Fascicule 1
Des problèmes d'optimisation de référence

Xavier Gandibleux

7 septembre 2024

Contenu du fascicule

1 Affectation linéaire, LAP

1.1 Introduction

Le *problème d'affectation* considère une liste finie de tâches et de ressources pour traiter ces dernières. Affecter une ressource à une tâche engendre un coût, et l'objectif consiste à trouver une solution qui engendre le coût minimal (ou maximal). Dans la version de base, on considère qu'il y a exactement autant de ressources qu'il y a de tâches, et que la fonction à optimiser est linéaire. On parle dans ce cas de *problème d'affectation linéaire* (Linear Assignment Problem, LAP).

Plus formellement, soit n tâches à affecter à n ressources. Une tâche ne peut être affectée qu'à une et une seule ressource et réciproquement. Pour l'ensemble des couples $i-j$, le coût de l'affectation noté c_{ij} avec $i, j = 1, \ldots, n$ est connu. Le LAP consiste à déterminer l'affectation de coût minimum (ou maximum).

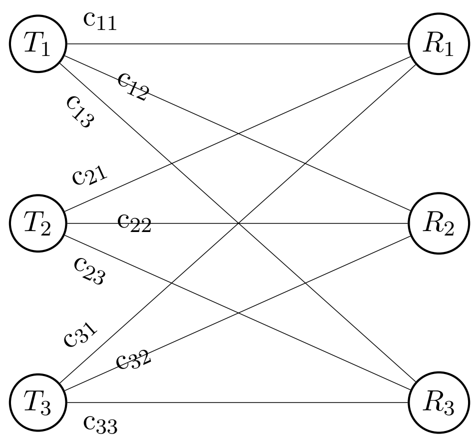
1.2 Modèle

Soit la variable x_{ij} qui vaut 1 si on affecte la tâche i à la ressource j, 0 sinon. Une formulation du LAP sous forme de problème linéaire en variables 0-1 est donnée par

$$\left[\begin{array}{lll} \min z(x) = & \displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij} & \\ s/c & \displaystyle\sum_{i=1}^{n} x_{ij} = 1 & j = 1, \ldots, n \\ & \displaystyle\sum_{j=1}^{n} x_{ij} = 1 & i = 1, \ldots, n \\ & x_{ij} \in \{0, 1\} & i = 1, \ldots, n \\ & & j = 1, \ldots, n \end{array}\right] \qquad \text{(LAP)}$$

1.3 Exemple

Soit 3 tâches T_1, T_2, T_3, 3 ressources R_1, R_2, R_3 et les coût d'affectation c_{ij} ($i = 1, \ldots, 3; j = 1, \ldots, 3$). Un tel problème peut se représenter par un graphe biparti :



(LAP)

1.4 Commentaire

Par la propriété de totale unimodularité de la matrice des contraintes, le problème LAP est connu pour être facile à résoudre.

<u>Définition</u> (matrice unimodulaire) : une matrice carrée d'entiers A est unimodulaire si son déterminant vaut +1, 0 ou -1.

<u>Définition</u> (matrice totalement unimodulaire) : on dit que une matrice A est totalement unimodulaire si toute sous-matrice carrée a pour déterminant -1, 0 ou +1.

<u>Théorème</u> : soit un programme linéaire (PL) $\{\min cx \text{ avec } Ax \geq d,\ x \geq 0\}$; si A est totalement unimodulaire et d entier, alors la solution optimale du PL relaxé est entière pour tout vecteur c.

Pour le LAP, le problème est linéaire, la matrice des contraintes est totalement unimodulaire et le second membre est entier. La solution optimale du problème peut donc être obtenue, par exemple, à l'aide de l'algorithme du simplexe en relâchant la contrainte d'intégrité sur l'intervalle $0 \leq x_{ij} \leq 1,\ i,j = 1,\ldots,n$. Pour des questions d'efficacité, un algorithme adhoc sera préféré, tel que l'algorithme dit "hongrois" pour calculer la solution optimale d'un problème LAP.

