



IUP CACHAN
40, avenue du Général Leclerc
94 300 CACHAN

Etude d'optimisation selon la méthode du simplexe

Formation Ingénieur-Maître
en Génie électrique et Informatique Industrielle
Promotion 2003

SOMMAIRE

<i>1/ Bref historique.....</i>	<i>3</i>
<i>2/ Introduction.....</i>	<i>3</i>
<i>3/ Propri�t� du simplexe</i>	<i>4</i>
<i>4/ Algorithme du simplexe.....</i>	<i>4</i>
<i>5/ M�thode du simplexe - d�finitions.....</i>	<i>5</i>
<i>6/ Forme canonique</i>	<i>5</i>
<i>7/ Forme normalis�e</i>	<i>6</i>
<i>8/ Exemple.....</i>	<i>6</i>
<i>A/ Forme canonique.....</i>	<i>6</i>
<i>B/ Forme normalis�e.....</i>	<i>7</i>
<i>C/ R�solution selon la m�thode simplexe.....</i>	<i>7</i>
<i>D/ R�solution selon la m�thode simplexe avec Matlab.....</i>	<i>9</i>
<i>1/ 1�re solution</i>	<i>9</i>
<i>2/ 2�re solution</i>	<i>14</i>
<i>9/ Cas rencontr� dans l'industrie dans le cadre de mes fonctions de Responsable Maintenance</i>	<i>16</i>
<i>10/ Annexe.....</i>	<i>28</i>
<i>commande linprog sous Matlab.....</i>	<i>30</i>

1/ BREF HISTORIQUE

- 1907. de la Vallée Poussin. Solution de $kAx - bk1$.
- 1930s. Kantorovich (Union soviétique). Applications à l'économie. Planifications de production.
- 1940s. Dantzig. Problèmes logistiques militaires durant la deuxième guerre mondiale. (1947: Algorithme du simplexe.)
- 1950s–60s. Identification d'applications dans beaucoup d'autres domaines. Optimisation de structures, théorie du contrôle, conception de filtres, etc.
- 1979. Algorithme de l'ellipsoïde (Khachiyan). Plus efficace que l'algorithme du simplexe dans le pire des cas, mais moins rapide en pratique.
- 1984. Algorithme de point intérieur (Karmarkar). Complexité polynomiale. Plus efficace en pratique.
- 1984–aujourd'hui. Beaucoup de variations des méthodes de point intérieur. Logiciels spécialistes pour des problèmes de grande taille.

2/ INTRODUCTION

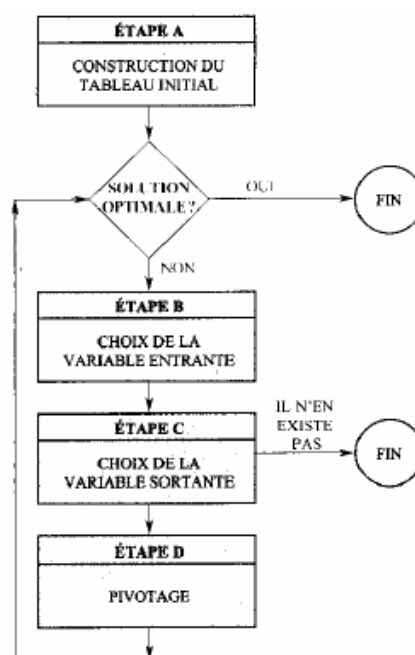
- Seule méthode exacte pour solutionner des problèmes linéaires de grande taille
- Méthode itérative algébrique où l'on circule séquentiellement sur les sommets à l'intérieur de la zone de solution jusqu'à l'obtention de la solution optimale

3/ PROPRIETE DU SIMPLEXE

- Zone de solution du problème linéaire toujours convexe.
 - une surface est convexe si elle est située toute entière du même côté d'un plan tangent.
- S'il existe une seule solution optimale au problème linéaire, elle est obligatoirement localisée sur un sommet de la zone de solution.
- S'il existe de multiples solutions optimales, au moins deux d'entre elles doivent être localisées sur des sommets adjacents.
- Le nombre de sommets de la zone de solution est fini.
- Si la solution réalisable localisée à un sommet donné n'a pas de voisin adjacent dont la solution est supérieure, ce sommet est la solution optimale.

4/ ALGORITHME DU SIMPLEXE

1. Déterminer une solution de base réalisable.
2. Vérifier si la solution actuelle est optimale.
3. Déterminer la variable hors base qui va devenir variable de base.
3. Déterminer la variable de base qui sortira de la solution.
4. Effectuer les opérations linéaires (pivots) selon la technique de Gauss-Jordan.



