Programmation Effective – TD 08 : Méthode hongroise

matthieu.gallet@ens-lyon.fr

mardi 1er avril 2008

1 Introduction

L'objet de ce TD est d'explorer une méthode de recherche opérationnelle appelée *méthode hongroise*, expliquée par Harold Kuhn en 1955 (*The Hungarian method for the assignment problem*). La recherche opérationnelle est un terme assez général pour désigner un ensemble de méthode d'analyse et d'optimisation de problèmes rencontrés dans l'industrie (comme la gestion des ressources, l'ordonnancement de projets, le choix des investissements). La méthode hongroise permet de résoudre des problèmes d'affectation, qui consistent à trouver la meilleure façon de relier un ensemble donné de tâches à un ensemble de ressources (machines-outils, processeurs, etc.).

1.1 Problème et notations

On suppose que l'on a un ensemble de m tâches, ainsi qu'un ensemble de $n \geq m$ ressources (ou travailleur, ou processeur). Ni les ressources, ni les tâches ne sont *a priori* identiques, et on suppose donc qu'on dispose d'une matrice e, de taille $n \times m$, telle que $e_{i,j}$ représente le coût payé si on affecte la tâche j sur le processeur i. Naturellement, on cherche à diminuer le coût à payer pour traiter l'ensemble des tâches. Si au contraire, $e_{i,j}$ représente un profit engendré par le placement de la tâche j sur la ressource i, il faudra modifier un peu la matrice avant l'exécution de l'algorithme pour que la minimisation correspond à un profit maximal.

1.2 Mise sous la forme d'un problème de graphe

Considérons un ensemble de sommets $L = \{s_1, \ldots, s_m\}$ et un autre ensemble de sommets $R = \{t_1, \ldots, t_n\}$. Pour chaque $e_{i,j}$, on crée une arête (s_i, t_j) de poids très exactement égal à $e_{i,j}$. Nous obtenons alors facilement un graphe biparti $G = (S, A)$ avec $S = L \cup R$. Considérons une affectation quelconque. Si t_j est affecté à la ressource s_i, alors on sélectionne l'arête correspondante. On obtient donc à partir de l'affectation d'origine un couplage. De même, si on a un couplage quelconque, on en tire aisément une affectation valide. Ainsi, rechercher la meilleure affectation revient à chercher un couplage M de poids minimal (ou un couplage de poids maximal, quitte encore une fois à modifier un peu les poids).

2 Algorithme de couplage biparti de Hopcroft-Karp

Contrairement au but recherché, cet algorithme ne fonctionne qu'avec les graphes non pondérés mais donne une première idée de l'algorithme. Quelques définitions :

chaîne alternante : Une chaîne élémentaire P de $G = (S, A)$ est une chaîne alternante si ses arêtes appartiennent alternativement à M et à $A - M$.

chaîne améliorante : Une chaîne élémentaire P de $G = (S, A)$ est une chaîne améliorante par rapport à M si elle est alternante, si elle commence en un sommet non couplé de L, si elle finit en un sommet non couplé de R.

différence symétrique : La différence symétrique de deux ensembles A et B est définie par $A \oplus B = (A - B) \cup (B - A)$.

On peut montrer que si M est un couplage et P une chaîne améliorante par rapport à M, alors la différence symétrique $M \oplus P$ est un couplage et $|M \oplus P| = |M| + 1$. On peut également montrer que si $P_1, P_2, \ldots, P_k$ sont des chaînes améliorantes par rapport à M sans sommet comun, alors la différence symétrique $M \oplus (P_1 \cup P_2 \cup \ldots \cup P_k)$ est un couplage de cardinalité $|M| + k$. Équipé de ces remarques, on peut examiner la méthode de Hopcroft-Karp décrite par l'algorithme 1.

Algorithm 1 Hopcroft-Karp(G)

```
M ← ∅
répéter
   soit 𝒫 ← {P_1, P_2, ..., P_k} un ensemble maximum de plus courtes chaînes améliorantes par rapport
   à M sans sommet commun en temps O(|A|)
   M ← M ⊕ (P_1 ∪ P_2 ∪ ... ∪ P_k)
jusqu'à 𝒫 = ∅
renvoyer M
```

L'algorithme de Hopcroft-Karp peut s'exécuter en temps $O\left(\sqrt{|S|}|A|\right)$, soit $O\left(nm\sqrt{n+m}\right)$.

3 Méthode hongroise, ou méthode de Kuhn

3.1 Sans considérations de graphes

Dans la suite, nous supposons que nous avons 4 machines a, b, c et d, ainsi que 4 tâches α, β, γ et δ, et que la matrice e soit de la forme :

$$\begin{pmatrix} & \alpha & \beta & \gamma & \delta \\ a & 0 & 1 & 3 & 5 \\ b & 4 & 8 & 4 & 0 \\ c & 2 & 0 & 2 & 5 \\ d & 7 & 1 & 0 & 2 \end{pmatrix}$$

Comme on a une matrice de coûts, à l'évidence la meilleure solution est simplement d'affecter α à a, β à c, γ à d et enfin δ à b, permettant un coût nul (difficile, donc, de faire mieux). Cet exemple permet de comprendre pourquoi on cherche à faire apparaître des zéros, et montre qu'une solution (ou affectation) sera bien un ensemble de zéros de la matrice, c'est-à-dire un ensemble de coordonnées *(machine, tâche)*.

