M1 ANDROIDE 2020/2021

Rapport de projet MOGPL

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

Smail Zidelmal - Maria Hattab
G1

1 - Répartition de patients dans les unités de soin

1.1) un programme linéaire (éventuellement en variables Booléenes) 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 :

Programme linéaire :

la fonction objectif :

$$min\ (\sum_{i=1}^{n} \sum_{j=1}^{k} d_{ij} x_{ij} v_i)$$

les contraintes :

$$\sum_{i=1}^{n} x_{ij}\, v_i \leq (1+\alpha)/k \sum_{i=1}^{n} vi \qquad , \forall\ j \in [1,k]$$

$$\sum_{j=1}^{k} x_{ij} = 1 \qquad , \forall\ i \in [1,n]$$ (un seul secteur j pour une ville i)

$$x_{ij} \in \{0,1\}$$

Explication :

Les x_{ij} sont les variables de décision avec i ∈ [1,n] et j ∈ [1,k].

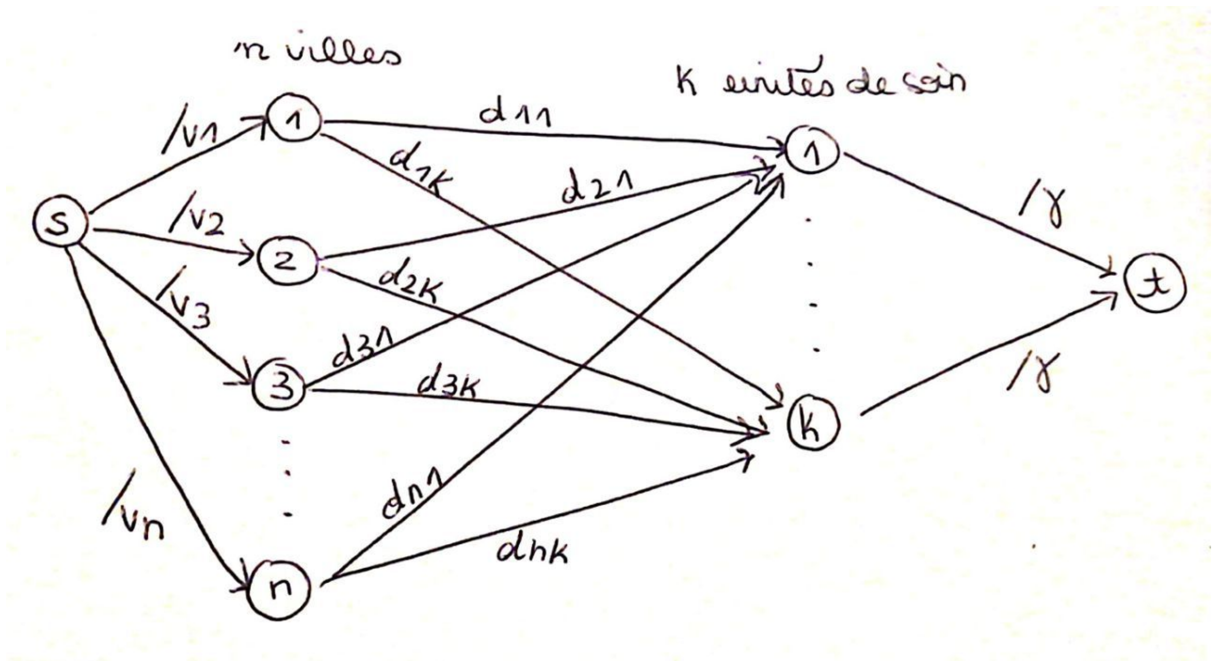
n : nombre de villes, k : nombre de secteurs.

$x_{ij} = 1\ si\ i\ \varepsilon\ J$ (la ville i est dans le secteur j)

$x_{ij} = 0\ sinon$

1.1)b) On ne peut pas traiter le même problème par un algorithme de flot maximum à coût minimum.

En effet, en modélisant le problème dans un réseau de transport, on aurait les capacités vi de la source s aux n villes, les capacités dij des n villes aux k unités de soin et des capacités gamma des k unités de soin au puits t.



