'on y gagnerait un peu plus. En effet, en lançant ce programme, on voit que X4B prend la place de X5B dans la base et la fonction objectif reste inchangée (sauf un petit rajout à cause de la modification de la fonction objectif). Cette technique permet de forcer LINDO a chercher la deuxième solution possible du problème, car quand il y en a plusieurs, LINDO n'en affiche qu'une.

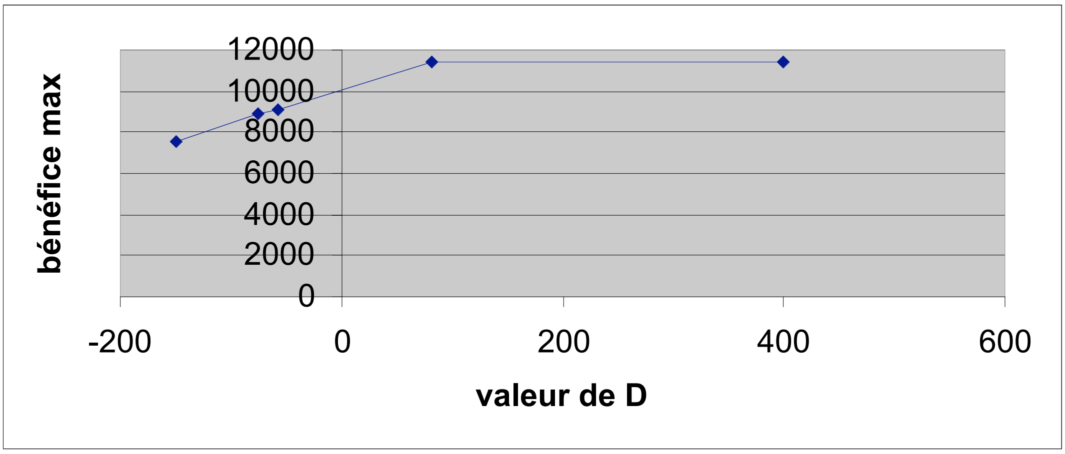
```
                RANGES IN WHICH THE BASIS IS UNCHANGED:

                                     OBJ COEFFICIENT RANGES
 VARIABLE          CURRENT          ALLOWABLE         ALLOWABLE
                    COEF            INCREASE          DECREASE
      X3A        15.890000          1.999999          INFINITY
      X3B        17.889999          INFINITY          1.670000
      X4B        16.500010           .999990          INFINITY
      X4C        17.500000          INFINITY           .999990
      X5B        15.220000           .999999          INFINITY
      X5C        16.219999          1.280001           .999999

                                     RIGHTHAND SIDE RANGES
      ROW          CURRENT          ALLOWABLE         ALLOWABLE
                     RHS            INCREASE          DECREASE
        2       218.000000          1.033294        218.000000
        3       114.000000          1.033294          INFINITY
        4       111.000000        155.966705          INFINITY
        5        37.000000          6.333667           .114799
        6        35.000000          9.503351         16.754524
        7        35.000000          INFINITY         22.332666
        8       600.000000        100.507050         57.008705


 : para 6 50

    VAR        VAR    PIVOT    RHS       DUAL PRICE       OBJ
    OUT        IN      ROW     VAL      BEFORE PIVOT      VAL

                             35.0000       7.67847       10073.9
      X5C  SLK    6      8   47.5042       7.67847       10169.9
                             50.0000       .000000E+00   10169.9
```

On a fait l'analyse post-optimale pour voir si l'on peut augmenter B3 ou B4 sans changer de base (et le nouveau bénéfice est alors facile à calculer car il est linéaire par rapport au changement). Le prix dual de B3 est inférieur à 2 pour une augmentation, donc en augmentant la contrainte de 2 on observe un changement de base. Pour B4 le prix dual est supérieur à 1 donc il n'y a pas de changement de base.

