

CONTRÔLE DE RECHERCHE OPÉRATIONNELLE

FÉVRIER 2015

Tout document autorisé. Calculatrices et calculettes interdites.
Le total des points fait 30. La note sera laissée sur 20.

1. PLUS COURT CHEMIN – 2 PTS

On considère le réseau de rues représenté sur la Figure 1 en fin d'énoncé sous la forme d'un graphe non-orienté. Une voiture est située en s et souhaite se rendre en t par le trajet le plus court. Les distances sont indiquées à proximité des arêtes.

Indiquer le trajet le plus court, en justifiant votre réponse. (2 pts)

2. AFFECTATION DE TÂCHES, COUPLAGE ET MAKESPAN – 3 PTS

Soit $G = (V, E)$ un graphe biparti. Rappelons que $\nu(G)$ est la cardinalité maximale d'un couplage de G .

Supposons que G soit muni d'une fonction de poids $w : E \rightarrow \mathbb{R}_+$. Parmi tous les couplages M de cardinalité $\nu(G)$, on veut trouver celui minimisant $\max_{e \in M} w(e)$.

1. Montrer que ce problème peut être résolu en temps polynomial. Indication : dans un graphe biparti, un couplage de cardinalité maximale se trouve en temps polynomial. (2 pts)

Considérons un ensemble W d'employés et un ensemble T de tâches à effectuer. On veut affecter chaque tâche à un unique employé, mais pas plus d'une tâche par employé. On suppose donc que $|W| \geq |T|$.

D'autre part, on connaît $d(w, t)$ le temps mis par l'employé w pour réaliser la tâche t , et ce pour tout $w \in W$ et tout $t \in T$. Lorsque l'affectation des tâches aura été effectuée, tous les employés commenceront en même temps. On veut que le temps total requis pour la réalisation de l'ensemble des tâches soit le plus petit possible.

2. Montrer que la méthode proposée en 1. permet de résoudre ce problème en temps polynomial. (1 pt)

3. RELAXATION LAGRANGIENNE POUR UN PROBLÈME D'ENTREPÔTS – 11 PTS

On considère un ensemble I de magasins devant être approvisionnés régulièrement par des entrepôts. Ces entrepôts ne sont pas encore construits, mais on a un ensemble J de positions potentielles. Pour chaque position $j \in J$, on connaît le coût d'ouverture $c_j \geq 0$ de l'entrepôt, la capacité $k_j \geq 0$ qu'il aurait, ainsi que le coût de transport $a_{ij} \geq 0$ d'une fraction $1/d_i$ de la marchandise de cette position au magasin i .

Pour chaque magasin i , on connaît sa demande total $d_i \geq 0$. Un magasin peut être approvisionné par plusieurs entrepôts.

Le problème de la minimisation des coûts se modélise sous la forme du programme linéaire mixte suivant.

$$\begin{aligned}
 \min \quad & \sum_{i \in I, j \in J} a_{ij} x_{ij} + \sum_{j \in J} c_j y_j \\
 \text{s.c.} \quad & \sum_{j \in J} x_{ij} = 1 & \forall i \in I & (i) \\
 & \sum_{i \in I} d_i x_{ij} \leq k_j y_j & \forall j \in J & (ii) \\
 & x_{ij} \in [0, 1] & \forall i \in I, j \in J & (iii) \\
 & y_j \in \{0, 1\} & \forall j \in J & (iv)
 \end{aligned}$$

1. Donner la signification de chacune des variables et de chacune des contraintes (i), (ii), (iii) et (iv). (2 pts)

2. Procéder à la relaxation lagrangienne de la contrainte (i) : en introduisant des variables duales réelles $\lambda = (\lambda_i)_{i \in I}$, montrer que la borne inférieure $\mathcal{G}(\lambda)$ obtenue ainsi peut se calculer en temps polynomial pour tout $\lambda \in \mathbb{R}^I$ donné. (2 pts)