1.5 Variantes

Le problème se décline en différentes versions selon certaines caractéristiques de la situation à modéliser. Par exemple :

— Quand les coefficients c_{ij} traduisent une notion de profit, la fonction économique du problème d'affectation linéaire est à maximiser et la solution optimale représente l'affectation de bénéfice maximum.

— Quand certaines affections tâche-ressource sont interdites a priori, le problème s'exprime de la même manière, en fixant $c_{ij} = \infty$ pour les affectations (i, j) interdites.

— Plusieurs variantes de structure sont liées au problème d'affectation, comme le *problème d'affectation généralisé* (Generalized Assignment Problem, GAP) ou encore le *problème d'affectation avec contrainte de ressource* (Resource Constrainted Assignement Problem, RCAP).

— Le problème d'affectation connait une variante importante dénommée *problème d'affectation quadratique* (QAP). Soit un ensemble de n localisations et de n activités. Pour chaque paire de localisation, une distance est spécifiée, et pour chaque paire d'activités, un flot ou un poids est spécifié (par exemple traduisant la quantité de matériel transporté entre les deux activités). Le problème consiste à assigner toutes les activités aux différentes localisations en minimisant la somme des distances multipliée par les flots correspondants.

1.6 Applications

Tout naturellement, le LAP permet de représenter une multitude de situations relevant de l'affectation tâches-ressources (processeurs-jobs, équipes-travaux, etc.). On le retrouve également comme sous-problème de situations plus complexes (comme en confection d'emploi du temps). Les variantes évoquées connaissent aussi des applications importantes, comme par exemple le QAP se rencontre en transport aérien et en conception VLSI (Very-Large-Scale Integration) dans le secteur de l'électronique par exemple.

1.7 Lectures et ressources conseillées

— *The Hungarian Method for the assignment problem.* Harold W. Kuhn, Naval Research Logistic Quarterly, 2 :83-97, 1955.

— *Assignment Problems.* R.E. Burkard, M. Dell'Amico, S. Martello. 393 pages. SIAM : Society for Industrial and Applied Mathematics. Philadelphia, 2012.
`http://www.assignmentproblems.com`

— *Assignment problems : A golden anniversary survey.* David W. Pentico. *European Journal of Operational Research* 176 (2007) 774-793.

— *QAPLIB – A quadratic assignment problem library.* R.E. Burkard, S. Karisch and F. Rendl, Journal of Global Optimization 10, 1997, 391-403.
`http://www.opt.math.tu-graz.ac.at/qaplib/`

2 Voyageur de commerce, TSP

2.1 Introduction

On parle du *problème du voyageur de commerce* (Traveling Salesman Problem, TSP) par analogie avec la situation d'un représentant itinérant qui, partant de son domicile, doit visiter une liste définie de clients et revenir à son domicile, le tout en parcourant la distance la plus courte. Dans la version de base, on considère que toutes les villes sont atteignables depuis n'importe quelle ville (graphe complet) et le parcours villeA-villeB est indifférent au parcours villeB-villeA (cas symétrique).

Plus formellement, soit n villes et c_{ij} le coût (ou la distance) correspondant au trajet $i-j$. L'ensemble des villes et des coûts est décrit par un graphe $G = (V, E, d)$ complet pondéré non orienté, où l'ensemble des sommets V représente l'ensemble des villes, l'ensemble des arêtes E constitue l'ensembles des routes et les valeurs des arêtes donnent les distances d. Le problème consiste à déterminer le plus petit circuit hamiltonien (un tour de plus court chemin), c'est-à-dire un circuit passant une et une seule fois par les n villes, de coût minimum.