On suppose ensuite que la limitation de la capacité de stockage est de la forme 60000 + 100D, où D est un paramètre. On utilise la commande PARA pour faire varier la valeur de D de –150 à 400.

```
 : para 8 1000

    VAR        VAR    PIVOT    RHS       DUAL PRICE       OBJ
    OUT        IN      ROW     VAL      BEFORE PIVOT      VAL

                             450.000       17.5000       7544.89
 SLK    6        X5B     6   524.990       17.5000       8857.21
      X4B        X5C     3   541.958       16.2200       9132.44
 SLK    7  SLK    8      7   682.505       16.2200       11412.1
                             1000.00       .000000E+00   11412.1
```

On trace ensuite le graphique à partir des résultats de LINDO :

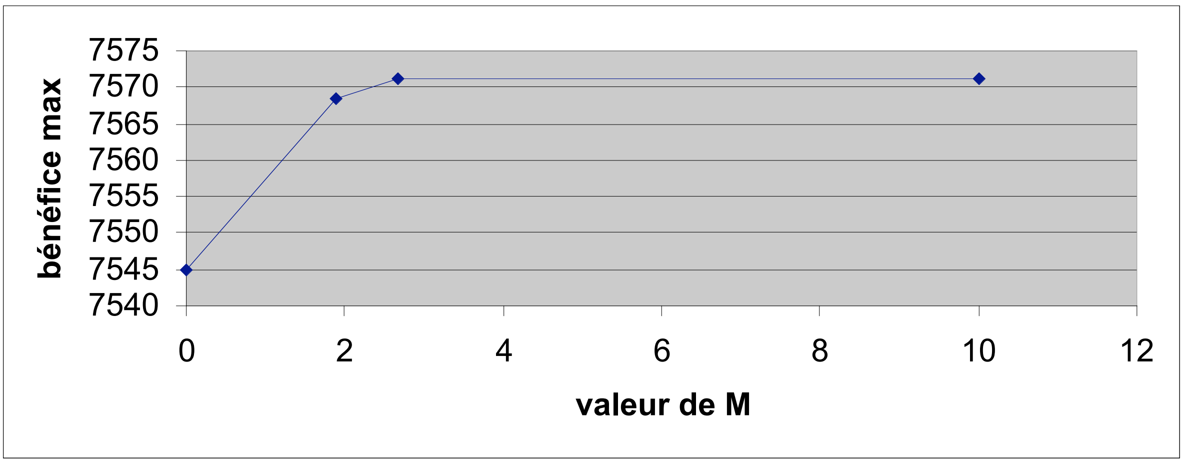



INTERPRETATION:

On voit qu'à partir d'une certaine valeur de D (ici 82.505), le bénéfice est constant, donc à partir de ce moment on ne peut pas gagner plus en rajoutant de la capacité de stockage.

On suppose maintenant que le nombre d'heures disponibles par semaine est susceptible de passer de 35 à 35 + M pour B3, à 35 + 2M pour B4 et à 35 + 3M pour B5. On a fait l'étude de la variation du profit maximal en fonction de M pour 0 < M < 10 :

```
: para 9 10

        VAR        VAR      PIVOT     RHS         DUAL PRICE        OBJ
        OUT        IN       ROW       VAL        BEFORE PIVOT       VAL

                                   .000000E+00    12.5112         7544.89
          X4B   SLK     3       3   1.88520       12.5112         7568.48
  SLK     4   SLK     5       4   2.66290       3.51035         7571.21
                                   10.0000       .000000E+00     7571.21
```




Encore une fois, le bénéfice maximum est constant à partir d'une certaine valeur de M (ici 2.6629) donc ce n'est pas la peine d'augmenter d'avantage le nombre d'heures sur chaque machine, de toute façon ceci n'augmentera pas le bénéfice maximum.

# Exercice 8 : Problème de circuit hamiltonien

On modélise un problème de trajet minimal en temps.
La modélisation est incomplète car elle n'impose pas de passer par tous les arrêts avant de revenir au point de départ. Si la partie X des points d'arrêts accessibles à partir de 1 ne recouvre pas le circuit {1 ,2 ,3 ,4 ,5 ,6} on impose alors à X d'être « non stable » c'est-à-dire :

$$\sum_{i\in X, j\in \overline{X}} x_{ij} \geq 1 \quad \exists i \in X,\ j \in \overline{X} : x_{ij} = 1$$

Il faut rajouter au moins $\frac{2^n - n}{2}$ contraintes de non stabilité pour que les points d'arrêts recouvrent l'ensemble {1,2 ,3 ,4 ,5 ,6}. En pratique dans la résolution on modifie une partie des contraintes « à la demande » et le nombre de contraintes à rajouter est très inférieur à : $\frac{2^n - n}{2}$.
On est ici dans le cas d'une analyse polyédrale, en effet on se limite à une partie des contraintes, utile dans le voisinage du point optimal.