5/ MÉTHODE DU SIMPLEXE – DÉFINITIONS

Systèmes d'équations équivalents

- Systèmes qui possèdent le même ensemble de solutions

Variable de base

- Variable qui a un coefficient unitaire positif dans une des équations du système et un coefficient nul partout ailleurs

Opérations pivot

- Opération de Gauss-Jordan pour transformer un système d'équations équivalent dans lequel une variable devient de base

Système canonique

- Système d'équations où il y a une variable de base par équation

Solution de base

- Système d'équations où les variables hors base sont fixées à zéro résolu pour les variables de base

6/ FORME CANONIQUE

a/ Problème de maximisation

$$\begin{aligned} \text{Max} \quad & \sum_{j=1}^n c_j x_j \\ \text{sujet à} \quad & \sum_{j=1}^n a_{ij} x_j \leq b_i \quad i = 1, \dots, m \\ & x_j \geq 0 \quad j = 1, \dots, n \end{aligned}$$

b/ Problème de minimisation

$$\begin{aligned} \text{Min} \quad & \sum_{j=1}^n c_j x_j \\ \text{sujet à} \quad & \sum_{j=1}^n a_{ij} x_j \geq b_i \quad i = 1, \dots, m \\ & x_j \geq 0 \quad j = 1, \dots, n \end{aligned}$$

7/ FORME NORMALISEE

a/ Problème de maximisation

$$\begin{aligned} \text{Max} \quad & \sum_{j=1}^n c_j x_j \\ \text{sujet à} \quad & \sum_{j=1}^n a_{ij} x_j = b_i \quad i = 1, \dots, m \\ & x_j \geq 0 \quad j = 1, \dots, n \end{aligned}$$

b/ Problème de minimisation

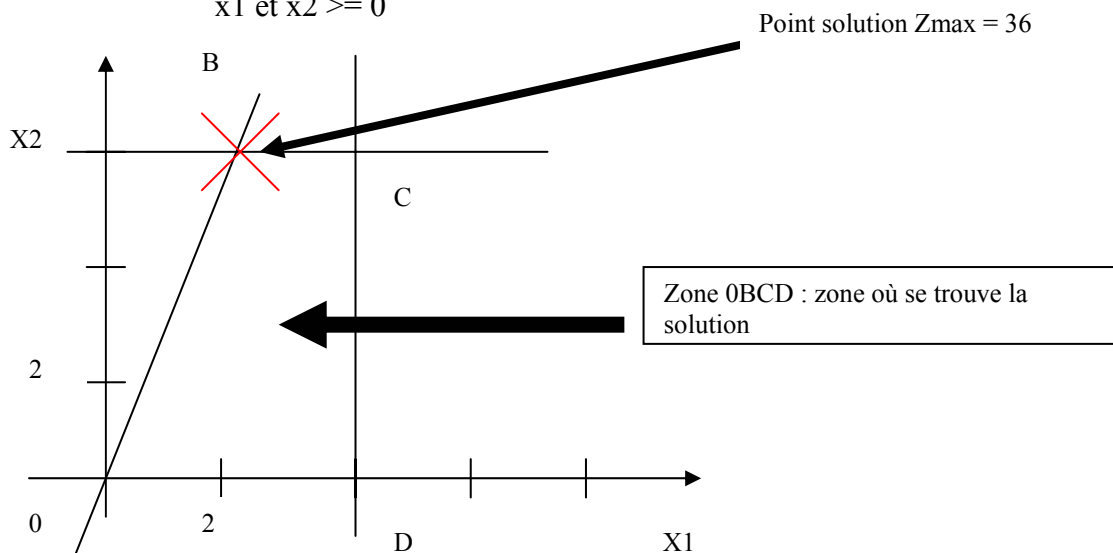
$$\begin{aligned} \text{Min} \quad & \sum_{j=1}^n c_j x_j \\ \text{sujet à} \quad & \sum_{j=1}^n a_{ij} x_j = b_i \quad i = 1, \dots, m \\ & x_j \geq 0 \quad j = 1, \dots, n \end{aligned}$$

8/ EXEMPLE

Maximiser $Z = 3x_1 + 5x_2$

A/ Forme canonique

Contraintes : $x_1 \leq 4$
 $2x_2 \leq 12$
 $3x_1 + 2x_2 \leq 18$
 $x_1 \text{ et } x_2 \geq 0$



B/ Forme normalisée

$$Z \quad - 3 x_1 - 5 x_2 = 0 \quad \text{équation (0)}$$

$$x_1 = 4 \quad \text{équation (1)}$$

$$2 x_2 = 12 \quad \text{équation (2)}$$

$$3 x_1 + 2 x_2 = 18 \quad \text{équation (3)}$$

avec $x_1, x_2, x_3, x_4, x_5 \geq 0$

C/ Résolution selon la méthode du simplexe

1/ Initialisation avec intégration d'une variable de base dans chacune des équations

$$Z \quad - 3 x_1 - 5 x_2 = 0 \quad (0)$$

$$x_1 + x_3 = 4 \quad (1)$$

$$2 x_2 + x_4 = 12 \quad (2)$$

$$3 x_1 + 2 x_2 + x_5 = 18 \quad (3)$$

avec $x_1, x_2, x_3, x_4, x_5 \geq 0$

Z , x_3 , x_4 et x_5 sont les variables de base.
 x_1 et x_2 sont les variables hors base.