Cependant, l'hypothèse d'un unique zéro par ligne et colonne est bien évidemment trop forte et il s'agit de s'y ramener au moins en partie. Quand on aura plusieurs zéros, il faudra faire un choix.

On dispose de deux types de marquages qu'on peut apposer aux zéros de la matrice, « étoilé » et « préparé ». Voici une description de l'algorithme :

1. Dans chaque ligne de e, prendre le plus petit élément et le soustraire à l'ensemble de la ligne. Ainsi, on est sûr de créer au moins un zéro.
2. Trouver un zéro Z dans la matrice résultante. S'il n'y a pas de zéros étoilé dans sa ligne ou dans sa colonne, le marquer comme étant étoilé. Répéter cette étape pour chaque zéro de la matrice.
3. Couvrir chaque colonne (on rappelle que les colonnes correspondent aux tâches) contenant un zéro étoilé. S'il y a m colonnes couvertes, alors on a une affectation complète décrite par les zéros étoilés, on peut arrêter ici l'algorithme (étape 7).
4. Trouver un zéro non couvert et le marquer comme préparé. S'il n'y a pas de zéro étoilé dans la rangée contenant ce zéro préparé, aller à l'étape 5. Sinon, couvrir cette rangée et découvrir la colonne contenant le zéro étoilé. Continuer de cette façon jusqu'à ce qu'il n'y ai plus de zéros découverts. Noter la plus petite valeur non couverte et aller à l'étape 6.

5. Construire une suite de zéros en alternants zéros préparés et zéros étoilés de la manière suivante : on note Z_0 le zéro préparé non couvert trouvé à l'étape 4. On note alors Z_1 le zéro étoilé dans la colonne de Z_0 s'il existe. On note Z_2 un zéro préparé dans la rangée de Z_1, etc. On continue ainsi jusqu'à tomber sur sur un zéro préparé tel qu'il n'y ait pas de zéro étoilé dans sa colonne. Enlever le marquage (étoilé ou préparé) de tous les zéros de la suite, marquer comme étoilés les zéros de cette même suite, découvrir chaque ligne de la matrice. Finalement, retourner à l'étape 3.
6. Ajouter la valeur trouvée à l'étape 4 à tous les éléments de chaque ligne couverte, et la soustraire à tous les éléments de chaque colonne non couverte. Retourner à l'étape 4 sans modifier les marquages de couverture ou de zéros préparés ou étoilés.
7. Fin de l'algorithme : les zéros étoilés donnent les couplages à faire : si $e_{i,j}$ est un zéro étoilé, alors il faut associer la tâche j à la ressource i.

3.2 Retour aux graphes

Nouvelles définitions :

étiquetage : On appelle étiquetage une fonction l de S dans $\mathbb{R}$

étiquetage faisable : Un étiquetage faisable (ou réalisable) vérifie pour tout $i \in L$, pour tout $j \in R$, la condition $l(i) + l(j) \geq e_{i,j}$.

graphe d'égalité : On appelle G_l le graphe d'égalité engendré par l le graphe défini par $G_l = (S, A_l)$, avec $A_l = \{(i, j) : l(i) + l(j) = e_{i,j}\}$.

Théorème 1 (Théorème de Kuhn-Munkres). *Si l est un étiquetage faisable et que M est un couplage parfait dans G_l, alors M est un couplage de poids maximum.*

Interprétons maintenant l'algorithme précédent en termes de graphes :
– Les zéros étoilés (de coordonnées (i, j)) sont les arêtes (i, j) sélectionnées dans notre couplage.
– Si ces zéros étoilés ne forment pas un couplage parfait, alors on va chercher une chaîne améliorante de zéros (et oui, ils sont alternés)
– Les modifications des valeurs de la matrice (étape 6) correspondent à la recherche d'un nouvel étiquetage du graphe.

3.3 Un exemple d'application

Une généreuse grand-mère décide de gâter 4 de ses petits-enfants (Alison, Brad, Charleen, Dylan) en leur offrant à chacun un cadeau. Férue de technologie, elle est décidée à acheter 4 cadeaux différents :
– une console de jeu vidéo
– un téléphone portable
– un baladeur MP3
– un appareil photo numérique

Elle n'achètera qu'un seul appareil de chaque type. Chaque enfant aura donc l'un de ces cadeaux. Pour savoir comment répartir ces cadeaux, elle demande à ses petits-enfants d'indiquer pour chacun de ces types d'objets le prix de l'appareil de ses rêves. Généreuse mais pas folle, elle décide après avoir eu les réponses d'attribuer les cadeaux de façon à minimiser sa dépense totale. Aidez-la donc à trouver une affectation des cadeaux qui écorne au minimum son budget, sachant qu'elle a eu les réponses suivantes :

	Alison	Brad	Charleen	Dylan
Console	130	650	165	330
Téléphone	170	250	90	250
Baladeur	330	320	335	330
Photo	100	400	230	400

4 Applications

Pas trouvé d'applications sur ce thème :(En revanche, résolvez tout de même les problèmes suivants : Spreadsheet (http://acm.uva.es/p/v1/196.html), Rare Order (http://acm.uva.es/p/v2/200.html), The House Of Santa Claus (http://acm.uva.es/p/v2/291.html), Jack and Jill (http://acm.uva.es/p/v6/697.html), ainsi que Street Numbers (http://acm.uva.es/p/v1/138.html).