Résolution du PL:

```
MIN
26X12+43X13+16X14+30X15+26X16+7X21+16X23+X24+30X25+25X26+20X31+13X32+35X34+5X35+21X
41+16X42+25X43+18X45+18X46+12X51+46X52+27X53+28X54+5X56+23X61+5X62+5X63+9X64+5X65+1
0X36
SUBJECT TO
X12+X13+X14+X15+X16 =1
X21+X23+X24+X25+X26 =1
X31+X32+X34+X35+X36 =1
X41+X42+X43+X45+X46 =1
X51+X52+X53+X54+X56 =1
X61+X62+X63+X64+X65 =1
X21+X31+X41+X51+X61 =1
X12+X32+X42+X52+X62 =1
X13+X23+X43+X53+X63 =1
X14+X24+X34+X54+X64 =1
X15+X25+X35+X45+X65 =1
X16+X26+X36+X46+X56 =1
end
int 30
```

On obtient le résultat suivant :

```
        OBJECTIVE FUNCTION VALUE
        1)    54.0000000
  VARIABLE        VALUE          REDUCED COST
       X12                .000000         26.000000
       X13                .000000         43.000000
       X14              1.000000        16.000000
       X15                .000000         30.000000
       X16                .000000         26.000000
       X21              1.000000         7.000000
       X23                .000000         16.000000
       X24                .000000          1.000000
       X25                .000000         30.000000
```

```
      X26                .000000        25.000000
      X31                .000000        20.000000
      X32                .000000        13.000000
      X34                .000000        35.000000
      X35               1.000000         5.000000
      X41                .000000        21.000000
      X42               1.000000        16.000000
      X43                .000000        25.000000
      X45                .000000        18.000000
      X46                .000000        18.000000
      X51                .000000        12.000000
      X52                .000000        46.000000
      X53                .000000        27.000000
      X54                .000000        28.000000
      X56               1.000000         5.000000
      X61                .000000        23.000000
      X62                .000000         5.000000
      X63               1.000000         5.000000
      X64                .000000         9.000000
      X65                .000000         5.000000
      X36                .000000          .000000
```

La valeur minimale du trajet effectué par l'autocar de ramassage scolaire est donc de 54 min.

Si à présent on impose à la variable X d'être non stable, on résoud alors le problème suivant :

```
min  26X12+43X13+16X14+30X15+26X16+7X21+16X23+X24+30X25+25X26
+20X31+13X32+35X34+5X35+21X41+16X42+25X43+18X45+18X46
+12X51+46X52+27X53+28X54+5X56+23X61+5X62+5X63+9X64+5X65+0X36
SUBJECT  TO
X12+X13+X14+X15+X16 =1
X21+X23+X24+X25+X26 =1
X31+X32+X34+X35+X36 =1
X41+X42+X43+X45+X46 =1
X51+X52+X53+X54+X56 =1
X61+X62+X63+X64+X65 =1
X21+X31+X41+X51+X61 =1
X12+X32+X42+X52+X62 =1
X13+X23+X43+X53+X63 =1
X14+X24+X34+X54+X64 =1
X15+X25+X35+X45+X65 =1
X16+X26+X36+X46+X56 =1
X13+X15+X16+X23+X25+X26+X43+X45+X46>1
END
INT 30
```