Ce qui pose problème est le fait qu'on ne puisse pas contrôler le fait que tous les habitants d'une même ville aillent dans un même secteur.

1.2)programme linéaire et comparaison de la qualité des solutions obtenues:

- Les 3 villes choisies arbitrairement : Toulouse, Nice, Nantes (0,1,2)

Résultat pour alpha = 0.1	Résultat pour alpha = 0.2
---- secteur Toulouse : Toulouse Montpellier Bordeaux Dijon ---- secteur Nice : Nice Strasbourg Saint-Étienne Toulon Grenoble ---- secteur Nantes : Nantes Lille Rennes Reims Le Havre Angers Valeur de la fonction objectif : 273.0120777107589	---- secteur Toulouse : Toulouse Montpellier Bordeaux ---- secteur Nice : Nice Strasbourg Saint-Étienne Toulon Grenoble ---- secteur Nantes : Nantes Lille Rennes Reims Le Havre Dijon Angers Valeur de la fonction objectif : 272.74793968298496

- Les 4 villes choisies arbitrairement : Toulouse, Bordeaux, Nantes, Strasbourg (0, 5, 2, 4)

Résultat pour alpha = 0.1	Résultat pour alpha = 0.2
---- secteur Toulouse : Toulouse, Montpellier, Toulon ---- secteur Bordeaux: Nice, Bordeaux, Saint-Étienne, Grenoble ---- secteur Nantes : Nantes, Rennes, Angers, Le Havre ---- secteur Strasbourg : Strasbourg, Lille, Dijon, Reims Valeur de la fonction objectif : 270.1569678776827	---- secteur Toulouse : Toulouse, Nice, Toulon ---- secteur Bordeaux: Montpellier, Bordeaux, Saint-Étienne ---- secteur Nantes : Nantes, Rennes, Le Havre, Angers ---- secteur Strasbourg : Strasbourg, Lille, Reims, Grenoble, Dijon Valeur de la fonction objectif : 260.98653976714826

- Comparaison de la qualité des solutions obtenues :

  On peut remarquer qu'en faisant augmenter alpha, on obtient potentiellement une meilleure solution qu'avec un alpha plus petit. Cela s'explique par le fait qu'augmenter alpha signifie augmenter la capacité d'un secteur (c'est-à-dire gamma : la population totale des villes composant un secteur) ce qui rend des solutions jusque-là non réalisables pour un alpha plus petit réalisables.

  Pour un même alpha, en faisant augmenter le nombre de secteurs, on obtient également une meilleure solution puisqu'en rajoutant un secteur en plus, on obtient potentiellement pour une ville donnée, une distance de cette ville à ce nouveau secteur plus petite (et en particulier lorsqu'une ville est dans le secteur d'elle-même). Cela explique donc la différence de solution optimale en faisant varier k.

2 - Localisation optimale des unités de soin

2.1) Reprendre le problème de la section précédente dans le cas de ce scénario plus général pour déterminer la localisation optimale de k unités de soin et les secteurs de service associés :

Programme linéaire :

la fonction objectif :

$$min\ (\sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} x_{ij} v_i)$$

les contraintes :

$\sum_{i=1}^{n} x_{ij}\, v_i < \gamma$, ∀ j ∈ {1,n} --> $\gamma = (1+\alpha)/k \sum_{i=1}^{n} vi$

$\sum_{j=1}^{n} x_{ij} = 1$, ∀ i ∈ {1,n}

$\sum_{i=1}^{n} y_i = k$, ∀ i ∈ {1,n} (pour limiter les unités de soin à k)

$x_{ij} - y_j \leq 0$, ∀ i ∈ {1,n} , ∀ j ∈ {1,n} (s'il n'y a pas d'unité dans cette ville on ne peut pas lui affecter une ville)

On a désormais n*n variables x_{ij} et n variables y_i.

Afin de trouver la localisation optimale des secteurs pour un k donné, on fait augmenter alpha jusqu'à obtenir une valeur constante de la valeur de la fonction objectif.