Solution initiale

$$\begin{aligned} x_1, x_2 &= 0 \\ x_3 &= 4 \\ x_4 &= 4 \\ x_5 &= 4 \\ Z &= 0 \end{aligned}$$

2/ Opération pivot pour faire rentrer x_2 dans la base

Equation (0) = Equation (0) + 5/2 Equation (2)

Equation (3) = Equation (3) – Equation (2)

Soit un système d'équation :

$$\mathbf{Z} \quad -3x_1 \quad +5/2 x_4 \quad = 30 \quad (0)$$

$$x_1 \quad + x_3 \quad = 4 \quad (1)$$

$$2x_2 \quad + x_4 \quad = 12 \quad (2) \quad \text{ou} \quad x_2 \quad + 1/2 x_4 \quad = 6$$

$$3x_1 \quad -x_4 \quad + x_5 \quad = 6 \quad (3)$$

avec $x_1, x_2, x_3, x_4, x_5 \geq 0$

$\mathbf{Z}$, x_3 , x_2 et x_5 sont les variables de base.
 x_1 et x_4 sont les variables hors base.

3/ Opération pivot pour faire rentrer x_1 dans la base

Soit le système d'équation :

$$\mathbf{Z} \quad -3x_1 \quad +5/2 x_4 \quad = 30 \quad (0)$$

$$x_1 \quad + x_3 \quad = 4 \quad (1)$$

$$x_2 \quad + 1/2 x_4 \quad = 6 \quad (2)$$

$$3x_1 \quad -x_4 \quad + x_5 \quad = 6 \quad (3)$$

avec $x_1, x_2, x_3, x_4, x_5 \geq 0$

$\mathbf{Z}$, x_3 , x_2 et x_5 sont les variables de base.
 x_1 et x_4 sont les variables hors base.

$$\text{Equation (3)} = 1/3 \text{Equation (3)}$$

$$\text{Equation (1)} = \text{Equation (1)} - \text{Equation (3)}$$

$$\text{Equation (0)} = \text{Equation (0)} + 3 \text{Equation (3)}$$

$$\mathbf{Z} \quad +3/2 x_4 \quad +x_5 \quad = 36 \quad (0)$$

$$+ x_3 \quad + 1/3 x_4 \quad -1/3 x_5 \quad = 2 \quad (1)$$

$$x_2 \quad + 1/2 x_4 \quad = 6 \quad (2)$$

$$x_1 \quad -1/3 x_4 \quad + 1/3 x_5 \quad = 2 \quad (3)$$

avec $x_1, x_2, x_3, x_4, x_5 \geq 0$

$\mathbf{Z}$, x_3 , x_2 et x_1 sont les variables de base.

X5 et x4 sont les variables hors base.

Les variables x1 et x2 étant des variables de base, la solution optimale est $Z=36$ avec:

$$x1 = 2, x2 = 6, x3 = 2, x4 = 0 \text{ et } x5 = 0$$

D/ Résolution selon la méthode du simplexe avec MATLAB

1/ 1ère solution

clear all

% initialisation de variable de travail

o1=0

p1=0

q1=0

o2=0

p2=0

q2=0

o3=0

p3=0

q3=0

% matrice correspondant à : $x1 = 4$ équation (1)

$2 x2 = 12$ équation (2)

$3 x1 + 2 x2 = 18$ équation (3)

A=[1 0;0 2; 3 2]

% Matrice identité pour intégrer une variable de base par équation

I = eye (3)

%matrices représentant le système d'équation suivant :

$$x1 + x3 = 4 \quad (1)$$

$$2 x2 + x4 = 12 \quad (2)$$

$$3 x1 + 2 x2 + x5 = 18 \quad (3)$$

matrice=[A, I] % en rouge

cible= [4;12;18] %en vert

```
% Décomposition de la matrice par colonne afin d'identifier les x1,x2,x3,x4 et x5
C1=matrice(:,1) %x1
C2=matrice(:,2) %x2
C3=matrice(:,3) %x3
C4=matrice(:,4) %x4
C5=matrice(:,5) %x5

% B composé de 3 composantes parmi x1,x2, x3,x4 ou x5 tel que B x <= cible
% Max z = base * x où base représente 3 composantes représentatives de l'optimisation à
% résoudre soit l'équation 0   Z= 3x1 +5x2+0x3+0x4+0x5
% Objectif :  x1 et x2 variables de base
%             recherche maximum en intégrant successivement x3 et x4 et x5
%             sous la contrainte x1,x2,x3,x4 et x5 >0

% itération de base : x3 x4 x5 variables de base

B=[C3 C4 C5]

S0= inv(B)*cible

base= [0 0 0]

Max = base * S0

% variable x1 x2 de base x4 x5 sortent
% B et base composées de x1 x2 et x3

B=[C1 C2 C3];
S1= inv(B)*cible
base = [3 5 0];
Max1 = base * S1

% variable x1 x2 de base x3 x5 sortent
% B et base composées de x1 x2 et x4

B=[C1 C2 C4];
S2= inv(B)*cible
base = [3 5 0];
Max2 = base * S2

% variable x1 x2 de base x4 x3 sortent
% B et base composées de x1 x2 et x5

B=[C1 C5 C2];
S3= inv(B)*cible
base = [3 0 5];
Max3 = base * S3
```