On obtient le résultat suivant:

```
LP OPTIMUM FOUND AT STEP     22
 OBJECTIVE VALUE    60.5000000
 FIX ALL VARS (    18)  WITH RC >   1.50000
 SET       X45 TO >=   1 AT    1, BND= -67.00     TWIN=-.1000E+31
 DELETE       X45 AT LEVEL     1
 ENUMERATION COMPLETE. BRANCHES=    1 PIVOTS    24

 LAST INTEGER SOLUTION IS THE BEST FOUND
 RE-INSTALLING BEST SOLUTION...

        OBJECTIVE FUNCTION VALUE
1)62.0000000
  VARIABLE        VALUE          REDUCED COST
      X12         1.000000         26.000000
      X13          .000000         43.000000
      X14          .000000         16.000000
      X15          .000000         30.000000
      X16          .000000         26.000000
      X21          .000000          7.000000
```

```
X23          .000000         16.000000
X24         1.000000          1.000000
X25          .000000         30.000000
X26          .000000         25.000000
X31          .000000         20.000000
X32          .000000         13.000000
X34          .000000         35.000000
X35          .000000          5.000000
X41          .000000         21.000000
X42          .000000         16.000000
X43          .000000         25.000000
X45         1.000000         18.000000
X46          .000000         18.000000
X51         1.000000         12.000000
X52          .000000         46.000000
X53          .000000         27.000000
X54          .000000         28.000000
X56          .000000          5.000000
X61          .000000         23.000000
X62          .000000          5.000000
X63         1.000000          5.000000
X64          .000000          9.000000
X65          .000000          5.000000
X36         1.000000           .000000
```

La solution obtenue n'est pas réalisable, 'après le tableau on a un coefficient de 1 entre X36 et X63. Il y a donc un aller-retour infini entre l'arrêt 3 et l'arrêt 6.

Si l'on rajoute une contrainte, on obtient les résultats suivant :

```
MIN
26X12+43X13+16X14+30X15+26X16+7X21+16X23+X24+30X25+25X26+20X31+13X32+35X34+5X35+21X
41+16X42+25X43+18X45+18X46+12X51+46X52+27X53+28X54+
5X56+23X61+5X62+5X63+9X64+5X65+0X36
SUBJECT TO
X12+X13+X14+X15+X16 = 1
X21+X23+X24+X25+X26 = 1
X31+X32+X34+X35+X36 = 1
X41+X42+X43+X45+X46 = 1
X51+X52+X53+X54+X56 = 1
X61+X62+X63+X64+X65 = 1
X21+X31+X41+X51+X61 = 1
X12+X32+X42+X52+X62 = 1
X13+X23+X43+X53+X63 = 1
X14+X24+X34+X54+X64 = 1
X15+X25+X35+X45+X65 = 1
X16+X26+X36+X46+X56 = 1
X13+X15+X16+X23+X25+X26+X43+X45+X46>1
X13+X16+X23+X26+X43+X46+X53+X56>1
END
INT 30
```

Résultats:

```
        OBJECTIVE FUNCTION VALUE
        1)    63.0000000   =20
  VARIABLE        VALUE          REDUCED COST
       X12               .000000        26.000000
       X13               .000000        43.000000
       X14              1.000000        16.000000
       X15                .000000        30.000000
       X16                .000000        26.000000
       X21              1.000000         7.000000
       X23                .000000        16.000000
       X24                .000000         1.000000
       X25                .000000        30.000000
       X26                .000000        25.000000
```

| | | |
|---|---|---|
| X31 | .000000 | 20.000000 |
| X32 | .000000 | 13.000000 |
| X34 | .000000 | 35.000000 |
| X35 | 1.000000 | 5.000000 |
| X41 | .000000 | 21.000000 |
| X42 | .000000 | 16.000000 |
| X43 | 1.000000 | 25.000000 |
| X45 | .000000 | 18.000000 |
| X46 | .000000 | 18.000000 |
| X51 | .000000 | 12.000000 |
| X52 | .000000 | 46.000000 |
| X53 | .000000 | 27.000000 |
| X54 | .000000 | 28.000000 |
| X56 | 1.000000 | 5.000000 |
| X61 | .000000 | 23.000000 |
| X62 | 1.000000 | 5.000000 |
| X63 | .000000 | 5.000000 |
| X64 | .000000 | 9.000000 |
| X65 | .000000 | 5.000000 |
| X36 | .000000 | .000000 |

La solution obtenue est optimale car on obtient bien une tournée. Le car met donc 63 min pour faire la tournée et passer par tous les arrêts.
Si on avait voulu écrire toutes les contraintes de non stabilité, il aurait fallu en rajouter 25.