- Pour k = 3 :

Résultat optimal obtenu avec alpha = 0.4
--- secteur Montpellier : Toulouse Nice Montpellier Saint-Étienne Toulon Grenoble --- secteur Nantes : Nantes Bordeaux Rennes Angers --- secteur Reims : Strasbourg Lille

Reims Le Havre Dijon Valeur de la fonction objectif : 208.62693365128348

Localisation optimale des unités : Nantes, Montpellier, Reims.

- Pour k = 4 :

Résultat optimal obtenu avec alpha = 0.3
--- secteur Toulouse : Toulouse Bordeaux -- secteur Nantes : Rennes Nantes Angers -- secteur Reims Strasbourg Lille Reims Le Havre Dijon -- secteur Toulon : Nice Montpellier Saint-Étienne Toulon Grenoble Valeur de la fonction objectif : 166.3152004488708

Localisation optimale des unités : Toulouse, Nantes, Reims, Toulon.

- Pour k = 5 :

Résultat optimal obtenu avec alpha = 0.2
--- secteur Toulouse : Toulouse Bordeaux

```
--- secteur Nice :
Nice
Toulon

--- secteur Rennes :
Nantes
Rennes
Le Havre
Angers

--- secteur Montpellier :
Montpellier
Saint-Étienne
Grenoble

--- secteur Reims :
Strasbourg
Lille
Reims

Valeur de la fonction objectif : 134.99248197503155
```

Localisation optimale des unités : Toulouse, Nice, Rennes, Montpellier, Reims.

- Comparaison de la qualité des solutions obtenues :

  Comme remarqué précédemment, en faisant augmenter le nombre de secteurs k, on trouve une meilleure solution puiqu'on se rapproche de la solution où chaque ville est son propre secteur.

- Comparaison des solutions à celles obtenues en première partie :

  Par exemple, pour k = 3 et alpha = 0,1, on trouve bien une meilleure solution qu'en première partie.
  En effet, on a 228.0857769673166 < 273.0120777107589 avec une localisation différente des secteurs. On trouve une meilleure solution puisqu'on cherche la meilleure localisation des secteurs possible alors qu'en première partie on choisissait manuellement la localisation des secteurs.

2.2) Ecrire un programme linéaire en variables mixtes qui reprend la question 2.1 avec ce nouvel objectif :

la fonction objectif :

$$min\ \ Z$$

les contraintes :

$Z \geq \sum_{j=1}^{n} d_{ij}x_{ij}$, $\forall\ i \in [1, n]$ —> *autrement* $\sum_{j=1}^{n} d_{ij}x_{ij} - z \leq 0$

$\sum_{i=1}^{n} x_{ij}\ v_i < \gamma$, $\forall$ j $\in$ {1,n} --> $\gamma = (1+\alpha)/k \sum_{i=1}^{n} vi$

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

$\sum_{i=1}^{n} y_i = k$, $\forall$ i $\in$ {1,n} --> (pour limiter les unités de soin à k)

$x_{ij} - y_j \leq 0$, $\forall$ i $\in$ {1,n} , $\forall$ j $\in$ {1,n} --> (s’il n’y a pas d’unité dans cette ville on

ne peut pas lui affecter une ville)

- Pour k = 3 :

Résultat optimal obtenu avec alpha = 0.4
--- secteur Montpellier : Toulouse Nice Montpellier Saint-Étienne Toulon Grenoble --- secteur Nantes : Nantes Bordeaux Rennes Angers --- secteur Reims : Strasbourg Lille Reims Le Havre Dijon Valeur de la fonction objectif : 347.0

Localisation optimale des unités : Nantes, Montpellier, Reims.

- Pour k = 4 :

Résultat optimal obtenu avec alpha = 0.2

```
--- secteur Bordeaux :
Toulouse
Nantes
Bordeaux

--- secteur Montpellier :
Nice
Montpellier
Toulon
Grenoble

--- secteur Dijon :
Strasbourg
Saint-Étienne
Dijon

--- secteur Le Havre :
Lille
Rennes
Reims
Le Havre
Angers

Valeur de la fonction objectif : 347.0
```

Localisation optimale des unités : Montpellier, Bordeaux, Le Havre, Dijon.