2.2 Modèle

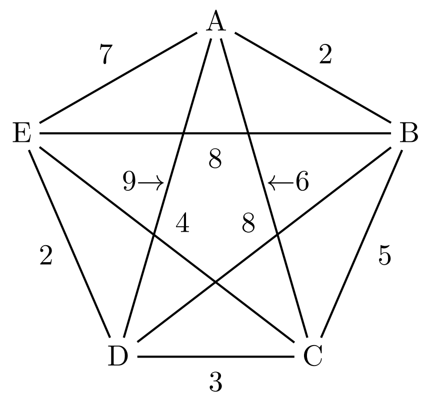
Soit la variable x_{ij} qui vaut 1 si on effectue le trajet $i-j$, 0 sinon. Une formulation sous forme de problème linéaire en variables 0-1 du TSP de base, dit symétrique (aucune contrainte ou restriction n'est donné a priori), est donnée par

$$\left[\begin{array}{lllll}
\min z & = & \displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij} & & \\
s/c & & \displaystyle\sum_{i=1}^{n} x_{ij} & = 1 & j = 1, \ldots, n \\
 & & \displaystyle\sum_{j=1}^{n} x_{ij} & = 1 & i = 1, \ldots, n \\
 & & \displaystyle\sum_{(i,j)\in S} x_{ij} & \leq \mid S \mid -1 & \forall S \subset \{1, \ldots, n-1\},\ S \neq \emptyset \\
 & & x_{ij} & \in \{0, 1\} & i, j = 1, \ldots, n
\end{array}\right] \tag{TSP}$$

où S représente un sous-ensemble de l'ensemble $\{1, \ldots, n\}$.

2.3 Exemple

Soit un TSP symétrique composé de 5 villes A, B, C, D, E à visiter et des distances d_{ij} $(i = 1, \ldots, 5; j = 1, \ldots, 5)$. Un tel problème peut se représenter par un graphe complet :



(TSP)

2.4 Commentaire

Le solveur “concorde” (`http://www.math.uwaterloo.ca/tsp/concorde.html`) permet de résoudre exactement des TSP symétriques comportant jusqu'à 85900 villes (record établi en 2016).

2.5 Variantes

Le problème se décline en différentes versions selon certaines caractéristiques de la situation à modéliser. Par exemple :

- si certains chemins n'existent pas entre deux lieux à visiter (cas des routes à sens unique), ou que le parcours ville A-B n'est pas équivalent au parcours B-A, le TSP est dit asymétrique (A-TSP).
- si des fenêtres de temps pour visiter les clients sont exprimées, le problème devient un TSP-TW (pour TSP with Time Windows).
- si la visite de l'ensemble des clients est impossible, il est envisageable de visiter les clients intéressants. Le problème devient un TSP avec profits, lequel comporte deux objectifs : minimiser la distance parcourue et maximiser le gain collecté par la visite des clients.
- lorsque l'on évoque une tournée qui est réalisée par un couple véhicule terrestre/drone aérien, situation qui rencontre une attention particulière de la communauté scientifique depuis 2015, on parle de TSP-D (TSP with drone).

2.6 Applications

Les problèmes de tournées (routing) couvrent un vaste ensemble d'applications issues notamment du domaine des transports et de la logistique. Le TSP se retrouve au centre de nombreux problèmes de tournées de véhicules (VRP, pour Vehicle Routing Problem), un TSP étant simplement un VRP mettant en oeuvre un véhicule unique qui possède une capacité de transport illimitée.

Parmi les problèmes de VRP, citons le CVRP (Capacitated VRP), famille de VRP avec capacités sur les sommets), le CARP (Capacitated Arc Routing Problem), famille de VRP avec les capacités sur les arcs ou encore le DARP (Dial-A-Ride Problem), famille de VRP à la demande. Une multitude de situations particulières spécialise le problème à traiter selon que, par exemple il s'agit d'un problème pick-up ou delivery ou les deux simultanément, où le parc de véhicule est homogène ou pas, etc.

