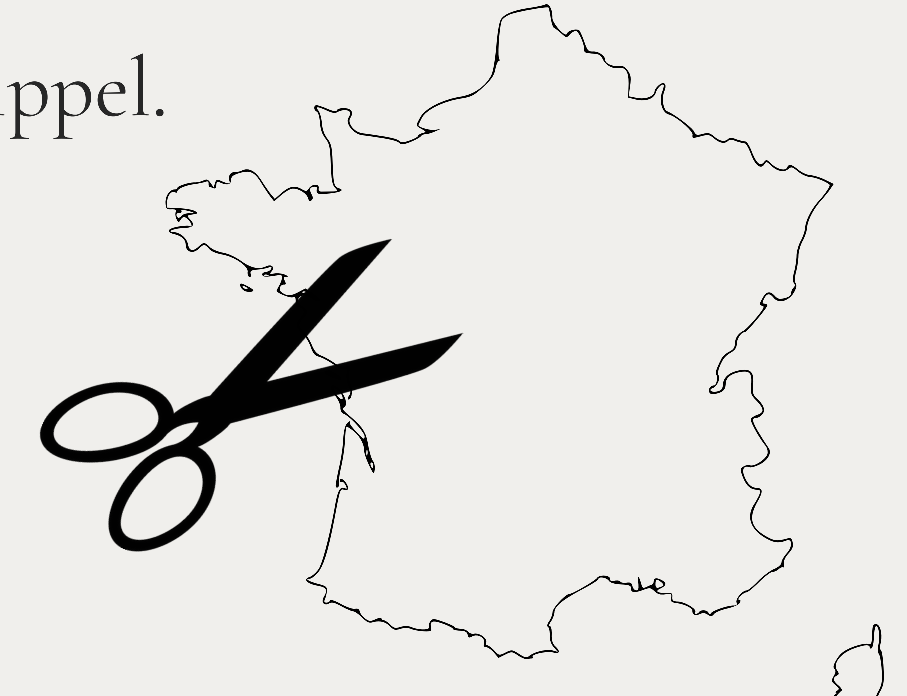
Découpage électoral

Victor LESUR, Camille GOUJET et Israa BEN SASSI
à l'attention de Arnauld Knippel.



INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES ROUEN NORMANDIE

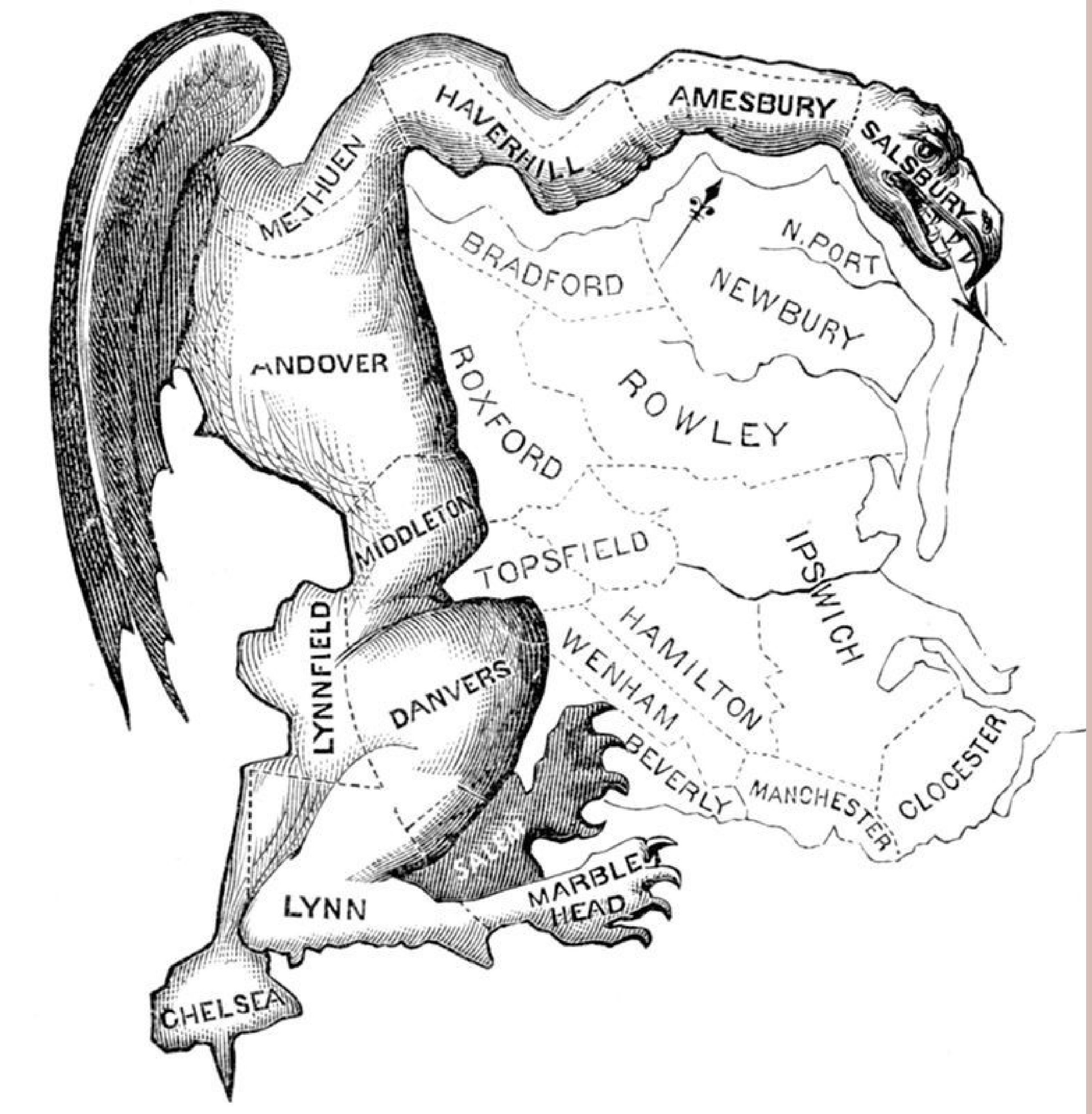
Sommaire

Sommaire

Sommaire

I. Introduction

I.1. Quelques définitions



1.Gerrymandering

gouverneur Elbridge Gerry.
Un découpage de la forme d'une salamandre.

