Algorithmique et Programmation

TD n° 9 : Programmation linéaire

École normale supérieure – Département d'informatique
algoL3@di.ens.fr

2015-2016

Les variables en **gras** sont des vecteurs.
Rappel : une des formes d'un programme linéaire et son dual est

$$\begin{array}{rr} \max \boldsymbol{cx} & \min \boldsymbol{yb} \\ A\boldsymbol{x} \leq \boldsymbol{b} & \boldsymbol{y}A \geq \boldsymbol{c} \\ \boldsymbol{x} \geq \boldsymbol{0} & \boldsymbol{y} \geq \boldsymbol{0} \end{array}$$

Exercice 1. Couverture d'ensemble

Étant donné un ensemble $\mathcal{S}$ de sous-ensemble de $\{1, 2, \ldots, n\}$, on veut choisir un nombre minimum d'éléments de $\{1, 2, \ldots, n\}$ où tout $S \in \mathcal{S}$ continent au moins un de ces éléments (une *couverture* de $\mathcal{S}$).

1. Écrire un programme linéaire en nombre entier correspondant à ce problème.
2. Écrire le dual de la relaxation fractionnaire.
   À quel problème correspond le dual si on y ajoute des contraintes d'intégralité ?
3. Montrer que toute solution primal entière est supérieur à toute solution dual entière sans utiliser le théorème de dualité des programmes linéaires.
4. Montrer que le ratio entre l'optimums primal entier et l'optimums primal fractionnaire peut être linéaire en n.
5. Supposons que tous les éléments de $\mathcal{S}$ ont taille au plus k.
   Donner un algorithme (polynomial en n et $|\mathcal{S}|$) qui prends une solution primal fractionnaire $\boldsymbol{x}$ (pas nécéssairement optimale) et calcul une solution primal entière de valeur (taille) au plus k fois la valeur de $\boldsymbol{x}$.

Exercice 2. Flots

Soit un graphe $G = (V, E, c)$ orienté pondéré et de capacités positives $c : E(G) \to \mathbb{Q}^+$. Soient $s, t \in V(G)$. Nous cherchons à calculer le flot f maximum entre s et t.

Montrer que ce problème peut s'écrire sous forme de problème d'optimisation linéaire. Redémontrer à l'aide des outils de la programmation linéaire le théorème « flot maximum - coupe minimale ».

Exercice 3. Combinaison convexe

Une *combinaison convexe* (ou *moyenne pondérée*) de deux vecteurs $\boldsymbol{x}$ et $\boldsymbol{y}$ est un vecteur $\lambda\boldsymbol{x} + (1 - \lambda)\boldsymbol{y}$ où $\lambda \in [0, 1]$.

Plus généralement, une *combinaison convexe* de vecteurs $\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_k}$ est un vecteur $\sum_{i=1}^{k} \lambda_i \boldsymbol{x_i}$ où $\forall i, \lambda_i \geq 0$ et $\sum_{i=1}^{k} \lambda_i = 1$.

Montrer que toute combinaison convexe de solutions d'un programme linéaire sont une solution au programm linéaire.

Remarque : L'algorithme du simplexe donne toujours une solution qui est combinaison convexe d'aucune autre solution (un « point extrême »).

Exercice 4. COUPLAGE

Considérons le programme linéaire suivant $PL_1(G)$ pour trouver un couplage maximum dans un graphe G.

$$\begin{aligned} \text{maximiser} \quad & \sum_{e \in E(G)} x_e \\ \text{sous les contraintes} \quad & \sum_{u \in e} x_e \leq 1 \ \forall v \in V(G) \\ & 0 \leq x_e \leq 1 \ \forall e \in E(G) \end{aligned}$$

Nous pouvons créer un *programmation linéaire en nombres entiers* $PLNE_1(G)$ en ajoutant une contrainte supplémentaire qui restreint toutes les variables à prendre des valeurs entières. Par conséquent nous appellons $PL_1(G)$ la *relaxation continue* de $PLNE_1(G)$.

1. Trouver un graphe G pour lequel la valeur de la solution optimum à $PLNE_1(G)$ est diffèrente de la valeur optimum de $PL_1(G)$.
2. Le but du reste de l'exercise consiste à démontrer que toute solution à $PL_1(G)$ est combinaison convexe de solutions de $PLNE_1(G)$ pour un graphe biparti G.

   Dans ce cas, on dit que le nombre d'entrées fractionnaires d'une solution $\boldsymbol{x}$ est le nombre de variables x_e tel que $0 < x_e < 1$.

   Soit G un graphe biparti et $\boldsymbol{x}$ une solution à $PL_1(G)$. Montrer que s'il existe un cycle C où $0 < x_e < 1$ pour toute arête $e \in E(C)$ alors $\boldsymbol{x}$ est combinaison convexe de solutions avec moins d'entrées fractionnaires.
3. Montrer que toute solution à $PL_1(G)$ est combinaison convexe de solutions de $PLNE_1(G)$ pour un graphe biparti G.
4. Montrer que les énoncés de l'exercice précédent restent vrais si on remplace l'objectif $\sum_{e \in E(G)} x_e$ par $\sum_{e \in E(G)} c_e x_e$ pour des constantes $\boldsymbol{c}$ (on fait ce remplacement dans $PL_1(G)$ et $PLNE_1(G)$).

   En déduire un algorithme pour trouver un couplage maximum pondéré (c'est-à-dire un couplage dont la somme des poids de ses arêtes est maximum) dans un graphe biparti pondéré.
5. Donner un algorithme pour trouver un couplage parfait de poids maximum dans un graphe biparti pondéré (c'est-à-dire un couplage parfait dont la somme des poids de ses arêtes est maximum).
6. Supposons que nous ajoutons la contrainte C_3 suivante à $PL_1(G)$ pour obtenir un nouveau programme linéaire $PL_2(G)$.

   $$\sum_{e \in E(G[S])} x_e \leq \frac{|S| - 1}{2} \ \forall S \subseteq V, |S| \text{ impair}$$

   Quel est la valeur de la solution optimale de PL_2 pour le graphe trouvé en (a) (utiliser la méthode du simplexe ou une autre preuve) ? C_3 correspond à combien de contraintes en général ?
7. (**) Montrer que pour un graphe G, toute solution dans $PL_2(G)$ est une combinaison convexe de solutions dans $PLNE_2(G)$.