Rapport de projet

MU4IN200 - Modélisation, optimisation, graphes, programmation linéaire

Optimisation appliquée à la localisation d'unités de soin et à la prise en charge des patients

Master d'informatique spécialité androide

Première année

Année universitaire 2020 - 2021

Encadrant : Patrice Perny
Groupe : Zitong Yang
Vincent Fu

Table des matières

.1 Répartition de patients dans les unités de soin

Question 1.1

En utilisant les notations et les variables de l'énoncé, le programme linéaire qui détermine les secteurs de service des k unités de soin de manière à minimiser la distance moyenne de chaque habitant à l'unité de soin dont il dépend est :

$$(P_1)\left\{\begin{array}{ll} \min & z = \sum_{j\in J}\sum_{i\in I} d_{ij}x_{ij} \\ s.c. & \left|\begin{array}{lll} \forall\, i \in I, & \sum_{j\in J} x_{ij} & \geq 1 \\ \forall\, j \in J, & \sum_{i\in I} v_i x_{ij} & \leq \dfrac{1+\alpha}{k}\sum_{i\in I} v_i \\ \forall\, i \in I, \forall\, j \in J, & x_{ij} \in \{0,1\} & \end{array}\right. \end{array}\right.$$

La première contrainte correspond à la nécessité que chaque ville soit associée à un unique secteur. Comme on est dans le cas d'une minimisation, ces contraintes seront nécessairement saturées (que l'on met un égal ou un supérieur ou égal revient au même).
La deuxième contrainte correspond à l'équilibrage des charges des secteurs.

En outre, il n'est pas possible de traiter le problème par un algorithme de flot max coût min car il n'est pas possible d'exprimer la deuxième contrainte dans l'algorithme.

Question 1.2

On pose les matrices du problème sous les formes suivantes (pour la lisiblité du code) :

$$x = \begin{pmatrix} x_{1j_1} \\ x_{2j_1} \\ \vdots \\ x_{1j_2} \\ x_{2j_2} \\ \vdots \\ x_{n-1j_k} \\ x_{nj_k} \end{pmatrix},\ c = \begin{pmatrix} d_{1j_1} \\ d_{2j_1} \\ \vdots \\ d_{1j_2} \\ d_{2j_2} \\ \vdots \\ d_{n-1j_k} \\ d_{nj_k} \end{pmatrix},\ A = \begin{pmatrix} a_{1i_{11}} & a_{1i_{21}} & \cdots & a_{1i_{12}} & a_{1i_{22}} & \cdots & a_{1i_{n-1k}} & a_{1i_{nk}} \\ a_{2i_{11}} & a_{2i_{21}} & \cdots & a_{2i_{12}} & a_{2i_{22}} & \cdots & a_{2i_{n-1k}} & a_{2i_{nk}} \\ \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ a_{n+ki_{11}} & a_{n+ki_{21}} & \cdots & a_{n+ki_{12}} & a_{n+ki_{22}} & \cdots & a_{n+ki_{n-1k}} & a_{n+ki_{nk}} \end{pmatrix},\ b = \begin{pmatrix} b_{i_1} \\ b_{i_2} \\ \vdots \\ b_{i_n} \\ b_{j_1} \\ b_{j_2} \\ \vdots \\ b_{j_n} \end{pmatrix}$$

où $a_{i'i_{pq}}$ correspond au coefficient de x_{pq} de la i' ème contrainte, b_{i_p} correspond au second membre de la p ème contrainte (qui concerne l'association d'une ville à un unique secteur) et b_{j_q} correspond au second membre de la $n+q$ ème contrainte (qui concerne l'équilibrage des charges).