% Affichage des valeurs trouvées en intégrant x1 et x2 successivement à x3 puis x4 puis x5

Max1

Max2

Max3

% 1^{er} cas Max1 supérieur à Max 2 et Max 3 ; validation si les composantes x(i) positives

% sinon recherche solution sur Max 2 et Max3

if ((Max1 >= Max2)&(Max1 >= Max3)&(S1(1,1)>=0)&(S1(2,1)>=0)&(S1(3,1)>=0))

disp('max1 est la solution');

Max1

disp('x1=');

S1(1,1)

disp('x2=');

S1(2,1)

else

p1=1;

end

if ((p1==1)&(Max2 >= Max3)&(S2(1,1)>=0) & (S2(2,1)>=0) & (S2(3,1)>=0))

disp('max2 est la solution');

Max2

disp('x1=');

S2(1,1)

disp('x2=');

S2(2,1)

else

o1=1;

end

if ((o1==1)&(S3(1,1)>=0) & (S3(2,1)>=0) & (S3(3,1)>=0))

disp('max3=est la solution');

Max3

disp('x1=');

S3(1,1)

disp('x2=');

S3(2,1)

else

q1=1

end

% 2^{ème} cas Max2 supérieur à Max 1 et Max 3 ; validation si les composantes x(i) positives

% sinon recherche solution sur Max 1 et Max3

```

if ((Max2 >= Max1)&( Max2 >= Max3)&(S2(1,1)>=0)&(S2(2,1)>=0)&(S2(3,1)>=0))
    disp('max1 est la solution');
    Max2
    disp('x1=');
    S2(1,1)
    disp('x2=');
    S2(2,1)
else
    p2=1;
end

```

```

if ((p2==1)&( Max1 >= Max3)&(S1(1,1)>=0) & (S1(2,1)>=0) & (S1(3,1)>=0))
    disp('max1 est la solution');
    Max1
    disp('x1=');
    S1(1,1)
    disp('x2=');
    S1(2,1)
else
    o2=1;
end

```

```

if ((o2==1)&(S3(1,1)>=0) & (S3(2,1)>=0) & (S3(3,1)>=0))
    disp('max3=est la solution');
    Max3
    disp('x1=');
    S3(1,1)
    disp('x2=');
    S3(2,1)
else
    q2=1;
end

```

% 3ème cas Max3 supérieur à Max 2 et Max 1 ; validation si les composantes x(i) positives
 % sinon recherche solution sur Max 2 et Max1

```

if ((Max3 >= Max1)&( Max3 >= Max2)&(S3(1,1)>=0)&(S3(2,1)>=0)&(S3(3,1)>=0))
    disp('max3 est la solution');
    Max3
    disp('x1=');
    S3(1,1)
    disp('x2=');

```

```

        S3(2,1)
    else

        p3=1;
    end

    if ((p3==1)&( Max1 >= Max2)&(S1(1,1)>=0) & (S1(2,1)>=0) & (S1(3,1)>=0))
        disp('max1 est la solution');
        Max1
        disp('x1=');
        S1(1,1)
        disp('x2=');
        S1(2,1)
    else

        o3=1;
    end

    if ((o3==1)&(S2(1,1)>=0) & (S2(2,1)>=0) & (S2(3,1)>=0))
        disp('max2=est la solution');
        Max2
        disp('x1=');
        S2(1,1)
        disp('x2=');
        S2(2,1)
    else

        q3=1;
    end

    end

    % cas où pas de solution

    if ((o1==1)& (p1==1)& (q1==1)&(o2==1)& (p2==1)& (q2==1)&(o3==1)& (p3==1)& (q3==1))

        disp('pas de solution');

    end

```

Résultats :

Max1 =

36

x1=

ans =

2

x2=

ans =

6

2/ 2^{ème} solution

il existe sous matlab une fonction permettant de minimiser une équation sous la contrainte de plusieurs autres : cette commande est « linprog »

La documentation indique :

Linear Programming with Equalities and Inequalities

Min Ft(x) such as Aeq . X = Beq
 A . X <= B
 X >= 0

The problem is

and you can load the matrices and vectors A, Aeq, b, beq, f and the lower bounds lb into the MATLAB workspace with

load sc50b

This problem in sc50b.mat has 48 variables, 30 inequalities and 20 equalities.