2.7 Lectures et ressources conseillées

— *The Traveling Salesman Problem : A Guided Tour of Combinatorial Optimization*. E. L. Lawler, Jan Karel Lenstra, A. H. G. Rinnooy Kan, D. B. Shmoys. 476 pages. Wiley, 1985. ISBN-10 : 0471904139

— *The Traveling Salesman Problem and Its Variations*. Gregory Gutin, Abraham P. Punnen. 830 pages. Springer, 2007. ISBN : 0387444599

— *The Traveling Salesman Problem*.
`https://www.tsp.gatech.edu/`

— *TSPLIB - a library of sample instances for the TSP* .
`http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/`

3 Sac-à-dos, KP

3.1 Introduction

Le problème est dit de *sac-à-dos* (Knapsack Problem, KP) par analogie avec la situation d'un campeur qui part en randonnée et qui est confronté au choix de l'équipement vital à emporter. Il doit remplir son sac d'objets de manière à maximiser la valeur des objets choisis sans dépasser la capacité du sac.

Plus formellement, soit n objets $j = 1, \ldots, n$ de valeur p_j et de poids w_j. Soit un sac-à-dos de capacité en poids ω. Le problème consiste à maximiser la valeur des objets choisis sans dépasser la capacité du sac.

3.2 Modèle

Dans le KP de base, appelé sac-à-dos unidimensionnel en variables binaires (01UKP), on considère une seule contrainte de ressource et l'ensemble des items à choisir est disponible en un seul exemplaire. Soit la variable x_j qui vaut 1 si l'objet j est choisi, 0 sinon. Une formulation du 01UKP sous forme de problème linéaire en variables 0-1 est donnée par

$$\left[\begin{array}{cclll} \max z(x) & = & \sum_{j=1}^{n} p_j x_j & & \\ s/c & & \sum_{j=1}^{n} w_j x_j & \leq & \omega \\ & & x_j \in \{0,1\} & & j = 1, \ldots n \end{array}\right] \tag{01UKP}$$

Usuellement le 01UKP est considéré avec des p_j et des w_j entiers positifs, ainsi que $w_j \leq \omega$, $j = 1, \ldots, n$. Ces hypothèses ne sont pas restrictives.

3.3 Exemple

Soit 5 items, de profit $p = (15, 7, 14, 18, 17)$, de poids $w = (12, 7, 15, 24, 23)$ et une capacité maximale $\omega = 53$. Le 01UKP correspondant s'écrit :

$$\left[\begin{array}{ccll} \max z(x) & = & 15x_1 + 7x_2 + 14x_3 + 18x_4 + 17x_5 & \\ s/c & & 12x_1 + 7x_2 + 15x_3 + 24x_4 + 23x_5 & \leq \quad 53 \\ & & x_1, x_2, x_3, x_4, x_5 \in \{0,1\} & \end{array}\right]$$

3.4 Variantes

Plusieurs variantes autour du problème de base sont connues ; parmi celles-ci :

- quand plusieurs contraintes de ressources sont à considérer, on parle de problème MKP pour multidimensionnel. La variante bi-dimensionnelle représente une situation importante.
- quand les items sont disponibles en plusieurs exemplaires, on parle de problème en nombres entiers. Quand le nombre d'exemplaire est limité, on parle de variante bornée (bounded), sinon de variante non bornée (unbounded).

Les problèmes de sac-à-dos se retrouvent au sein d'une multitude d'autres problèmes, comme le bin-packing où il s'agit de mettre un ensemble d'éléments fixés dans des sacs de même capacité en utilisant le moins possible de sacs.

3.5 Applications

Ce problème couvre un nombre important d'applications dans des domaines comme :

- la finance : choix d'investissements (capital budgeting), portfolio selection, interbank clearing systems, etc.
- la cryptographie : système Merkle-Hellman en cryptographie ;
- le chargement de navire, de satellite, etc. (cargo loading) ;
- la découpe (cutting stock problem).

Comme pour le LAP, on le retrouve également comme sous-problème de situations plus complexes (le RCAP en est une illustration immédiate).