3. Au lieu de procéder à la relaxation lagrangienne de la contrainte (i), on peut procéder à celle de (ii). En introduisant des variables duales réelles positives $(\mu_j)_{j \in J}$, montrer que la borne inférieure $\mathcal{H}(\mu)$ obtenue ainsi peut se calculer en temps polynomial pour tout $\mu \in \mathbb{R}_+^J$ donné. (2 pts)

On peut montrer avec des outils dépassant le cadre de ce cours que l'on a toujours

$$\max_{\lambda \in \mathbb{R}^I} \mathcal{G}(\lambda) \geq \max_{\mu \in \mathbb{R}_+^J} \mathcal{H}(\mu) \quad (1)$$

4. Calculer explicitement la valeur de $\max_{\lambda \in \mathbb{R}^I} \mathcal{G}(\lambda)$ et de $\max_{\mu \in \mathbb{R}_+^J} \mathcal{H}(\mu)$ lorsque l'on a simultanément $|I| = |J| = 1$ et $k \geq d$. On pourra donc omettre les indices dans l'écriture des formules. (4 pts)

5. Lorsque $|I| = |J| = 1$ et $k \geq d$ simultanément, pour quelles valeurs des paramètres l'inégalité (1) est-elle stricte ? (1 pt)

4. REPOSITIONNEMENT DANS UN SYSTÈME DE VÉLOS PARTAGÉS – 5 PTS

On considère une ville équipée d'un système de vélos partagés. Ce système est formé d'un ensemble S de stations. Toute station $s \in S$ a une capacité k_s , qui est le nombre maximum de vélos pouvant être présents simultanément dans cette station. Un camion est utilisé pour déplacer les vélos entre les stations. Il peut transporter au plus q vélos simultanément.

On se place pendant la nuit ; on suppose donc qu'il n'y a pas d'utilisateurs déplaçant les vélos. Chaque station s a actuellement i_s vélos et l'on souhaite qu'elle en ait f_s . On suppose que $\sum_{s \in S} i_s = \sum_{s \in S} f_s$ et qu'il n'y a pas d'apport externe de vélos. L'objectif de cet exercice est de proposer une méthode qui teste si une tournée fixée du camion permet effectivement

de faire passer le nombre de vélos de i_s à f_s pour chaque station s . Une telle tournée sera qualifiée d'*équilibrante*.

Considérons donc une tournée décrite par la séquence de stations visitées par le camion $s_1, s_2, \dots, s_\ell$. Une même station peut être présente plusieurs fois dans la séquence. Notons n_s le nombre de fois que la station s est présente dans la séquence (on a donc $\sum_{s \in S} n_s = \ell$).

La méthode proposée par cet exercice consiste dans un premier temps à construire le graphe $D = (V, A)$ suivant. On fait de chaque station s un nombre de copies égal à n_s : $s^{(1)}, \dots, s^{(n_s)}$. Si une station n'est pas présente dans la séquence, elle n'apparaît donc pas dans D . On introduit de plus deux sommets fictifs o et d . Les arcs de ce graphe sont les suivants :

- (a) pour tout s présent dans la séquence, on construit un arc $(o, s^{(1)})$ et un arc $(s^{(n_s)}, d)$.
- (b) pour tout couple (s, t) de stations consécutives dans la séquence, on ajoute un arc $(s^{(\alpha)}, t^{(\beta)})$, où α est l'occurrence correspondante de s et β celle de t .
- (c) pour tout s présent dans la séquence et tout entier $k \in \{1, \dots, n_s - 1\}$, on ajoute un arc $(s^{(k)}, s^{(k+1)})$.

Un exemple est donné sur la Figure 2 pour une séquence t, u, v, u, v, t, v, w .

Si s n'est pas présent dans la séquence et si $i_s \neq f_s$, alors la tournée n'est pas équilibrante. Dans la suite, on suppose que si s n'est pas présent dans la séquence, alors $i_s = f_s$.