Comme nous cherchons à maximiser $F(x) = 3x_1 + 5x_2$, nous cherchons à minimiser $F(x) = -3x_1 - 5x_2$ sujet à :

$$x_1 \leq 4$$

$$2x_2 \leq 12$$

$$3x_1 + 2x_2 \leq 18$$

En conséquence

$$F = [-3 ; -5]$$

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 3 & 2 \end{bmatrix}$$

$$B = \begin{bmatrix} 4 & 12 & 18 \end{bmatrix}$$

Programme :

```
clear all
f=[-3 -5 ]
A=[ 1 0 ; 0 2;3 2]

B=[4 12 18]

lb = zeros (2,1)

[x,fval,exitflag,output,lambda] = linprog(f,A,B,[],[],lb)
```

% x est la matrice solution pour x1 et x2 et fval est la solution maximal est valeur absolue

Resultats :

x =

2.0000
6.0000

|fval| =

36.0000

9/ Cas rencontré dans l'industrie dans le cadre de mes fonctions de Responsable Maintenance

Dans le cadre des 35 heures, l'ensemble des collaborateurs a soumis le souhait d'avoir leur RTT le lundi ou le mercredi ou le vendredi.

On considère qu'au moins 50% de l'effectif est concerné par les rtt semaine.

X1 représente ces 50% de l'effectif .

X2 représente l'effectif prenant leur RTT le lundi.

X3 représente l'effectif prenant leur RTT le mercredi.

X4 représente l'effectif prenant leur RTT le vendredi.

La production impose un minimum d'effectif le lundi, le mercredi et le vendredi

Lundi : 21 personnes

Mercredi : 18 personnes

Vendredi : 16 personnes

L'effectif globale est toujours supérieur à 16.

L'objectif est de minimiser le nombre total de l'effectif ?

Soit minimise $Z = X1 + X2 + X3 + X4$ ou maximiser $Z = -X1 - X2 - X3 - X4$ sujet à :

$$X1 + X3 + X4 \geq 21 \text{ ou } -X1 - X3 - X4 \leq -21$$

$$X1 + X2 + X4 \geq 18 \text{ ou } -X1 - X2 - X4 \leq -18$$

$$X1 + X2 + X3 \geq 16 \text{ ou } -X1 - X2 - X3 \leq -16$$

$$X1 - X2 - X3 - X4 \geq 1 \text{ ou } -X1 + X2 + X3 + X4 \leq -1 \text{ (1 pour s'assurer des 50% d'effectif)}$$

$$2 X1 \geq 16$$

Programme sous matlab

Remarque : Le fait d'avoir autant de variables que d'équation à résoudre simplifie le programme sous matlab car l'ensemble des variables de base correspond aux variables de l'équation à maximiser

%BOCCHIERI

```
%Projet.m
%Methode du simplexe
```

```
clear all
```

```
% variable de travail
```

```
o1=0
p1=0
q1=0
r1=0
s1=0
o2=0
p2=0
q2=0
r2=0
s2=0
o3=0
p3=0
q3=0
r3=0
s3=0
o4=0
p4=0
q4=0
r4=0
s4=0
o5=0
p5=0
q5=0
r5=0
s5=0
```

```
% A et cible matrices représentant le système d'équations
```

```
A=[-1 0 -1 -1;-1 -1 0 -1;-1 -1 -1 0;-1 1 1 1;-2 0 0 0]
```

```
I = eye (5)
```

```
matrice =[A, I]
```

```
cible= [-21;-18;-16;-1;-16]
```

```
% découpage de la matrice par colonne x1 x2 x3 x4 x5 x6 x7x8 x9
```

```
C1=matrice(:,1)
C2=matrice(:,2)
C3=matrice(:,3)
```

```
C4=matrice(:,4)
C5=matrice(:,5)
C6=matrice(:,6)
C7=matrice(:,7)
C8=matrice(:,8)
C9=matrice(:,9)
```

% itération de base x5 x6 x7 x8 x9 variables de base

```
B=[C5 C6 C7 C8 C9]
```

```
cible1= inv(B)*cible
```

```
base= [0 0 0 0 0]
```

```
Max = base * cible1
```

% variable x1 x2 x3 x4 x5 de base

```
B=[C1 C2 C3 C4 C5];
S1= inv(B)*cible
base = [-1 -1 -1 -1 0];
Max1 = base * S1
```

% variable x1 x2 x3 x4 x6 de base

```
B=[C1 C2 C3 C4 C6];
S2= inv(B)*cible
base = [-1 -1 -1 -1 0];
Max2 = base * S2
```

% variable x1 x2 x3 x4 x7 de base

```
B=[C1 C2 C3 C4 C7];
S3= inv(B)*cible
base = [-1 -1 -1 -1 0];
Max3 = base * S3
```