Pour les unités de soin qui sont implantées dans Toulouse, Nice, Nantes et $\alpha = 0,1$, on obtient le dictionnaire et la valeur de la fonction objectif suivants :

```
Toulouse : {Bordeaux, Dijon, Montpellier, Toulouse}
Nice : {Grenoble, Nice, Saint-Étienne, Strasbourg, Toulon}
Nantes : {Angers, Le Havre, Lille, Nantes, Reims, Rennes}
z = 4716
```

Pour les unités de soin qui sont implantées dans Toulouse, Nice, Nantes et $\alpha = 0,2$, on obtient le dictionnaire et la valeur de la fonction objectif suivants :

```
Toulouse : {Bordeaux, Montpellier, Toulouse}
Nice : {Grenoble, Nice, Saint-Étienne, Strasbourg, Toulon}
Nantes : {Angers, Dijon, Le Havre, Lille, Nantes, Reims, Rennes}
```

`z = 4710`

Pour les unités de soin qui sont implantées dans Toulouse, Nice, Nantes, Montpellier et $\alpha = 0,1$, on obtient le dictionnaire et la valeur de la fonction objectif suivants :

```
Toulouse : {Bordeaux, Lille, Toulouse}
Nice : {Nice, Strasbourg, Toulon}
Nantes : {Angers, Le Havre, Nantes, Rennes}
Montpellier : {Dijon, Grenoble, Montpellier, Reims, Saint-Étienne}
z = 4555
```

Pour les unités de soin qui sont implantées dans Toulouse, Nice, Nantes, Montpellier et $\alpha = 0,2$, on obtient le dictionnaire et la valeur de la fonction objectif suivants :

```
Toulouse : {Bordeaux, Lille, Toulouse}
Nice : {Nice, Strasbourg, Toulon}
Nantes : {Angers, Le Havre, Nantes, Reims, Rennes}
Montpellier : {Dijon, Grenoble, Montpellier, Saint-Étienne}
z = 4283
```

On remarque que plus on augmente α, plus la solution optimale devient meilleure. Cela est prévisible car on réduit les contraintes sur les charges (les contraintes sur les charges deviennent moins « lourdes »).
De même qu'en augmentant le nombre d'unités de soin, on obtient une meilleure solution optimale puisqu'on répartit mieux nos secteurs et donc on diminue la valeur de la fonction objectif.

.2 Localisation optimale des unités de soin

Question 2.1

Soient $k \in [\![1, n]\!]$ et $y_j \in \{0, 1\}$ pour les variables de décision des emplacements des k unités (égale 1 si on place l'unité de soin au ville j, 0 sinon) pour $j \in I$. En utilisant les notations et les variables de l'énoncé de la partie 1, le programme linéaire qui détermine les emplacements des k unités de soin en plus de leurs secteurs de manière à minimiser la distance moyenne de chaque habitant à l'unité de soin dont il dépend est :

$$
(P_2)\left\{\begin{array}{l|c}
\min\ z = \sum_{j\in I}\sum_{i\in I} d_{ij}x_{ij} & \\
 & \forall\, i \in I,\ \sum_{j\in I} x_{ij} \geq 1 \\
 & \forall\, j \in I,\ \sum_{i\in I} v_i x_{ij} \leq \dfrac{1+\alpha}{k}\sum_{i\in I} v_i \\
s.c. & \sum_{j\in I} y_j = k \\
 & \forall\, j \in I,\ \sum_{i\in I} x_{ij} \leq y_j n \\
 & \forall\, i \in I, \forall\, j \in I,\ x_{ij} \in \{0,\,1\},\ y_j \in \{0,\,1\}
\end{array}\right.
$$

La troisième contrainte correspond à l'obligation de placer exactement k unités de soin. (Ici, il faut mettre un égal et non un supérieur ou égal car sinon on aurait pu placer n unités de soin et obtenir une valeur de la fonction objectif 0 par exemple.)
La quatrième contrainte correspond à l'affectation des villes aux secteurs admissibles (et à l'affectation d'aucune ville à un secteur non admissible).