3.6 Lectures et ressources conseillées

- *Knapsack Problems*. Silvano Martello, Paolo Toth. 296 pages. John Wikey and Sons Ltd, 1990. ISBN : 0 471 92420 2
  (téléchargeable sur le web à l'adresse `www.or.deis.unibo.it/knapsack.html`)
- *Knapsack Problems*. Hans Kellerer, Ulrich Pferschy, David Pisinger. 546 pages. Springer-Verlag, 2004. ISBN : 978-3-540-40286-2
- *The multidimensional 0–1 knapsack problem : An overview*. Arnaud Fréville. European Journal of Operational Research 155, 1–21, 2004.
- *David Pisinger's optimization codes for KP*
  `http://hjemmesider.diku.dk/~pisinger/codes.html`

4 Couplage généralisé, SPP

4.1 Introduction

Le *problème de couplage généralisé* (Set Packing Problem, SPP), le *problème de partitionnement d'ensembles* (Set Partitionning Problem, SPA) et le *problème de couverture d'ensembles* (Set Covering Problem, SCP) sont trois problèmes pouvant s'exprimer d'une manière très proche mais qui possèdent leur littérature respective.

Le SPP permet de représenter des situations de disjonction de ressources (au plus un train par canton simultanément, au plus un rendez-vous parent-élève simultanément, etc.). Une variable du problème traduit les ressources mobilisées par une activité, sachant que la ressource ne peut pas être partagée.

4.2 Modèle

Soit n activités possibles ; une activité j (avec $j = 1, \ldots, n$) apporte un profit c_j si celle ci est mise en œuvre ($x_j = 1$), et consommera un sous-ensemble des m ressources non partageables ($a_{ij} = 1$ pour les $i \in S_j \subseteq \{1, \ldots, m\}$ ressources consommée par l'activité j).

$$\left[\begin{array}{lll} \max z(x) = & \displaystyle\sum_{j=1}^{n} c_j x_j & \\ s/c & \displaystyle\sum_{j=1}^{n} a_{ij} x_j \leq 1 & i = 1, \ldots, m \\ & a_{ij} \in \{0,1\} & i = 1, \ldots, m \\ & & j = 1, \ldots, n \\ & x_j \in \{0,1\} & j = 1, \ldots, n \end{array}\right] \tag{SPP}$$

4.3 Exemple

Soit 5 activités potentielles (par exemple des trains au passage sur une infrastructure) de profit $c = (1, 1, 1, 1, 1)$ (pas de préférence entre les trains), tel que chaque activité (passage d'un train) demande respectivement les ressources (cantons de l'infrastructure) suivantes : $\{1, 2, 5\}$, $\{1, 3\}$, $\{2, 4\}$, $\{3, 6\}$, $\{2, 3, 6\}$:

$$\left[\begin{array}{cccccccccccccc} \max z(x) & = & x_1 & + & x_2 & + & x_3 & + & x_4 & + & x_5 & & & \\ s/c & & x_1 & + & x_2 & + & & + & & + & & \leq & 1 & (1) \\ & & x_1 & + & & + & x_3 & + & & + & x_5 & \leq & 1 & (2) \\ & & & + & x_2 & + & & + & x_4 & + & x_5 & \leq & 1 & (3) \\ & & & + & & + & x_3 & + & & + & & \leq & 1 & (4) \\ & & x_1 & + & & + & & + & & + & & \leq & 1 & (5) \\ & & & + & & + & & + & x_4 & + & x_5 & \leq & 1 & (6) \\ & & x_1, & & x_2, & & x_3, & & x_4, & & x_5 & \in & \{0,1\} & \end{array}\right]$$

Remarque : l'usage des sous-ensembles S_j formalise avantageusement ces propos comme le présente le problème en section 5.