% variable x1 x2 x3 x4 x8 de base

```
B=[C1 C2 C3 C4 C8];
S4= inv(B)*cible
base = [-1 -1 -1 -1 0];
Max4 = base * S4
```

% variable x1 x2 x3 x4 x9 de base

```
B=[C1 C2 C3 C4 C9];
S5= inv(B)*cible
base = [-1 -1 -1 -1 0];
Max5 = base * S5
```

```
Max1
Max2
Max3
Max4
Max5
```

% Max1 supérieur

```
if ((Max1 >= Max2)&( Max1 >= Max3)&( Max1 >= Max4)&( Max1 >=
Max5)&(S1(1,1)>=0)&(S1(2,1)>=0)&(S1(3,1)>=0)&(S1(4,1)>=0)&(S1(5,1)>=0))
    disp('max1 est la solution');
    Max1
    disp('x1=');
    S1(1,1)
    disp('x2=');
    S1(2,1)
    disp('x3=');
    S1(3,1)
    disp('x4=');
    S1(4,1)
else
    p1=1;
end
```

```
if ((p1==1)&( Max2 >= Max3)&( Max2 >= Max4)&( Max2 >=
Max5)&(S2(1,1)>=0)&(S2(2,1)>=0)&(S2(3,1)>=0)&(S2(4,1)>=0)&(S2(5,1)>=0))
    disp('max2 est la solution');
    Max2
    disp('x1=');
    S2(1,1)
    disp('x2=');
    S2(2,1)
    disp('x3=');
    S2(3,1)
    disp('x4=');
    S2(4,1)
else
    o1=1;
```

end

```
if ((o1==1)&( Max3 >= Max4)&( Max3 >= Max5)&(S3(1,1)>=0) & (S3(2,1)>=0) &
(S3(3,1)>=0)& (S3(4,1)>=0)& (S3(5,1)>=0))
```

```
    disp('max3=est la solution');
```

```
    Max3
```

```
    disp('x1=');
```

```
    S3(1,1)
```

```
    disp('x2=');
```

```
    S3(2,1)
```

```
    disp('x3=');
```

```
    S3(3,1)
```

```
    disp('x4=');
```

```
    S3(4,1)
```

```
else
```

```
    q1=1;
```

end

```
if ((q1==1)&( Max4 >= Max5)&(S4(1,1)>=0) & (S4(2,1)>=0) & (S4(3,1)>=0)& (S4(4,1)>=0)&
(S4(5,1)>=0))
```

```
    disp('max4=est la solution');
```

```
    Max4
```

```
    disp('x1=');
```

```
    S4(1,1)
```

```
    disp('x2=');
```

```
    S4(2,1)
```

```
    disp('x3=');
```

```
    S4(3,1)
```

```
    disp('x4=');
```

```
    S4(4,1)
```

```
else
```

```
    r1=1;
```

end

```
if ((r1==1)&(S5(1,1)>=0) & (S5(2,1)>=0) & (S5(3,1)>=0)& (S5(4,1)>=0)& (S5(5,1)>=0))
```

```
    disp('max5=est la solution');
```

```
    Max5
```

```
    disp('x1=');
```

```
    S5(1,1)
```

```
    disp('x2=');
```

```
    S5(2,1)
```

```
    disp('x3=');
```

```
    S5(3,1)
```

```
    disp('x4=');
```

```

        S5(4,1)
    else

        s1=1;

    end

% Max2 supérieur

if ((Max2 >= Max1)&( Max2 >= Max3)&( Max2 >= Max4)&( Max2 >=
Max5)&(S2(1,1)>=0)&(S2(2,1)>=0)&(S2(3,1)>=0)&(S2(4,1)>=0)&(S2(5,1)>=0))
    disp('max2 est la solution');
    Max2
    disp('x1=');
    S2(1,1)
    disp('x2=');
    S2(2,1)
    disp('x3=');
    S2(3,1)
    disp('x4=');
    S2(4,1)
else

    p2=1;
end

if ((p2==1)&( Max1 >= Max3)&( Max1 >= Max4)&( Max1 >=
Max5)&(S1(1,1)>=0)&(S1(2,1)>=0)&(S1(3,1)>=0)&(S1(4,1)>=0)&(S1(5,1)>=0))
    disp('max1 est la solution');
    Max1
    disp('x1=');
    S1(1,1)
    disp('x2=');
    S1(2,1)
    disp('x3=');
    S1(3,1)
    disp('x4=');
    S1(4,1)
else

    o2=1;
end

if ((o2==1)&( Max3 >= Max4)&( Max3 >= Max5)&(S3(1,1)>=0) & (S3(2,1)>=0) &
(S3(3,1)>=0)& (S3(4,1)>=0)& (S3(5,1)>=0))
    disp('max3=est la solution');
    Max3