- Pour k = 5 :

Résultat optimal obtenu avec alpha = 0.5
--- secteur Toulouse : Toulouse Montpellier Bordeaux --- secteur Montpellier : Nice Saint-Étienne Toulon Grenoble --- secteur Strasbourg : Strasbourg --- secteur Rennes :

Nantes
Rennes
Le Havre
Angers

--- secteur Reims :
Reims
Lille
Dijon

Valeur de la fonction objectif : 326.0

Localisation optimale des unités : Toulouse, Montpellier, Strasbourg, Rennes, Reims.

- Comparaison de la qualité des solutions obtenues à celle de la question précédente en terme de distance maximum d'un individu à son unité de soin :

  On peut remarquer que la valeur des solutions optimales a augmenté puisqu'on prend en compte la distance de chaque individu à un secteur donc on a de moins bonnes solutions à l'échelle d'une ville.

3 - Equilibrage des charges des unités de soin

3.1) Montrer que ce problème peut être formulé comme un problème de flot maximum à coût minimum dans un graphe que l'on représentera, ou comme un problème de transport:

On cherche à minimiser le coût des déplacements, ici on a k = 5 secteurs.

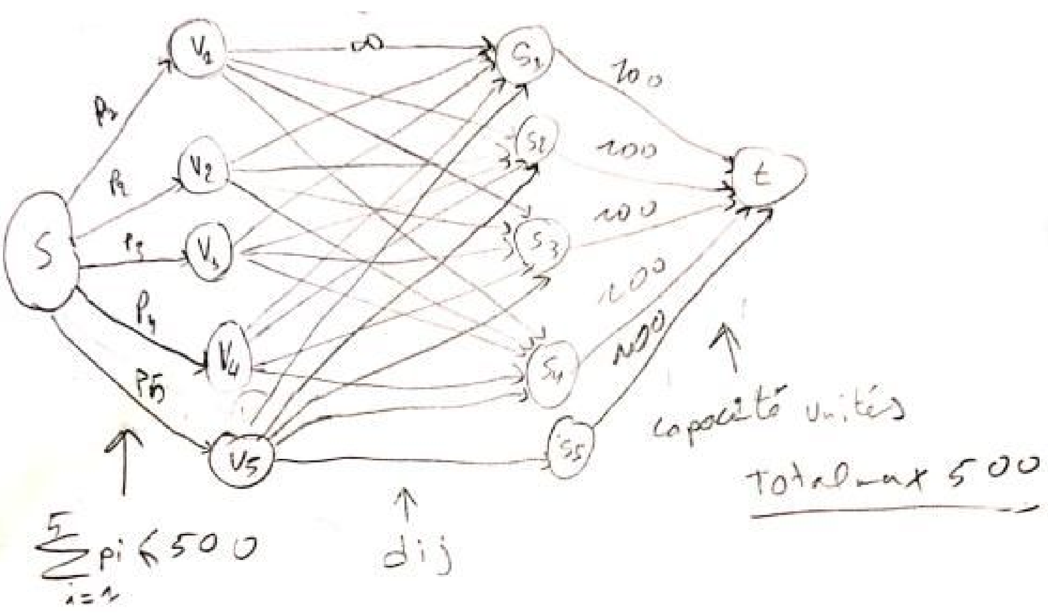
Les contraintes à rajouter par rapport au programme linéaire précédent :

$\sum_{j=1}^{k} p_j \leq 500$

$p_j \leq 100$ $\forall$ j $\in$ [1,k] (pour chaque unité de soin, on accueille au maximum 100 patients)

On peut désormais modéliser le problème comme un problème de flot maximum à coût minimum puisque les patients d'une ville ne se limitent plus à un seul secteur.