Pour $k = 3$ et $\alpha = 0,1$, on obtient comme solution optimale :

```
Montpellier : {Montpellier, Nice, Toulon, Toulouse}
Dijon : {Dijon, Grenoble, Lille, Reims, Saint-Étienne, Strasbourg}
Angers : {Angers, Bordeaux, Le Havre, Nantes, Rennes}
z = 3392
```

Pour $k = 3$ et $\alpha = 0,3$, on obtient comme solution optimale :

```
Nantes : {Angers, Bordeaux, Nantes, Rennes}
Montpellier : {Grenoble, Montpellier, Nice, Toulon, Toulouse}
Reims : {Dijon, Le Havre, Lille, Reims, Saint-Étienne, Strasbourg}
z = 3374
```

Pour $k = 4$ et $\alpha = 0,1$, on obtient comme solution optimale :

```
Nantes : {Angers, Bordeaux, Nantes, Rennes}
Reims : {Le Havre, Lille, Reims, Strasbourg}
Saint-Étienne : {Dijon, Grenoble, Saint-Étienne, Toulouse}
Toulon : {Montpellier, Nice, Toulon}
z = 2756
```

Pour $k = 4$ et $\alpha = 0,3$, on obtient comme solution optimale :

```
Nantes : {Angers, Bordeaux, Nantes, Rennes}
Montpellier : {Grenoble, Montpellier, Saint-Étienne, Toulouse}
Reims : {Dijon, Le Havre, Lille, Reims, Strasbourg}
Toulon : {Nice, Toulon}
z = 2740
```

Pour $k = 5$ et $\alpha = 0,1$, on obtient comme solution optimale :

```
Nantes : {Angers, Nantes, Rennes}
Bordeaux : {Bordeaux, Toulouse}
Reims : {Dijon, Le Havre, Lille, Reims}
Toulon : {Nice, Saint-Étienne, Toulon}
Grenoble : {Grenoble, Montpellier, Strasbourg}
z = 2658
```

Pour $k = 5$ et $\alpha = 0,2$, on obtient comme solution optimale :

```
Bordeaux : {Bordeaux, Toulouse}
Reims : {Lille, Reims, Strasbourg}
Saint-Étienne : {Dijon, Grenoble, Saint-Étienne}
Toulon : {Montpellier, Nice, Toulon}
Angers : {Angers, Le Havre, Nantes, Rennes}
z = 2088
```

En tenant compte uniquement de $\alpha = 0,1$, on en déduit que les localisations optimales sont Montpellier, Dijon et Angers pour 3 unités, Nantes, Reims, Saint-Étienne et Toulon pour 4 unités et Nantes, Bordeaux, Reims, Toulon et Grenoble pour 5 unités.
En augmentant α, on obtient évidemment une meilleure solution qui se traduit par un changement d'implantation d'unité d'une ou deux villes pour une petite fluctuation de α. De plus, plus k augmente, plus l'écart entre les valeurs des fonctions objectifs devient grand pour une petite fluctuation de α (c'est le cas pour $k = 5$ par exemple).
Enfin, en comparant aux solutions du 1.2, on remarque qu'on a une meilleure solution puisque la valeur de la fonction objectif 2.1 est inférieure à celle du 1.2.

Question 2.2

En tenant compte de l'équité d'accès au soin (distance de chaque individu à son unité de soin), et en modifiant le programme linéaire 2.1, on obtient comme programme linéaire :

$$(P_2') \begin{cases} \min \; z = r \\ s.c. \left| \begin{array}{c} \forall\, i \in I,\ \sum_{j \in I} x_{ij} \geq 1 \\ \forall\, j \in I,\ \sum_{i \in I} v_i x_{ij} \leq \dfrac{1+\alpha}{k} \sum_{i \in I} v_i \\ \sum_{j \in I} y_j = k \\ \forall\, j \in I,\ \sum_{i \in I} x_{ij} \leq y_j n \\ \forall\, i \in I,\ r \geq \sum_{j \in I} d_{ij} x_{ij} \\ \forall\, i \in I, \forall\, j \in I,\ x_{ij} \in \{0,\, 1\},\ y_j \in \{0,\, 1\} \\ r \geq 0 \end{array} \right. \end{cases}$$

La cinquième contrainte correspond à l'équité d'accès au soin car si $x_{ij} = 1$, alors $d(i, f(i)) = d_{ij}$. Or, comme chaque ville ne peut qu'être affectée à un unique secteur, nécessairement $\forall\, p \neq j,\ x_{ip} = 0$ et donc $d(i, f(i)) = d_{ij} = \sum_{p \in I} d_{ip} x_{ip}$.