```

```

disp('x1=');
S3(1,1)
disp('x2=');
S3(2,1)
disp('x3=');
S3(3,1)
disp('x4=');
S3(4,1)
else

    q2=1;

end

if ((q2==1)&( Max4 >= Max5)&(S4(1,1)>=0) & (S4(2,1)>=0) & (S4(3,1)>=0)& (S4(4,1)>=0)&
(S4(5,1)>=0))
    disp('max4=est la solution');
    Max4
    disp('x1=');
    S4(1,1)
    disp('x2=');
    S4(2,1)
    disp('x3=');
    S4(3,1)
    disp('x4=');
    S4(4,1)
else

    r2=1;

end

if ((r2==1)&(S5(1,1)>=0) & (S5(2,1)>=0) & (S5(3,1)>=0)& (S5(4,1)>=0)& (S5(5,1)>=0))
    disp('max5=est la solution');
    Max5
    disp('x1=');
    S5(1,1)
    disp('x2=');
    S5(2,1)
    disp('x3=');
    S5(3,1)
    disp('x4=');
    S5(4,1)
else

    s2=1;

end

```

% Max3 supérieur

```

if ((Max3 >= Max1)&( Max3 >= Max2)&( Max3 >= Max4)&( Max3 >=
Max5)&(S3(1,1)>=0)&(S3(2,1)>=0)&(S3(3,1)>=0)&(S3(4,1)>=0)&(S3(5,1)>=0))
    disp('max3 est la solution');
    Max3
    disp('x1=');
    S3(1,1)
    disp('x2=');
    S3(2,1)
    disp('x3=');
    S3(3,1)
    disp('x4=');
    S3(4,1)
else
    p3=1;
end

```

```

if ((p3==1)&( Max1 >= Max2)&( Max1 >= Max4)&( Max1 >=
Max5)&(S1(1,1)>=0)&(S1(2,1)>=0)&(S1(3,1)>=0)&(S1(4,1)>=0)&(S1(5,1)>=0))
    disp('max1 est la solution');
    Max1
    disp('x1=');
    S1(1,1)
    disp('x2=');
    S1(2,1)
    disp('x3=');
    S1(3,1)
    disp('x4=');
    S1(4,1)
else
    o3=1;
end

```

```

if ((o3==1)&( Max2 >= Max4)&( Max2 >= Max5)&(S2(1,1)>=0) & (S2(2,1)>=0) &
(S2(3,1)>=0)& (S2(4,1)>=0)& (S2(5,1)>=0))
    disp('max2=est la solution');
    Max2
    disp('x1=');
    S2(1,1)
    disp('x2=');
    S2(2,1)
    disp('x3=');
    S2(3,1)
    disp('x4=');

```

```

        S2(4,1)
    else

        q3=1;

    end

    if ((q3==1)&( Max4 >= Max5)&(S4(1,1)>=0) & (S4(2,1)>=0) & (S4(3,1)>=0)& (S4(4,1)>=0)&
    (S4(5,1)>=0))
        disp('max4=est la solution');
        Max4
        disp('x1=');
        S4(1,1)
        disp('x2=');
        S4(2,1)
        disp('x3=');
        S4(3,1)
        disp('x4=');
        S4(4,1)
    else

        r3=1;

    end

    if ((r3==1)&(S5(1,1)>=0) & (S5(2,1)>=0) & (S5(3,1)>=0)& (S5(4,1)>=0)& (S5(5,1)>=0))
        disp('max5=est la solution');
        Max5
        disp('x1=');
        S5(1,1)
        disp('x2=');
        S5(2,1)
        disp('x3=');
        S5(3,1)
        disp('x4=');
        S5(4,1)
    else

        s3=1;

    end

    % Max4 supérieur

    if ((Max4 >= Max1)&( Max4 >= Max2)&( Max4 >= Max3)&( Max4 >=
    Max5)&(S4(1,1)>=0)&(S4(2,1)>=0)&(S4(3,1)>=0)&(S4(4,1)>=0)&(S4(5,1)>=0))
        disp('max4 est la solution');

```

```

Max4
disp('x1=');
S4(1,1)
disp('x2=');
S4(2,1)
disp('x3=');
S4(3,1)
disp('x4=');
S4(4,1)
else

    p4=1;
end

if ((p4==1)&( Max1 >= Max2)&( Max1 >= Max3)&( Max1 >=
Max5)&(S1(1,1)>=0)&(S1(2,1)>=0)&(S1(3,1)>=0)&(S1(4,1)>=0)&(S1(5,1)>=0))
    disp('max1 est la solution');
    Max1
    disp('x1=');
    S1(1,1)
    disp('x2=');
    S1(2,1)
    disp('x3=');
    S1(3,1)
    disp('x4=');
    S1(4,1)
else

    o4=1;
end

if ((o4==1)&( Max2 >= Max3)&( Max2 >= Max5)&(S2(1,1)>=0) & (S2(2,1)>=0) &
(S2(3,1)>=0)& (S2(4,1)>=0)& (S2(5,1)>=0))
    disp('max2=est la solution');
    Max2
    disp('x1=');
    S2(1,1)
    disp('x2=');
    S2(2,1)
    disp('x3=');
    S2(3,1)
    disp('x4=');
    S2(4,1)
else

    q4=1;

