

TD n°11

Flots dans les réseaux et applications

1 Chemins disjoints

Deux chemins sont arête disjoints s'il n'existe pas d'arête utilisée par les deux chemins, et sommet disjoints si les deux chemins n'ont aucun sommet commun, sauf leurs extrémités.

Exercice 1 Donnez un algorithme basé sur les flots qui détermine le nombre maximum de chemins de s à t qui sont arête disjoints.

Exercice 2 Donnez un algorithme basé sur les flots qui détermine le nombre maximum de chemins de s à t qui sont sommet disjoints.

2 Ordonnancement

Le problème d'ordonnancement de J tâches sur M processeurs identiques a les paramètres suivants. Chaque tâche a

- un temps de traitement p_i ,
- une date d'arrivée r_i (date à partir de laquelle on peut commencer le traitement) ,
- et une date de péremption (date avant laquelle la tâche doit être complétée) d_i .

Exercice 3 Montrez comment on peut modéliser le problème d'ordonnancement des tâches par un problème de flot dans un réseau. Indication : Il faut un sommet pour chaque tâche, un sommet pour chaque intervalle de temps, et relier une tâche à un intervalle si la tâche peut être exécutée dans cet intervalle. Il faut ensuite relier toutes les tâches à un sommet source s , et toutes les destinations à un sommet destination t .

1. Quelles sont les capacités des arêtes ?
2. Quel flot peut-on faire transiter sur le réseau s'il existe une solution au problème d'ordonnancement ?
3. Inversement, comment retrouver la solution au problème d'ordonnancement si un tel flot existe ?

3 Flots multi-source, multi-destination

Exercice 4 Supposons qu'une entreprise possède plusieurs usines et plusieurs entrepôts et elle doit faire transiter les marchandises dans un réseau vers les entrepôts. La capacité sur une arête (u, v) du réseau est limitée à $w_{u,v}$. Chaque usine i possède un stock de marchandises s_i qu'elle souhaite envoyer vers les entrepôts. On peut envoyer les marchandises vers n'importe quel entrepôt du moment qu'on n'en excède pas la capacité. L'usine j a une capacité de stockage disponible c_j . Montrer comment réduire ce problème à un problème de flot dans un réseau.

4 Flot de coût minimum

On considère maintenant le problème suivant. Dans un graphe orienté, on veut faire passer le plus de flot possible de e à s . Les arcs ont tous capacité 1. Toutefois, chaque arc (u, v) a un coût (en euros, disons) $prix(u, v) \geq 0$ que l'on doit payer si le flot utilise cet arc. En e et chaque arc est utilisé à 100%, ou à 0%. Le coût d'un flot est donc la somme des coûts des arcs utilisés. On veut faire passer le plus de flot possible, avec comme objectif secondaire de minimiser le coût.

Exercice 5 Si tous les coûts sont de 0, quelle est la solution au problème?

Exercice 6 Soit C une charge de flot, plus petite ou égale à la charge d'un flot maximum. On veut faire un algorithme qui calcule un flot de charge C entre e et s , de coût minimum.

Si $C = 1$ proposez un algorithme.

Exercice 7 Supposons que l'on ai déjà un flot de charge C . Proposer un algorithme, à base de chemin améliorant, qui

- soit propose un flot de charge $C + 1$, tel que l'augmentation de coût soit la moins cher possible
- soit dit que c'est impossible : C est le flot maximum

Exercice 8 En déduire un algorithme qui calcule le flot maximum de coût minimum. Quelle est sa complexité?

Exercice 9 Et si les capacités ne sont plus unitaires, mais quelconque? Un arc a donc un coût $prix(u, v)$ et une capacité $\omega(u, v)$. Peut-on étendre l'algorithme?