2.Découpage électoral

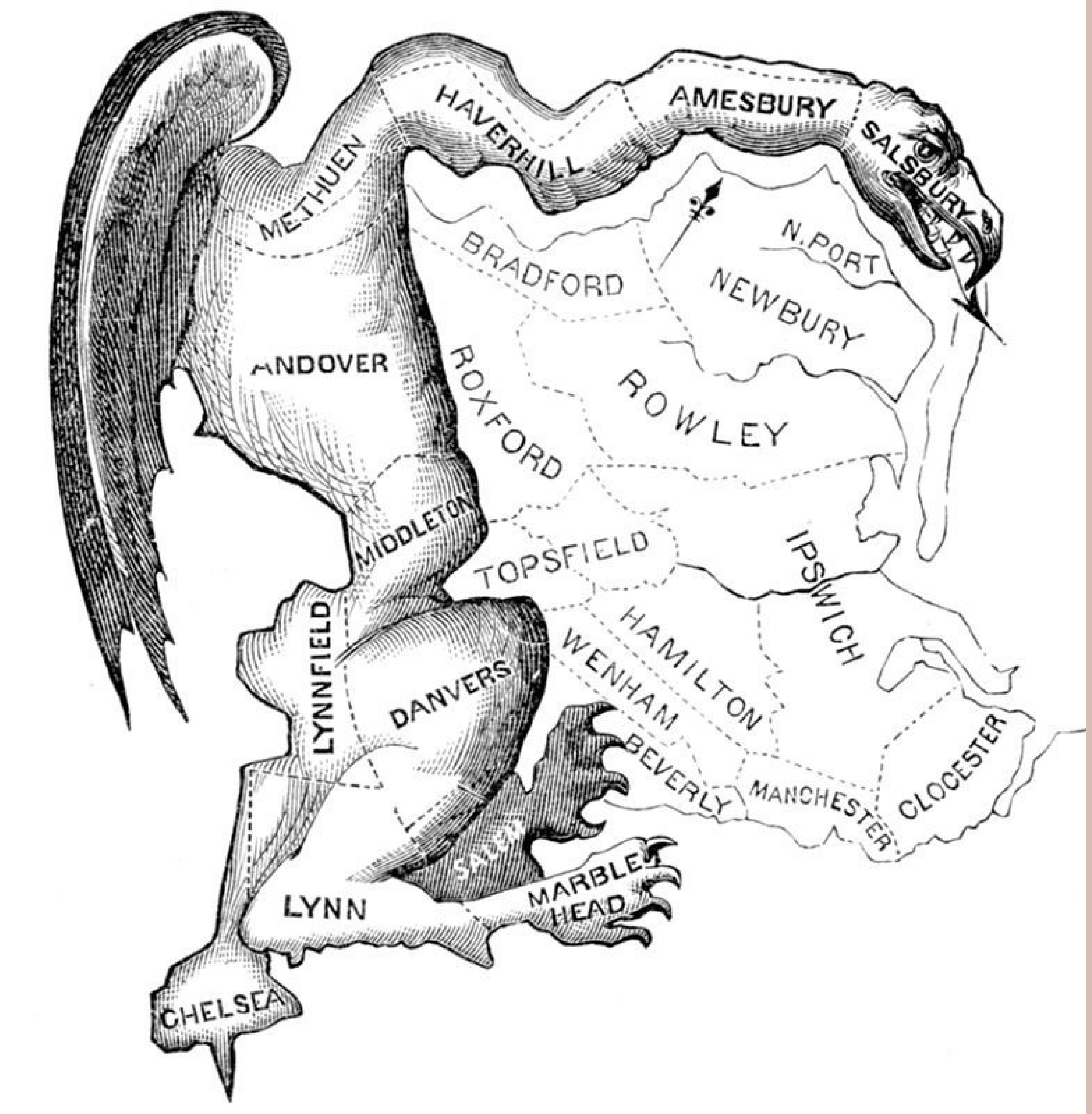
Le mécanisme pour subdiviser le territoire en circonscriptions électorales

3.Circonscription électorale

Une division du territoire effectuée dans le cadre d'une élection.
Chaque citoyen est rattaché à une circonscription et à une seule dans le cadre d'un vote.