4.4 Applications

Le SPP permet de représenter un nombre important de situations présentant des conflits de ressources, comme rencontré dans la mesure de la capacité d'une infrastructure ferroviaire (gare, jonction complexe, etc.), l'organisation d'interviews dans un salon de recrutements (flash meetings) ou encore le placement d'étiquettes sur une carte (PFLP, Point-feature label placement Problem).

4.5 Variantes

Lorsque tous les coûts sont identiques dans un SPP, on parle de Unicost Set Packing Problem (USPP), en contraste avec Weighted Set Packing Problem (WSPP) sinon. Aussi, le node packing problem (NPP) est une variante du SPP dans laquelle seules deux variables apparaissent dans une contrainte du problème :

$$x_i + x_j \leq 1 \quad i, j \in \{1, \ldots, n\}$$

4.6 Lectures et ressources conseillées

- *Integer and combinatorial optimization.* George L. Nemhauser, Laurence A. Wolsey. 763 pages. Wiley-Interscience Series in Discrete Mathematics and Optimization. 1999.
- *Applications of Set Covering, Set Packing and Set Partitioning Models : A Survey.* R.R. Vemuganti. In : DZ. Du, P. M. Pardalos (eds), "Handbook of Combinatorial Optimization". Springer, Boston, MA. 1998.
- *Modélisation et résolution de problèmes liés à l'exploitation d'infrastructures ferroviaires.* Xavier Delorme. Thèse de doctorat, Université de Valenciennes, 2003.

5 Partitionnement d'ensembles, SPA

5.1 Introduction

Le SPA permet d'exprimer des situations de missions impérieuses (par exemple, une activité est une tournée réalisée par un véhicule, une contrainte est un client qui doit être impérativement visité, une tournée visite un ensemble de clients, un client devant être visité une unique fois). A une activité j (avec $j = 1, \ldots, n$) du problème est associé un ensemble $S_j \subseteq \{1, \ldots, m\}$ traduisant les missions réalisées par celle-ci, sachant que toutes les missions doivent être réalisées, et cela une unique fois. Disposant de $\{S_1, \ldots, S_n\}$ une collection de sous ensembles de S avec $S_j \in S$, l'objectif est de trouver une partition

$$\bigcup_{j=1,\ldots,n} S_j = \{1, \ldots, m\} \quad \text{et} \quad \bigcap_{j=1,\ldots,n} S_j = \emptyset$$

de coût minimum.

5.2 Modèle

Soit n activités possibles ; une activité j (avec $j = 1, \ldots, n$) induit un coût c_j si celle ci est mise en œuvre ($x_j = 1$), et réalisera un sous-ensemble des m missions ($a_{ij} = 1$ pour les $i \in S_j \subseteq \{1, \ldots, m\}$ missions réalisées par l'activité j).

$$\left[\begin{array}{lcll} \min z(x) & = & \displaystyle\sum_{j=1}^{n} c_j x_j & \\ s/c & & \displaystyle\sum_{j=1}^{n} a_{ij} x_j = 1 & i = 1, \ldots, m \\ & & a_{ij} \in \{0,1\} & i = 1, \ldots, m \\ & & & j = 1, \ldots, n \\ & & x_j \in \{0,1\} & j = 1, \ldots, n \end{array}\right] \qquad \text{(SPA)}$$

5.3 Exemple

Soit 6 activités ($n = 6$) de coût $c = (c_1, c_2, c_3, c_4, c_5, c_6)$, 6 missions ($m = 6$) et les missions réalisées par chaque activité : $S_1 = \{4, 5\}$, $S_2 = \{3\}$, $S_3 = \{1, 3\}$, $S_4 = \{6\}$, $S_5 = \{2, 5\}$, $S_6 = \{1, 4\}$. La matrice $A = (a_{i,j})$ avec $i = 1, \ldots 6$ et $j = 1, \ldots 6$ s'écrit :

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

5.4 Applications

Les applications sont nombreuses dans le domaine du transport, dont notamment :

— la construction de rotations d'équipage dans le transport aérien (crew pairing) ;

— tournées de véhicules avec contrainte de capacité (CVRP).