1. Proposer, pour chaque arc de D , une capacité maximale de façon à ce qu'un algorithme de flot maximum permette de tester si la tournée est équilibrante. On veillera à prouver que si la tournée est équilibrante, alors la flot maximum aura pour valeur $\sum_{s \in S} i_s$, et que réciproquement, si le flot a cette valeur, c'est que la tournée est équilibrante. (4 pts)

On s'intéresse maintenant au problème consistant à trouver la tournée équilibrante nécessitant le moins de visites de stations et on souhaite le résoudre par une recherche locale.

2. Expliquer en quoi la méthode décrite précédemment – pour une tournée fixée donc – peut être utile pour décrire l'espace des solutions d'une telle recherche locale. (1 pt)

5. TOURNÉE LE LONG D'UNE VOIE FERRÉE – 9 PTS

On considère une voie ferrée sur laquelle a lieu des opérations de maintenance. Il n'y a pas de trafic. n points distincts sur la voie doivent subir une opération à l'aide d'une machine. Les points sont identifiés par leur abscisse sur la voie et numérotés par ordre croissant des abscisses : $p_1 < p_2 < \dots < p_{n-1} < p_n$. On suppose que la machine est initialement située en p_1 .

Chaque point p_i doit être *traité* au plus tôt à un instant $r_i \geq 0$. Cela signifie simplement que la machine doit être passée en p_i à un instant $t \geq r_i$. Elle peut y passer avant, mais cela ne permet pas de traiter le point p_i .

Le temps mis pour la machine à passer d'un point x à un point y de la voie est égal à $|x - y|$. On néglige le temps de l'opération, mais la machine est autorisée à attendre en un point p_i , ce qui peut être utile si la machine arrive avant l'instant r_i en p_i .

On veut pouvoir tester s'il existe un parcours de la machine satisfaisant ces contraintes, et si oui, trouver celui de durée totale minimale. On suppose que le parcours commence à

l'instant $t = 0$.

1. Si, pour tout i , on a $r_i = 0$, expliquer pourquoi la solution optimale est simple. Indiquer la formule simple donnant la durée totale de cette solution optimale. (1 pt)

On suppose maintenant les r_i quelconques, mais que l'on a $0 \leq r_1 \leq r_2 \leq \dots \leq r_n$.

2. Montrer que la solution optimale est encore simple. (2 pts)

3. Proposer une formule donnant la valeur de l'instant au plus tôt auquel on peut avoir traité l'ensemble des points. (1 pt)

Dans la suite, on ne suppose plus que $0 \leq r_1 \leq r_2 \leq \dots \leq r_n$. La structure des solutions optimales est alors beaucoup plus complexe.

Pour des indices j, k tels que $1 \leq j < k \leq n+1$, on définit $\pi(j, k)$ comme étant l'instant au plus tôt ayant permis de traiter l'ensemble des points dans $\{p_1, \dots, p_j\} \cup \{p_k, \dots, p_n\}$, tout en étant en p_j . Pour $k = n+1$, la quantité $\pi(j, n+1)$ est donc l'instant au plus tôt ayant permis de traiter l'ensemble des points dans $\{p_1, \dots, p_j\}$ tout en étant en j .

De même, pour des indices j, k tels que $0 \leq j < k \leq n$, on définit $\lambda(j, k)$ comme étant l'instant au plus tôt ayant permis de traiter l'ensemble des points dans $\{p_1, \dots, p_j\} \cup \{p_k, \dots, p_n\}$, tout en étant en p_k . Pour $j = 0$, la quantité $\lambda(0, k)$ est donc l'instant au plus tôt ayant permis de traiter l'ensemble des points dans $\{p_k, \dots, p_n\}$ tout en étant en k .