Pour $k = 3$ et $\alpha = 0.1$, on obtient comme solution optimale :
```
Montpellier : {Grenoble, Montpellier, Nice, Saint-Étienne, Toulon}
Bordeaux : {Bordeaux, Nantes, Rennes, Toulouse}
Reims : {Angers, Dijon, Le Havre, Lille, Reims, Strasbourg}
r = 458
```
Contre 501 (pour Dijon-Lille) pour le programme linéaire précédent.

Pour $k = 3$ et $\alpha = 0.4$, on obtient comme solution optimale :
```
Nantes : {Angers, Bordeaux, Nantes, Rennes}
Montpellier : {Grenoble, Montpellier, Nice, Saint-Étienne, Toulon, Toulouse}
Reims : {Dijon, Le Havre, Lille, Reims, Strasbourg}
r = 347
```
Contre 546 (pour Reims-Saint-Étienne) pour le programme linéaire précédent.

Pour $k = 4$ et $\alpha = 0.1$, on obtient comme solution optimale :
```
Nantes : {Angers, Bordeaux, Nantes, Rennes}
Montpellier : {Grenoble, Nice, Toulouse}
Reims : {Le Havre, Lille, Reims, Strasbourg}
Grenoble : {Dijon, Montpellier, Saint-Étienne, Toulon}
r = 372
```
Contre 536 (pour Saint-Étienne-Toulouse) pour le programme linéaire précédent.

Pour $k = 4$ et $\alpha = 0.2$, on obtient comme solution optimale :
```
Nantes : {Angers, Bordeaux, Nantes, Rennes}
Montpellier : {Grenoble, Saint-Étienne, Toulouse}
Reims : {Dijon, Le Havre, Lille, Reims, Strasbourg}
Toulon : {Montpellier, Nice, Toulon}
r = 347
```
Contre 347 (pour Nantes-Bordeaux) pour le programme linéaire précédent.

Pour $k = 5$ et $\alpha = 0.1$, on obtient comme solution optimale :
```
Toulouse : {Montpellier, Toulouse}
Montpellier : {Nice, Saint-Étienne, Toulon}
Bordeaux : {Bordeaux, Nantes}
Le Havre : {Angers, Le Havre, Lille, Rennes}
Dijon : {Dijon, Grenoble, Reims, Strasbourg}
r = 347
```
Contre 533 (pour Grenoble-Strasbourg) pour le programme linéaire précédent.