5.5 Lectures et ressources conseillées

— *The Set-Partitioning Problem : Set Covering with Equality Constraints*. Robert S. Garfinkel, George L. Nemhauser. Operations Research, 17(5), 848–856. 1969.

— *Set Partitioning : A survey*. Egon. Balas, Manfred. W. Padberg. SIAM Review, 18(4), 710–760. 1976.

— *Applications of Set Covering, Set Packing and Set Partitioning Models : A Survey.* R.R. Vemuganti. In : DZ. Du, P. M. Pardalos (eds), "Handbook of Combinatorial Optimization". Springer, Boston, MA. 1998.

6 Couverture d'ensembles, SCP

6.1 Introduction

Le SCP présente une proximité avec le SPP et le SPA, mais également avec la classe des *problèmes de localisation de services* (FLP pour Facility Location Problem). Il permet d'exprimer des situations dans lesquelles une exigence doit être au moins respectée (réponsant à des énoncés exprimant "mettre d'astreinte au moins un anesthésiste dans l'équipe de garde entre 2h et 4h du matin" ou encore "assurer la présence d'un bureau de poste dans l'arrondissement de résidence ou au moins dans un arrondissement limitrophe").

6.2 Modèle

Soit n activités possibles ; une activité j (avec $j = 1, \ldots, n$) induit un coût c_j si celle ci est mise en œuvre ($x_j = 1$), et satisfera un sous-ensemble des m exigences ($a_{ij} = 1$ pour les $i \in S_j \subseteq \{1, \ldots, m\}$ exigences satisfaites par l'activité j).

$$\left[\begin{array}{lll} \min z(x) = & \displaystyle\sum_{j=1}^{n} c_j x_j & \\ s/c & \displaystyle\sum_{j=1}^{n} a_{ij} x_j \geq 1 & i = 1, \ldots, m \\ & a_{ij} \in \{0, 1\} & i = 1, \ldots, m \\ & & j = 1, \ldots, n \\ & x_j \in \{0, 1\} & j = 1, \ldots, n \end{array}\right] \qquad \text{(SCP)}$$

6.3 Exemple

Les exemples numériques utilisés pour le SPP et SPA se transposent directement pour le SCP au moment d'écrire le vecteur des coûts et la matrice A intervenant dans le système des contraintes.

6.4 Applications

Le SCP est également utilisé pour représenter de nombreuses situations comme par exemple l'ouverture de restaurants de proximité dans une agglomération, le positionnement optimal d'unités d'intervention d'urgences (ambulances, pompiers) sur des territoires sensibles (agglomération, zone forrestière, etc.), la confection d'emploi du temps pour élaborer des équipes médicales en milieu hospitaliser répondant à une charge de travail estimée sur un empan temporel de 24 heures, mais aussi comme relaxation à des situations pour lesquelles il n'existe pas de solution admissible à un problème SPA.

6.5 Variantes

Comme pour le SPP, lorsque tous les coûts sont identiques dans un SCP, on parle de Unicost Set Covering Problem (USCP), en contraste avec Weighted Set Covering Problem (WSCP) sinon.

6.6 Lectures et ressources conseillées

- *Integer and combinatorial optimization.* George L. Nemhauser, Laurence A. Wolsey, 763 pages. Wiley-Interscience Series in Discrete Mathematics and Optimization. 1999.
- *Applications of Set Covering, Set Packing and Set Partitioning Models : A Survey.* R.R. Vemuganti. In : DZ. Du, P. M. Pardalos (eds), "Handbook of Combinatorial Optimization". Springer, Boston, MA. 1998.
- *Aspects of Set Packing, Partitioning, and Covering.* Ralf Borndörfer, Thèse de doctorat, TU Berlin, Allemagne, 1998.