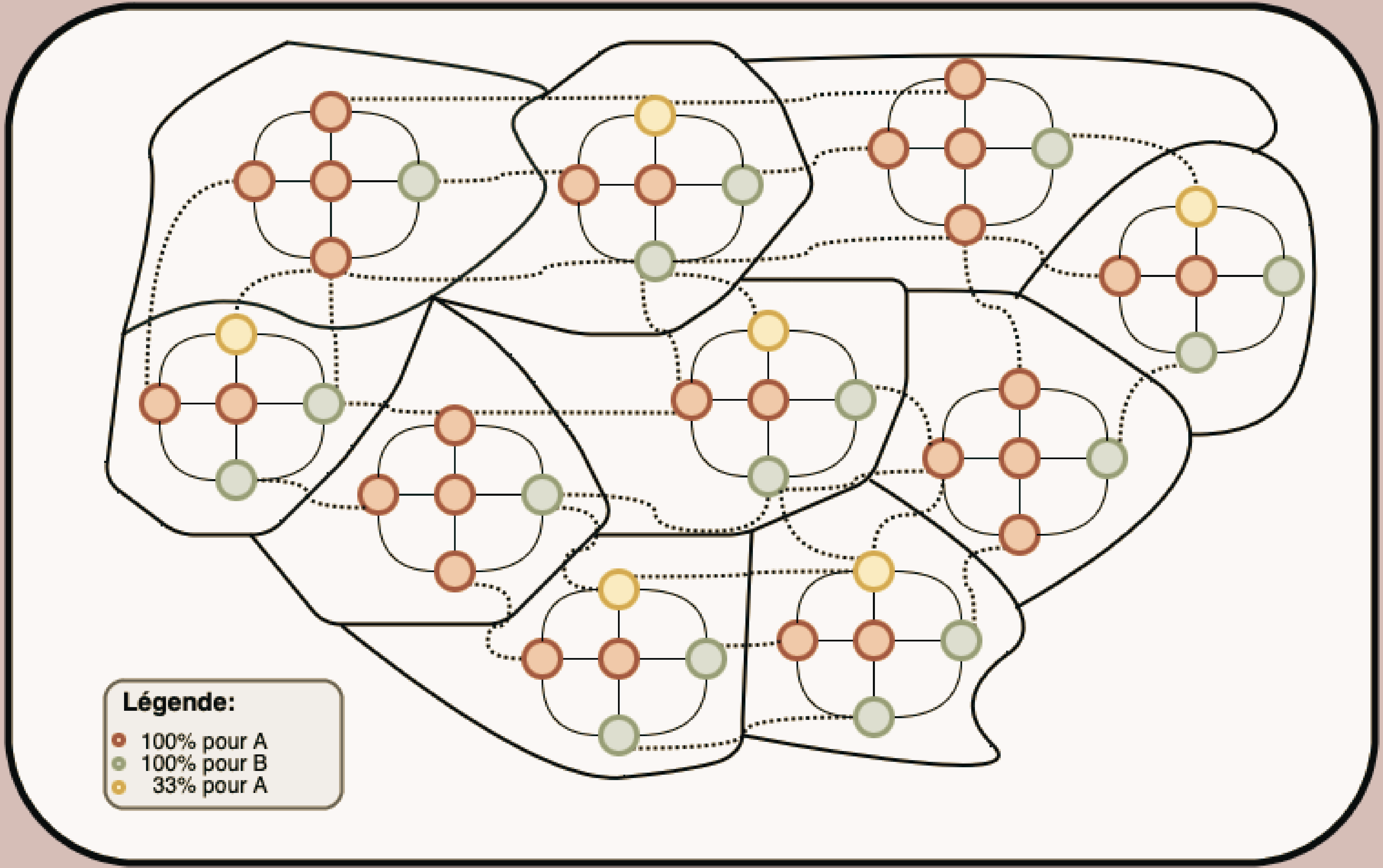
I. Introduction

I.2. Objectifs



II. Partie Théorique

1. Exemple du problème



10 000 électeurs
10 circonscriptions de 1 000 électeurs
Deux partis A et B
Le premier recueillant 4 000 suffrages
Le second 6 000.

- 6 circonscriptions gagnées par le parti A avec 53,3% des voix soit 3200 voix pour A et 2800 voix pour B
- 4 circonscriptions gagnées par le parti B avec 80% des voix, soit 800 voix pour A et 3200 voix pour B.

2. Énoncé

2. Énoncé

Les données: un graphe G=(X,U)

X={x1,x2,...,xn}

xi correspond à un canton.

nbVi: Le nombre de votants du canton.

pVi: Le pourcentage de votants pour A..

2. Énoncé

Les données: un graphe G=(X,U)

X={x1,x2,...,xn}

xi correspond à un canton.

nbVi: Le nombre de votants du canton.

pVi: Le pourcentage de votants pour A..

Xi
nbVi
pVi

2. Énonce

Les données: un graphe G=(X,U)

X={x1,x2,...,xn}

xi correspond à un canton.

nbVi: Le nombre de votants du canton.

pVi: Le pourcentage de votants pour A..

Xi
nbVi
pVi

on définit la **fonction p** qui calcule le poids du parti A dans une circonscription donnée

$$p : X \to \mathbb{R}$$
$$K_j \mapsto p(K_j)$$

$$p(K_j) = \frac{\sum_{i=1, x_i \in K_j}^{n} nbV_i \times pV_i}{\sum_{i=1, x_i \in K_j}^{n} nbV_i}$$

3. La fonction économique

$$f : X \to \mathbb{R}$$
$$K_j \mapsto f(K_j)$$

$$f(K_j) = \begin{cases} 1 & \text{si } p(K_j) \geq 0,5 \\ 0 & \text{sinon} \end{cases}$$

3. La fonction économique

$$f : X \to \mathbb{R}$$
$$K_j \mapsto f(K_j)$$

$$f(K_j) = \begin{cases} 1 & \text{si } p(K_j) \geq 0,5 \\ 0 & \text{sinon} \end{cases}$$

4.Sens

- Faire gagner le plus de circonscriptions au parti A $max\sum_{j=1}^{k} f(K_j)$
- Faire gagner le plus de circonscriptions au parti B $min\sum_{j=1}^{k} f(K_j)$

5. Choix de variable

Afin de pouvoir formuler le problème sous forme d'un problème mathématique,
Nous allons définir des variables

$$x_{ij} = \begin{cases} 1 & \text{si } x_i \in K_j \\ 0 & \text{sinon} \end{cases}$$

5. Choix de variable

Afin de pouvoir formuler le problème sous forme d'un problème mathématique, Nous allons définir des variables

$$x_{ij} = \begin{cases} 1 & \text{si } x_i \in K_j \\ 0 & \text{sinon} \end{cases}$$

En explicitant les variables de la sorte, on peut alors redéfinir la fonction $p(K_j)$ précédente :

$$p(K_j) = \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} \qquad \textbf{(1)}$$

6. Choix des contraintes

- **Règle:** le nombre de votants d'une circonscription doit être compris entre **80%** et **120%** du nombre moyen de votants dans une circonscription.