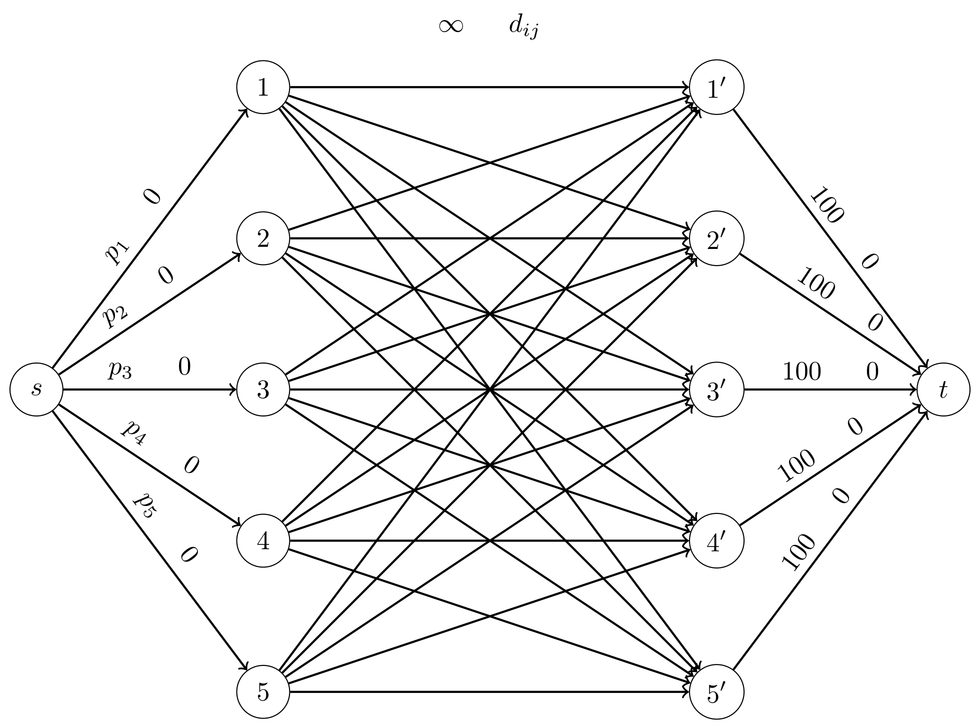
Pour $k = 5$ et $\alpha = 0.2$, on obtient comme solution optimale :
```
Toulouse : {Bordeaux, Toulouse}
Nice : {Montpellier, Nice, Toulon}
Reims : {Dijon, Lille, Reims}
Dijon : {Grenoble, Reims, Saint-Étienne, Strasbourg}
Angers : {Angers, Le Havre, Nantes, Rennes}
r = 332
```
Contre 347 (pour Reims-Strasbourg) pour le programme linéaire précédent.

En comparant les distances maximums d'un individu à son unité, on observe qu'on a une meilleure solution optimale contre le programme linaire précédent.
En outre, on remarque aussi que certaines villes qui ont des unités de soin ne sont pas forcément affectées à eux-mêmes (par exemple pour $k = 4$ et $\alpha = 0.1$, on a Montpellier qui est affecté à l'unité de Grenoble alors que Montpellier possède lui-même une unité de soin). Ces résultats ne sont pas si étonnants puisqu'on cherche uniquement à minimiser la distance maximale. On aurait donc pu encore mieux optimiser en affectant les villes ayant une unité de soin à eux-mêmes par l'ajout de contraintes supplémentaires.

.3 Equilibrage des charges des unités de soin

Question 3.1

Soit p_i le nombre de patients du secteur i pour $i \in [\![1, 5]\!]$. Le problème d'équilibrage des charges des unités de soin peut se traduire par un problème de flot maximum à coût minimum sous le graphe suivant :



où les valeurs à gauche au-dessus des arcs correspondent aux capacités et les valeurs à droite au-dessus des arcs correspondent aux coûts.

Question 3.2

Pour les villes de Toulouse, Nice, Reims, Dijon et Angers avec respectivement 250, 200, 0, 20 et 0 patients, on obtient comme résultat étape par étape :

— 100 patients de Toulouse restent à Toulouse.
— 20 patients de Dijon restent à Dijon.
— 100 patients de Nice restent à Nice.
— 100 autres patients de Toulouse sont affectés à Angers.
— 50 autres patients de Toulouse sont affectés à Dijon. (- 50 patients par la suite à cause de (*))
— 30 autres patients de Nice sont affectés à Dijon.
— 50 autres patients de Nice sont affectés à Reims. (chemin obtenu : Nice, Dijon, Toulouse, Reims) (*)
— 20 autres patients de Nice sont affectés à Reims.

Pour un coût total de déplacement non groupé (chaque déplacement ne prend qu'en charge un seul patient) de `z = 176000` d'affectation optimale :

— 100 patients de Toulouse restent à Toulouse.
— 20 patients de Dijon restent à Dijon.
— 100 patients de Nice restent à Nice.
— 100 autres patients de Toulouse sont affectés à Angers.
— 30 autres patients de Nice sont affectés à Dijon.
— 70 autres patients de Nice sont affectés à Reims.