7 Localisation de services, FLP

7.1 Introduction

Il existe autant de *problèmes de localisation de services* (Facility Location Problem, FLP) qu'il y a des situations pratiques qui se présentent. On désigne par FLP cette classe de problème d'optimisation qui peut se présenter comme un problème posé totalement en variables discrètes, mais aussi comme un problème posé en variables mixtes (discrètes et continues).

Un des modèles les plus simples est le FLP discret sans capacité (Uncapacity Facility Location Problem, UFLP) pour lequel des services (facilités) doivent être sélectionnés dans une liste de services potentiels établie a priori, afin d'être ouverts et ainsi desservir de manière optimale l'ensemble de demandeurs (clients) intervenants dans la situation, les demandes étant unitaires et non fractionnables. L'ouverture d'un service s'accompagne d'un coût de démarrage et l'association demandeur-service engendre un coût de connexion, les deux quantités étant à minimiser.

7.2 Modèle

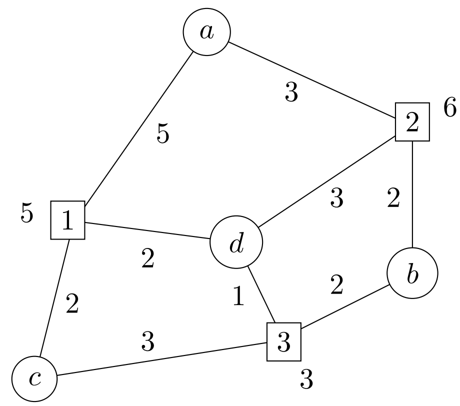
Soit I demandeurs (clients) et J services potentiels (facilités). On dispose de r_j (avec $j \in J$) qui véhicule le coût de démarrage pour le service j, et c_{ij} (avec $i \in I,\ j \in J$) qui donne le coût de connexion du demandeur i au service j.

$$\left[\begin{array}{lllll} \min z(x) & = & \sum_{i\in I}\sum_{j\in J} c_{ij}x_{ij} + \sum_{j\in J} r_j s_j & & \\ s/c & & \sum_{j\in J} x_{ij} = 1 & \forall i \in I & (1) \\ & & x_{ij} \leq s_j & \forall i \in I,\ \forall j \in J & (2) \\ & & x_{ij},\ s_j \in \{0,1\} & \forall i \in I,\ \forall j \in J & (3) \end{array}\right] \qquad \text{(UFLP)}$$

Les contraintes techniques signifient que chaque demandeur doit être connecté à un service (contraintes (1)) et la connexion d'un demandeur à un service ne peut se faire que si celui-ci est ouvert (contraintes (2)). Les contraintes d'intégrité traduisent que la demande n'est pas fractionnable et un service est ouvert ou non (contraintes (3)).

7.3 Exemple

L'exemple ci-dessous illustre un UFLP avec 3 facilités (symbole carré) et 4 clients (symbole rond). Les coûts de démarrage des services sont $r = (5, 6, 3)$ et les coûts des connexions sont repris sur les arêtes du graphe :



7.4 Applications

Les applications de FLP se retrouvent dans nombreux domaines opérationnels comme les réseaux informatique (déploiement d'un service par un nouvel opérateur à l'échelle d'un pays), l'aménagement du territoire (localisation de stations de transport en commun), la prévention de calamités (relocalisation de constructions hors zones inondables), la logistique humanitaire (déploiement d'antennes médicales suite à une catastrophe naturelle ou industrielle), la supply chain (emplacement de dépôts, de hubs, etc.).

7.5 Lectures et ressources conseillées

— *Facility Location. Applications and Theory*. Zvi Drezner, Horst W. Hamacher, (Eds.). Springer. 457 pages. 2004.
— *Facility Location. A Survey of Applications and Methods*. Zvi Drezner (Ed.). Springer Series in Operations Research. 571 pages. 1995
— *Facility Location. Concepts, Models, Algorithms and Case Studies*. Reza Zanjirani Farahani, Masoud Hekmatfar (Eds.). Springer - Contributions to Management Science. 549 pages. 2009.