```

end

```
if ((q4==1)&( Max3 >= Max5)&(S3(1,1)>=0) & (S3(2,1)>=0) & (S3(3,1)>=0)& (S3(4,1)>=0)&
(S3(5,1)>=0))
```

```
    disp('max3=est la solution');
```

```
    Max3
```

```
    disp('x1=');
```

```
    S3(1,1)
```

```
    disp('x2=');
```

```
    S3(2,1)
```

```
    disp('x3=');
```

```
    S3(3,1)
```

```
    disp('x4=');
```

```
    S3(4,1)
```

```
else
```

```
    r4=1;
```

end

```
if ((r4==1)&(S5(1,1)>=0) & (S5(2,1)>=0) & (S5(3,1)>=0)& (S5(4,1)>=0)& (S5(5,1)>=0))
```

```
    disp('max5=est la solution');
```

```
    Max5
```

```
    disp('x1=');
```

```
    S5(1,1)
```

```
    disp('x2=');
```

```
    S5(2,1)
```

```
    disp('x3=');
```

```
    S5(3,1)
```

```
    disp('x4=');
```

```
    S5(4,1)
```

```
else
```

```
    s4=1;
```

end

% Max5 supérieur

```
if ((Max5 >= Max1)&( Max5 >= Max2)&( Max5 >= Max3)&( Max5 >=
Max4)&(S5(1,1)>=0)&(S5(2,1)>=0)&(S5(3,1)>=0)&(S5(4,1)>=0)&(S5(5,1)>=0))
```

```
    disp('max5 est la solution');
```

```
    Max5
```

```
    disp('x1=');
```

```
    S5(1,1)
```

```
    disp('x2=');
```

```
    S5(2,1)
```

```

disp('x3=');
S5(3,1)
disp('x4=');
S5(4,1)
else

    p5=1;
end

if ((p5==1)&( Max1 >= Max2)&( Max1 >= Max3)&( Max1 >=
Max4)&(S1(1,1)>=0)&(S1(2,1)>=0)&(S1(3,1)>=0)&(S1(4,1)>=0)&(S1(5,1)>=0))
    disp('max1 est la solution');
    Max1
    disp('x1=');
    S1(1,1)
    disp('x2=');
    S1(2,1)
    disp('x3=');
    S1(3,1)
    disp('x4=');
    S1(4,1)
else

    o5=1;
end

if ((o5==1)&( Max2 >= Max3)&( Max2 >= Max4)&(S2(1,1)>=0) & (S2(2,1)>=0) &
(S2(3,1)>=0)& (S2(4,1)>=0)& (S2(5,1)>=0))
    disp('max2=est la solution');
    Max2
    disp('x1=');
    S2(1,1)
    disp('x2=');
    S2(2,1)
    disp('x3=');
    S2(3,1)
    disp('x4=');
    S2(4,1)
else

    q5=1;
end

if ((q5==1)&( Max3 >= Max4)&(S3(1,1)>=0) & (S3(2,1)>=0) & (S3(3,1)>=0)& (S3(4,1)>=0)&
(S3(5,1)>=0))
    disp('max3=est la solution');

```

```

Max3
disp('x1=');
S3(1,1)
disp('x2=');
S3(2,1)
disp('x3=');
S3(3,1)
disp('x4=');
S3(4,1)
else

r5=1;

end

if ((r5==1)&(S4(1,1)>=0) & (S4(2,1)>=0) & (S4(3,1)>=0)& (S4(4,1)>=0)& (S4(5,1)>=0))
    disp('max4=est la solution');
    Max4
    disp('x1=');
    S4(1,1)
    disp('x2=');
    S4(2,1)
    disp('x3=');
    S4(3,1)
    disp('x4=');
    S4(4,1)
else

s5=1;

end

```

% cas où pas de solution

```

if ((o1==1)& (p1==1)& (q1==1)& (r1==1)& (s1==1)&(o2==1)& (p2==1)& (q2==1)&
(r2==1)& (s2==1)&(o3==1)& (p3==1)& (q3==1)& (r3==1)& (s3==1)&(o4==1)& (p4==1)&
(q4==1)& (r4==1)& (s4==1)&(o5==1)& (p5==1)& (q5==1)& (r5==1)& (s5==1))
disp('pas de solution');
end

```

Résultat :

max5=est la solution

Max5 =

-21.8000

x1=

ans =

11.4000

x2=

ans =

0.8000

x3=

ans =

3.8000

x4=

ans =

5.8000

Ces résultats nous indiquent que l'effectif peut être estimé à 22 personnes avec une répartition pour les RTT 1 le lundi, 4 le mercredi, 6 le vendredi.

Annexe