$$\frac{0,8}{k}\sum_{i=1}^{n} nbV_i \leq \sum_{i=1,x_i \in K_j}^{n} nbV_i \leq \frac{1,20}{k}\sum_{i=1}^{n} nbV_i \qquad \textbf{(2)}$$

6. Choix des contraintes

- **Règle:** le nombre de votants d'une circonscription doit être compris entre **80%** et **120%** du nombre moyen de votants dans une circonscription.

$$\frac{0,8}{k}\sum_{i=1}^{n} nbV_i \leq \sum_{i=1, x_i \in K_j}^{n} nbV_i \leq \frac{1,20}{k}\sum_{i=1}^{n} nbV_i \qquad \textbf{(2)}$$

- la variable binaire x_{ij} traduit l'appartenance d'un canton à une circonscription

$$\forall i \in \{1, \ldots, n\}, \sum_{j=1}^{k} x_{ij} = 1 \qquad \textbf{(3)}$$

$$\forall j \in \{1, \ldots, k\}, \sum_{i=1}^{n} x_{ij} \geq 1 \qquad \textbf{(4)}$$

6. Choix des contraintes

- **Règle:** le nombre de votants d'une circonscription doit être compris entre **80%** et **120%** du nombre moyen de votants dans une circonscription.

$$\frac{0,8}{k}\sum_{i=1}^{n} nbV_i \leq \sum_{i=1, x_i \in K_j}^{n} nbV_i \leq \frac{1,20}{k}\sum_{i=1}^{n} nbV_i \qquad \textbf{(2)}$$

- la variable binaire x_{ij} traduit l'appartenance d'un canton à une circonscription

$$\forall i \in \{1, \ldots, n\}, \sum_{j=1}^{k} x_{ij} = 1 \qquad \textbf{(3)}$$

$$\forall j \in \{1, \ldots, k\}, \sum_{i=1}^{n} x_{ij} \geq 1 \qquad \textbf{(4)}$$

- Contrainte de connexité

Soit $G_j = (K_j, U_j)$ le sous-graphe représentant la circonscription j

$$x_{ij} \times x_{lj} \times d_{il} \leq D \qquad \textbf{(5)}$$

7. Formulation du problème

$$\max(z) = \sum_{j=1}^{k} \lfloor 2 \left(\frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} \right) \rfloor$$

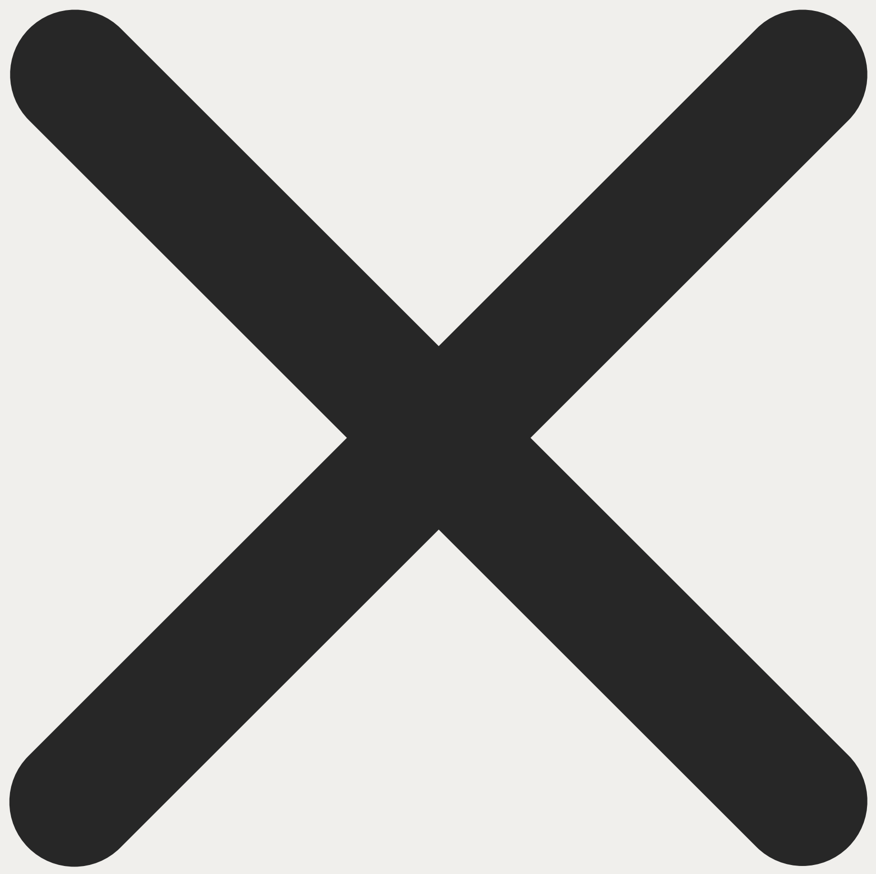
sous $\sum_{i=1}^{n} x_{ij} \times nbV_i \leq \frac{1,20}{k} \sum_{i=1}^{n} nbV_i$ $\quad \forall i \in \{1, \ldots, n\}$

$\sum_{i=1}^{n} x_{ij} \times nbV_i \geq \frac{0,8}{k} \sum_{i=1}^{n} nbV_i$ $\quad \forall j \in \{1, \ldots, k\}$

$\sum_{j=1}^{k} x_{ij} = 1$ $\quad \forall j \in \{1, \ldots, k\}$

$\sum_{i=1}^{n} x_{ij} \geq 1$ $\quad \forall j \in \{1, \ldots, k\}$

$x_{ij} \in \{0, 1\}$ $\quad \forall (i, j) \in \{1, \ldots, n\} \times \{1, \ldots, k\}$



7. Formulation du problème

$$z_j = \begin{cases} 1 & si \quad \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} \geq 0.5 \\ 0 & si \quad \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} < 0.5 \end{cases}$$