Formulation du problème comme un problème de flot maximum à coût minimum :



Par exemple

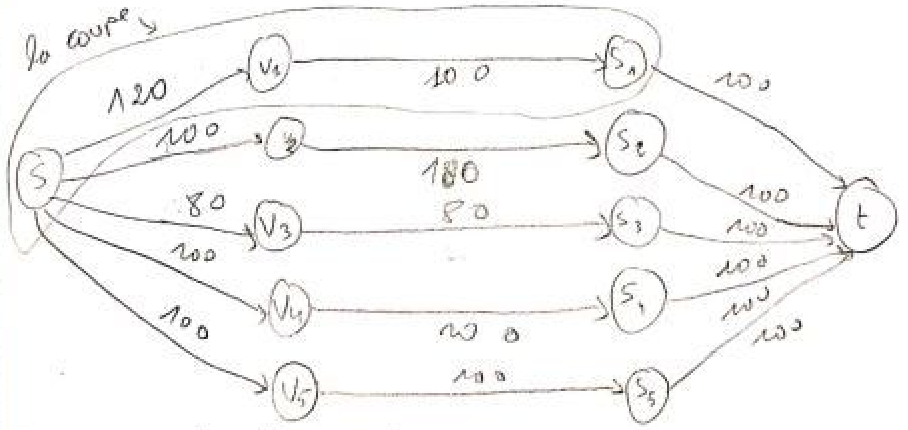
		100	100	100	100	100
	/	S_1	S_2	S_3	S_4	S_5
120:	P_1	d_{11}	d_{12}	d_{13}	d_{14}	d_{15}
80:	P_2	d_{21}	d_{22}	d_{23}	d_{24}	d_{25}
90	P_3	d_{31}	d_{32}	d_{33}	d_{34}	d_{35}
110:	P_4	d_{41}	d_{42}	d_{43}	d_{44}	d_{45}
100:	P_5	d_{51}	d_{52}	d_{53}	d_{54}	d_{55}

3.2) Résoudre ce problème par la méthode de votre choix pour différentes instances de p, certaines ayant au moins une composante supérieure à 100 :

On peut utiliser l'algorithme hongrois généralisé pour résoudre ce problème sous la forme de problème de flot maximum à coût minimum.

Par exemple, on part des secteurs Toulouse Nantes Montpellier Havre Dijon :

	100	100	100	100	100
P / S	Toulouse	Nantes	Montpellier	Havre	Dijon
Toulouse : 120	0	585	242	848	644
Nantes : 100	585	0	824	384	638
Montpellier 80	242	824	0	917	492
Havre : 100	848	384	917	0	506
Dijon : 100	644	638	492	506	0



il faut une autre itération donc :
faire −242 sur la ligne Toulouse / faire +242 sur la colonne Toulouse.

P / S	Toulouse	Nantes	Montpellier	Havre	Dijon
Toulouse	0	−242	0	−242	−242
Nantes	+242	0	/	/	/
Montpellier	+242	/	0	/	/
Havre	+242	/	/	0	/
Dijon	+242	/	/	/	0

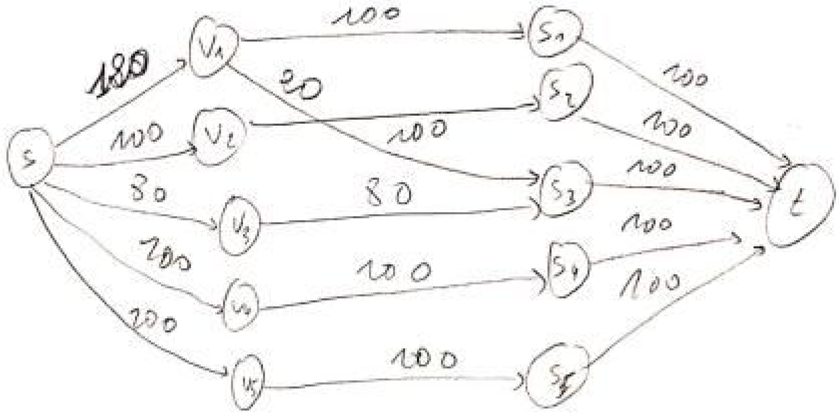


tableau avec nombre de patient. (le resultat).

P / S	toulouse	Nantes	Montpellier	Havre	Dijon
toulouse	100	0	20	0	0
Nantes	0	100	0	0	0
Montpellier	0	0	100	0	0
Havre	0	0	0	100	0
Dijon	0	0	0	0	100

$100 \times 0 + 20 \times 242 + 100 \times 0 + 100 \times 0 + 100 \times 0$

$= 4840$.

acheminement des ressources en satisfaisant les demandes et en minimisant le coût total d'acheminement.

-- Merci --