Pour des commodités d'écriture, on définit de plus $\pi(0, k) = \lambda(j, n+1) = +\infty$ pour tous $1 \leq j < k \leq n$, ainsi que $p_0 = -\infty$ et $p_{n+1} = +\infty$.

4. Montrer que pour tous indices j et k tels que $1 \leq j < k \leq n$, on a

$$\pi(j, k) \leq \max(r_j, p_j - p_{j-1} + \pi(j-1, k))$$

et

$$\pi(j, k) \leq \max(r_j, p_k - p_j + \lambda(j-1, k)).$$

(2 pts)

5. Montrer que pour tous indices j et k tels que $1 \leq j < k \leq n$, on a en fait

$$\pi(j, k) = \min(\max(r_j, p_j - p_{j-1} + \pi(j-1, k)), \max(r_j, p_k - p_j + \lambda(j-1, k))).$$

(1 pt)

On peut montrer par une méthode semblable que

$$\lambda(j, k) = \min(\max(r_k, p_k - p_j + \pi(j, k+1)), \max(r_k, p_{k+1} - p_k + \lambda(j, k+1))).$$

6. En conclure que le problème peut être résolu en $O(n^2)$. (1 pt)

7. Comment peut-on appeler ce type de méthode employée ? (1 pt)

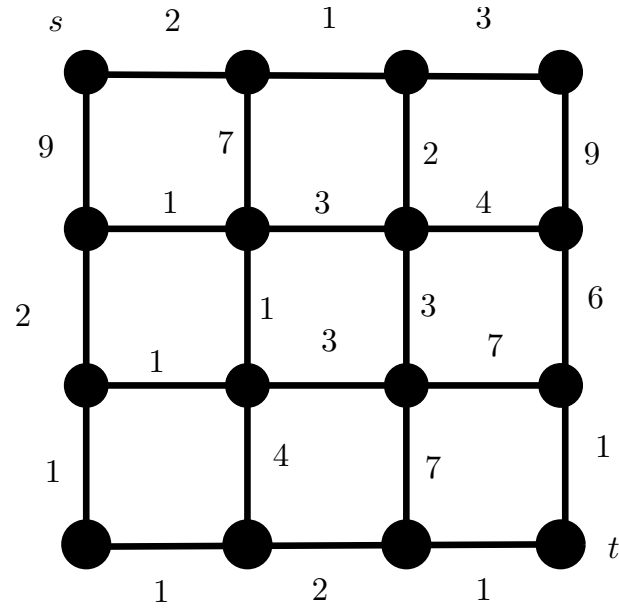
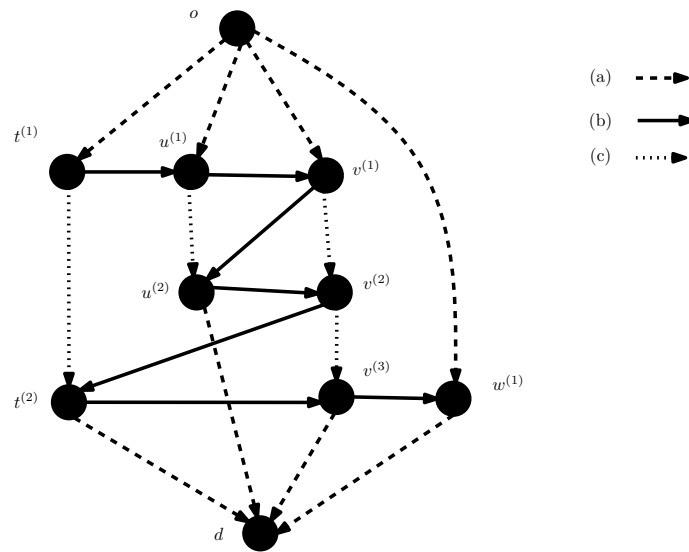


FIGURE 1. Le réseau de l'exercice 1


 FIGURE 2. Construction du graphe de l'exercice 4 pour une séquence t, u, v, u, v, t, v, w