7. Formulation du problème

$$z_j = \begin{cases} 1 & si \quad \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} \geq 0.5 \\ 0 & si \quad \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} < 0.5 \end{cases} \longrightarrow \quad \forall j \in \{1, \ldots, k\}, z_j \leq \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} + \frac{1}{2}$$

7. Formulation du problème

$$z_j = \begin{cases} 1 & si \quad \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} \geq 0.5 \\ 0 & si \quad \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} < 0.5 \end{cases} \longrightarrow \quad \forall j \in \{1, \ldots, k\}, z_j \leq \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} + \frac{1}{2}$$

$$z_j \times \sum_{i=1}^{n} x_{ij} \times nbV_i \leq \sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i + \frac{1}{2} \sum_{i=1}^{n} x_{ij} \times nbV_i$$

7. Formulation du problème

$$z_j = \begin{cases} 1 & si \quad \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} \geq 0.5 \\ 0 & si \quad \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} < 0.5 \end{cases} \longrightarrow \quad \forall j \in \{1, \ldots, k\}, z_j \leq \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} + \frac{1}{2}$$

$$z_j \times \sum_{i=1}^{n} x_{ij} \times nbV_i \leq \sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i + \frac{1}{2} \sum_{i=1}^{n} x_{ij} \times nbV_i$$

On pose
$$\begin{cases} y_{ij} \leq z_j & \text{(a)} \\ y_{ij} \leq x_{ij} & \text{(b)} \\ y_{ij} \geq z_j + x_{ij} - 1 & \text{(c)} \end{cases}$$

7. Formulation du problème

$$z_j = \begin{cases} 1 & si \quad \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} \geq 0.5 \\ 0 & si \quad \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} < 0.5 \end{cases} \longrightarrow \quad \forall j \in \{1, \dots, k\}, z_j \leq \frac{\sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i}{\sum_{i=1}^{n} x_{ij} \times nbV_i} + \frac{1}{2}$$

$$z_j \times \sum_{i=1}^{n} x_{ij} \times nbV_i \leq \sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i + \frac{1}{2} \sum_{i=1}^{n} x_{ij} \times nbV_i$$

On pose

$$\begin{cases} y_{ij} \leq z_j & \text{(a)} \\ y_{ij} \leq x_{ij} & \text{(b)} \\ y_{ij} \geq z_j + x_{ij} - 1 & \text{(c)} \end{cases}$$

Et de même pour la connexité, on introduit une nouvelle variable telle que :

$$\begin{cases} w_{ijl} \leq x_{ij} \\ w_{ijl} \leq x_{kj} \\ w_{ijl} \geq x_{lj} + x_{ij} - 1 \\ d_{il} \leq D \end{cases}$$

7. Formulation du problème

$$\max(z) = \sum_{j=1}^{k} z_j$$

sous

$$\begin{array}{ll}
\sum_{j=1}^{k} x_{ij} = 1 & \forall i \in \{1,\dots,n\} \\
\sum_{i=1}^{n} x_{ij} \geq 1 & \forall j \in \{1,\dots,k\} \\
\sum_{i=1}^{n} x_{ij} \times nbV_i \leq \frac{1{,}20}{k} \sum_{i=1}^{n} nbV_i & \forall j \in \{1,\dots,k\} \\
\sum_{i=1}^{n} x_{ij} \times nbV_i \geq \frac{0{,}8}{k} \sum_{i=1}^{n} nbV_i & \forall j \in \{1,\dots,k\} \\
x_{ij} \in \{0,1\} & \forall (i,j) \in \{1,\dots,n\} \times \{1,\dots,k\} \\
z_j \in \{0,1\} & \forall j \in \{1,\dots,k\} \\
y_{ij} \in \{0,1\} & \forall (i,j) \in \{1,\dots,n\} \times \{1,\dots,k\} \\
\sum_{i=1}^{n} y_{ij} \times nbV_i \leq \sum_{i=1}^{n} x_{ij} \times nbV_i \times pV_i + \frac{1}{2} \sum_{i=1}^{n} x_{ij} \times nbV_i & \forall (i,j) \in \{1,\dots,n\} \times \{1,\dots,k\} \\
y_{ij} \leq z_j & \forall (i,j) \in \{1,\dots,n\} \times \{1,\dots,k\} \\
y_{ij} \leq x_{ij} & \forall (i,j) \in \{1,\dots,n\} \times \{1,\dots,k\} \\
y_{ij} \geq z_j + x_{ij} - 1 & \forall (i,j) \in \{1,\dots,n\} \times \{1,\dots,k\} \\
d_{il} \leq D & \forall i,l \in \{1,\dots,n\}^2 \\
w_{ijl} \leq x_{ij} & \forall (i,l,j) \in \{1,\dots,n\}^2 \times \{1,\dots,k\} \\
w_{ijl} \leq x_{lj} & \forall (i,l,j) \in \{1,\dots,n\}^2 \times \{1,\dots,k\} \\
w_{ijl} \geq x_{lj} + x_{ij} - 1 & \forall (i,l,j) \in \{1,\dots,n\}^2 \times \{1,\dots,k\}
\end{array}$$

III. Partie pratique

1. Application sur un toy example

Regrouper au mieux les cantons en deux circonscriptions pour maximiser les élus du camp A

$$\begin{cases} \text{2 circonscriptions} \\ \text{5 cantons} \end{cases} \longrightarrow \begin{cases} k=2 \\ n=5 \end{cases}$$

$$\begin{cases} nbV_1 = 5 & pV_1 = 0.4 \\ nbV_2 = 6 & pV_2 = 0.6 \\ nbV_3 = 5 & pV_3 = 0.7 \\ nbV_4 = 6 & pV_4 = 0.2 \\ nbV_5 = 7 & pV_5 = 0.9 \end{cases}$$

$$\max(z) = z_1 + z_2$$

sous

$$
\begin{aligned}
& x_{i1} + x_{i2} = 1 && \forall i \in \{1,\ldots,5\} \\
& x_{1j} + x_{2j} + x_{3j} + x_{4j} + x_{5j} \geq 1 && \forall j \in \{1,\ldots,2\} \\
& x_{1j}nbV_1 + x_{2j}nbV_2 + x_{3j}nbV_3 + x_{4j}nbV_4 + x_{5j}nbV_5 \ \leq \ \tfrac{1{,}20}{2}\textstyle\sum_{i=1}^{5} nbV_i && \forall j \in \{1,\ldots,2\} \\
& x_{1j}nbV_1 + x_{2j}nbV_2 + x_{3j}nbV_3 + x_{4j}nbV_4 + x_{5j}nbV_5 \ \geq \ \tfrac{0{,}8}{2}\textstyle\sum_{i=1}^{5} nbV_i && \forall j \in \{1,\ldots,2\} \\
& x_{ij} \in \{0,1\} && \forall (i,j) \in \{1,\ldots,n\} \times \{1,\ldots,2\} \\
& z_j \in \{0,1\} && \forall j \in \{1,\ldots,2\} \\
& y_{ij} \in \{0,1\} && \forall (i,j) \in \{1,\ldots,5\} \times \{1,\ldots,2\} \\
& \textstyle\sum_{i=1}^{5} y_{ij} \times nbV_i \leq \sum_{i=1}^{5} x_{ij} \times nbV_i \times pV_i + \frac{1}{2}\sum_{i=1}^{5} x_{ij} \times nbV_i && \forall (i,j) \in \{1,\ldots,5\} \times \{1,\ldots,2\} \\
& y_{ij} \leq z_j && \forall (i,j) \in \{1,\ldots,5\} \times \{1,\ldots,2\} \\
& y_{ij} \leq x_{ij} && \forall (i,j) \in \{1,\ldots,5\} \times \{1,\ldots,2\} \\
& y_{ij} \geq z_j + x_{ij} - 1 && \forall (i,j) \in \{1,\ldots,5\} \times \{1,\ldots,2\} \\
& d_{il} \leq D && \forall i,l \in \{1,\ldots,5\}^2 \\
& w_{ijl} \leq x_{ij} && \forall (i,l,j) \in \{1,\ldots,5\}^2 \times \{1,\ldots,2 \\
& w_{ijl} \leq x_{lj} && \forall (i,l,j) \in \{1,\ldots,5\}^2 \times \{1,\ldots,2 \\
& w_{ijl} \geq x_{lj} + x_{ij} - 1 && \forall (i,l,j) \in \{1,\ldots,5\}^2 \times \{1,\ldots,2
\end{aligned}
$$

Démonstration

Resolution dans l'optique de faire perdre le parti A

Solution

```
sol = m.solve()
```

```
sol.display()
```

```
solution for: Découpage éléctoral
objective: 0
x_1_2 = 1
x_2_2 = 1
x_3_2 = 1
x_4_1 = 1
x_5_1 = 1
w_1_1_2 = 1
w_1_2_2 = 1
w_1_3_2 = 1
w_2_1_2 = 1
w_2_2_2 = 1
w_2_3_2 = 1
w_3_1_2 = 1
w_3_2_2 = 1
w_3_3_2 = 1
w_4_4_1 = 1
w_4_5_1 = 1
w_5_4_1 = 1
w_5_5_1 = 1
```

Resolution dans l'optique de faire perdre le parti A

Solution

```
sol = m.solve()
```

```
sol.display()
```

```
solution for: Découpage éléctoral
objective: 0
x_1_2 = 1
x_2_2 = 1
x_3_2 = 1
x_4_1 = 1
x_5_1 = 1
w_1_1_2 = 1
w_1_2_2 = 1
w_1_3_2 = 1
w_2_1_2 = 1
w_2_2_2 = 1
w_2_3_2 = 1
w_3_1_2 = 1
w_3_2_2 = 1
w_3_3_2 = 1
w_4_4_1 = 1
w_4_5_1 = 1
w_5_4_1 = 1
w_5_5_1 = 1
```

Resolution dans l'optique de faire gagner le parti A

```
m.maximize(sum(z))
```

```
solmax = m.solve()
```

```
solmax.display()
```

```
solution for: Découpage éléctoral
objective: 2
x_1_1 = 1
x_2_1 = 1
x_3_1 = 1
x_4_2 = 1
x_5_2 = 1
y_1_1 = 1
y_2_1 = 1
y_3_1 = 1
y_4_2 = 1
y_5_2 = 1
z_1 = 1
z_2 = 1
w_1_1_1 = 1
w_1_2_1 = 1
w_1_3_1 = 1
w_2_1_1 = 1
w_2_2_1 = 1
w_2_3_1 = 1
w_3_1_1 = 1
w_3_2_1 = 1
w_3_3_1 = 1
w_4_4_2 = 1
w_4_5_2 = 1
w_5_4_2 = 1
w_5_5_2 = 1
```

Resolution dans l'optique de faire perdre le parti A

Solution

```
sol = m.solve()
```

```
sol.display()
```

```
solution for: Découpage éléctoral
objective: 0
x_1_2 = 1
x_2_2 = 1
x_3_2 = 1
x_4_1 = 1
x_5_1 = 1
w_1_1_2 = 1
w_1_2_2 = 1
w_1_3_2 = 1
w_2_1_2 = 1
w_2_2_2 = 1
w_2_3_2 = 1
w_3_1_2 = 1
w_3_2_2 = 1
w_3_3_2 = 1
w_4_4_1 = 1
w_4_5_1 = 1
w_5_4_1 = 1
w_5_5_1 = 1
```

Resolution dans l'optique de faire gagner le parti A

```
m.maximize(sum(z))
```

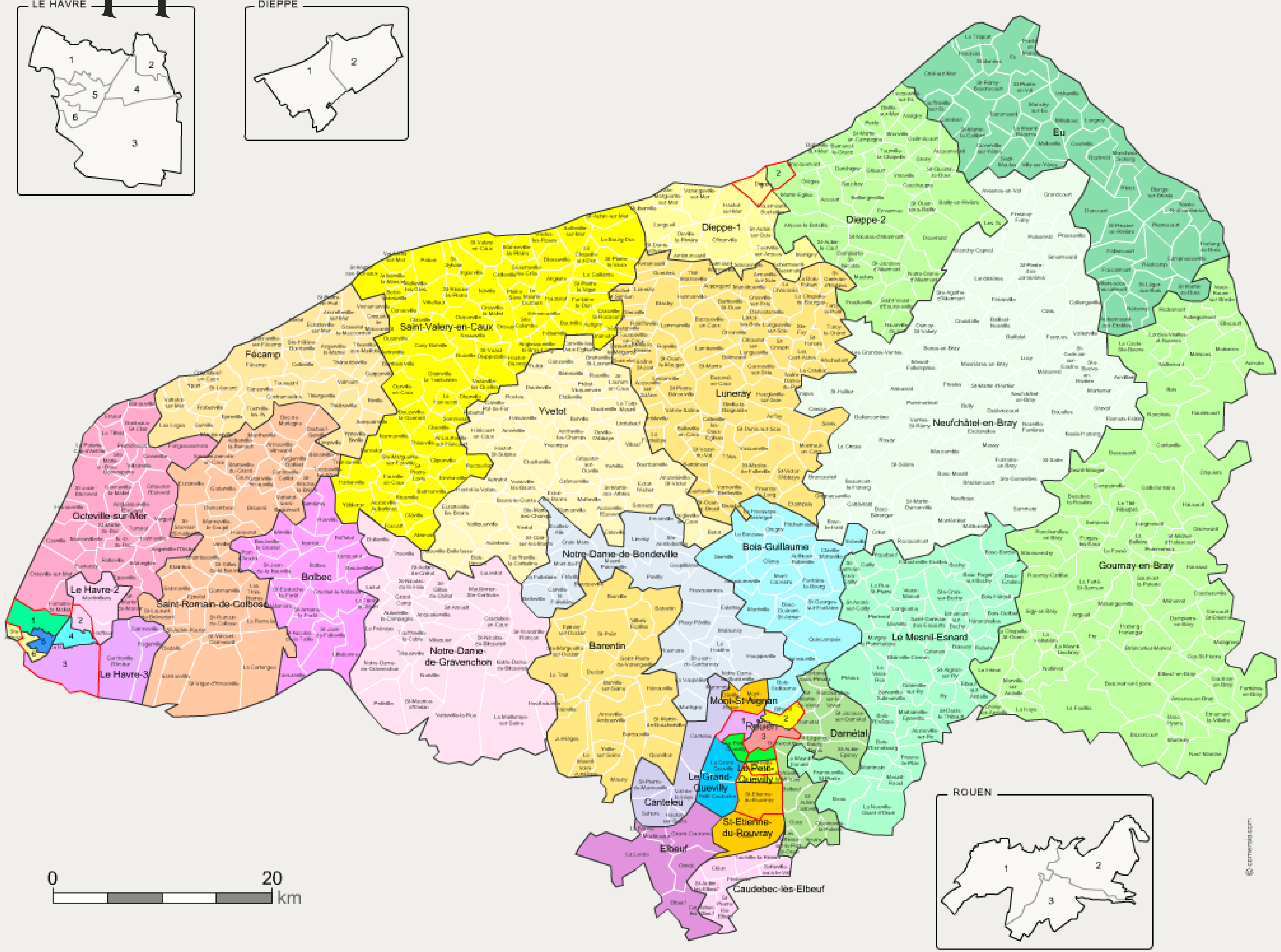
```
solmax = m.solve()
```

```
solmax.display()
```

```
solution for: Découpage éléctoral
objective: 2
x_1_1 = 1
x_2_1 = 1
x_3_1 = 1
x_4_2 = 1
x_5_2 = 1
y_1_1 = 1
y_2_1 = 1
y_3_1 = 1
y_4_2 = 1
y_5_2 = 1
z_1 = 1
z_2 = 1
w_1_1_1 = 1
w_1_2_1 = 1
w_1_3_1 = 1
w_2_1_1 = 1
w_2_2_1 = 1
w_2_3_1 = 1
w_3_1_1 = 1
w_3_2_1 = 1
w_3_3_1 = 1
w_4_4_2 = 1
w_4_5_2 = 1
w_5_4_2 = 1
w_5_5_2 = 1
```

```
z_1 = 1.0
x_1_1  =  1.0
x_2_1  =  1.0
x_3_1  =  1.0
x_4_1  =  0
x_5_1  =  0
z_2 = 1.0
x_1_2  =  0
x_2_2  =  0
x_3_2  =  0
x_4_2  =  1.0
x_5_2  =  1.0
```

2. Application sur des données réelles



Démonstration

Resolution dans l'optique de faire perdre le parti A

Solution

```
sol = m.solve()
```

```
sol.display()
```

```
solution for: Découpage éléctoral
objective: 0
x_1_1 = 1
x_2_1 = 1
x_3_1 = 1
x_4_1 = 1
x_5_1 = 1
x_6_1 = 1
x_7_1 = 1
x_8_1 = 1
x_9_1 = 1
x_10_1 = 1
x_11_1 = 1
x_12_1 = 1
x_13_1 = 1
x_14_1 = 1
x_15_1 = 1
x_16_1 = 1
x_17_1 = 1
x_18_1 = 1
x_19_1 = 1
x_20_1 = 1
x_21_1 = 1
x_22_1 = 1
x_23_1 = 1
x_24_1 = 1
x_25_1 = 1
x_26_1 = 1
x_27_1 = 1
x_28_1 = 1
x_29_1 = 1
x_30_1 = 1
x_31_1 = 1
x_32_1 = 1
x_33_1 = 1
x_34_1 = 1
```

Resolution dans l'optique de faire perdre le parti A

Solution

```
sol = m.solve()
```

```
sol.display()
```

```
solution for: Découpage éléctoral
objective: 0
x_1_1 = 1
x_2_1 = 1
x_3_1 = 1
x_4_1 = 1
x_5_1 = 1
x_6_1 = 1
x_7_1 = 1
x_8_1 = 1
x_9_1 = 1
x_10_1 = 1
x_11_1 = 1
x_12_1 = 1
x_13_1 = 1
x_14_1 = 1
x_15_1 = 1
x_16_1 = 1
x_17_1 = 1
x_18_1 = 1
x_19_1 = 1
x_20_1 = 1
x_21_1 = 1
x_22_1 = 1
x_23_1 = 1
x_24_1 = 1
x_25_1 = 1
x_26_1 = 1
x_27_1 = 1
x_28_1 = 1
x_29_1 = 1
x_30_1 = 1
x_31_1 = 1
x_32_1 = 1
x_33_1 = 1
x_34_1 = 1
```

Resolution dans l'optique de faire gagner le parti A

```
m.maximize(sum(z))
```

```
solmax = m.solve()
```

```
solmax.display()
```

```
solution for: Découpage éléctoral
objective: 2
x_1_3 = 1
x_2_1 = 1
x_3_1 = 1
x_4_3 = 1
x_5_3 = 1
x_6_1 = 1
x_7_3 = 1
x_8_1 = 1
x_9_1 = 1
x_10_3 = 1
x_11_3 = 1
x_12_2 = 1
x_13_3 = 1
x_14_3 = 1
x_15_3 = 1
x_16_3 = 1
x_17_3 = 1
x_18_3 = 1
x_19_1 = 1
x_20_3 = 1
x_21_3 = 1
x_22_1 = 1
x_23_1 = 1
x_24_1 = 1
x_25_3 = 1
x_26_1 = 1
x_27_1 = 1
x_28_3 = 1
x_29_1 = 1
x_30_1 = 1
x_31_1 = 1
x_32_3 = 1
x_33_1 = 1
x_34_3 = 1
```

Resolution dans l'optique de faire perdre le parti A

Solution

```
sol = m.solve()
```

```
sol.display()
```

```
solution for: Découpage éléctoral
objective: 0
x_1_1 = 1
x_2_1 = 1
x_3_1 = 1
x_4_1 = 1
x_5_1 = 1
x_6_1 = 1
x_7_1 = 1
x_8_1 = 1
x_9_1 = 1
x_10_1 = 1
x_11_1 = 1
x_12_1 = 1
x_13_1 = 1
x_14_1 = 1
x_15_1 = 1
x_16_1 = 1
x_17_1 = 1
x_18_1 = 1
x_19_1 = 1
x_20_1 = 1
x_21_1 = 1
x_22_1 = 1
x_23_1 = 1
x_24_1 = 1
x_25_1 = 1
x_26_1 = 1
x_27_1 = 1
x_28_1 = 1
x_29_1 = 1
x_30_1 = 1
x_31_1 = 1
x_32_1 = 1
x_33_1 = 1
x_34_1 = 1
```

Resolution dans l'optique de faire gagner le parti A

```
m.maximize(sum(z))
```

```
solmax = m.solve()
```

```
solmax.display()
```

```
solution for: Découpage éléctoral
objective: 2
x_1_3 = 1
x_2_1 = 1
x_3_1 = 1
x_4_3 = 1
x_5_3 = 1
x_6_1 = 1
x_7_3 = 1
x_8_1 = 1
x_9_1 = 1
x_10_3 = 1
x_11_3 = 1
x_12_2 = 1
x_13_3 = 1
x_14_3 = 1
x_15_3 = 1
x_16_3 = 1
x_17_3 = 1
x_18_3 = 1
x_19_1 = 1
x_20_3 = 1
x_21_3 = 1
x_22_1 = 1
x_23_1 = 1
x_24_1 = 1
x_25_3 = 1
x_26_1 = 1
x_27_1 = 1
x_28_3 = 1
x_29_1 = 1
x_30_1 = 1
x_31_1 = 1
x_32_3 = 1
x_33_1 = 1
x_34_3 = 1
```

```
1.0
x _ 1 _ 1  =  0
x _ 2 _ 1  =  1.0
x _ 3 _ 1  =  1.0
x _ 4 _ 1  =  0
x _ 5 _ 1  =  0
x _ 6 _ 1  =  1.0
x _ 7 _ 1  =  0
x _ 8 _ 1  =  1.0
x _ 9 _ 1  =  1.0
x _ 10 _ 1  =  0
x _ 11 _ 1  =  0
x _ 12 _ 1  =  0
x _ 13 _ 1  =  0
x _ 14 _ 1  =  0
x _ 15 _ 1  =  0
x _ 16 _ 1  =  0
x _ 17 _ 1  =  0
x _ 18 _ 1  =  0
x _ 19 _ 1  =  1.0
x _ 20 _ 1  =  0
x _ 21 _ 1  =  0
x _ 22 _ 1  =  1.0
x _ 23 _ 1  =  1.0
x _ 24 _ 1  =  1.0
x _ 25 _ 1  =  0
x _ 26 _ 1  =  1.0
x _ 27 _ 1  =  1.0
x _ 28 _ 1  =  0
x _ 29 _ 1  =  1.0
x _ 30 _ 1  =  1.0
x _ 31 _ 1  =  1.0
x _ 32 _ 1  =  0
x _ 33 _ 1  =  1.0
x _ 34 _ 1  =  0
x _ 35 _ 1  =  0
x _ 36 _ 1  =  1.0
x _ 37 _ 1  =  0
x _ 38 _ 1  =  1.0
x _ 39 _ 1  =  0
x _ 40 _ 1  =  0
x _ 41 _ 1  =  1.0
```

III.Conclusion