Algorithmique et complexité

TD 3/7 – Graphes à flots

Exercices complémentaires

Remarque: Les éléments de correction fournis dans le sujet ne sont pas forcément complets. Ils sont là pour vous guider dans votre travail personnel. Nous vous invitons à rédiger une solution comme vous le feriez à l'examen et, si vous avez des questions, à vous tourner vers votre chargé de TD.

Exercice 1 : Routage

On considère un serveur de stockage S connecté à un terminal T par un réseau avec quatre nœuds A, B, C, D. Les capacités des connexions entre les nœuds sont données, en Mbit/s dans le tableau ci-dessous.

	A	B	C	D	T
S	2	6	4		
A		3		7	
B				2	5
C		1			
D	3				6

L'utilisateur du terminal T doit télécharger un fichier de grande taille du serveur S. Par quelles connexions du réseau ce fichier doit-il être transmis pour que le temps d'attente de l'utilisateur soit le plus court ?

Question 1

A quel problème correspond ce problème de routage ? Quel algorithme de résolution pouvez-vous utiliser pour trouver une solution à ce problème ?

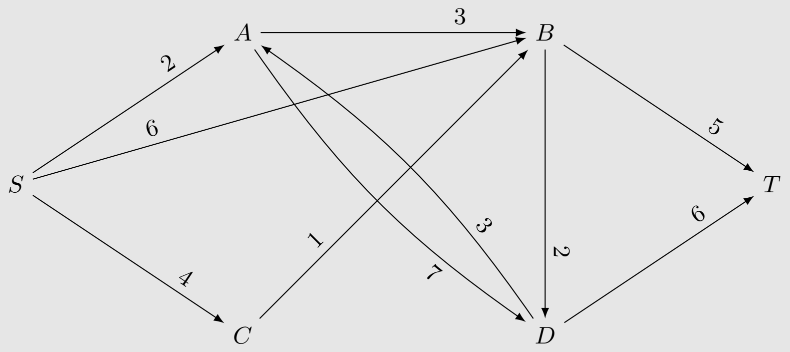
Élements de correction :

- Problème d'optimisation : problème de flot maximum dans un graphe de flot (réseau).
- Algorithme de Ford Fulkerson.

Question 2

Modéliser cette instance du problème et appliquer cet algorithme de résolution en donnant les étapes de son application. Quelle est la coupe minimale ?

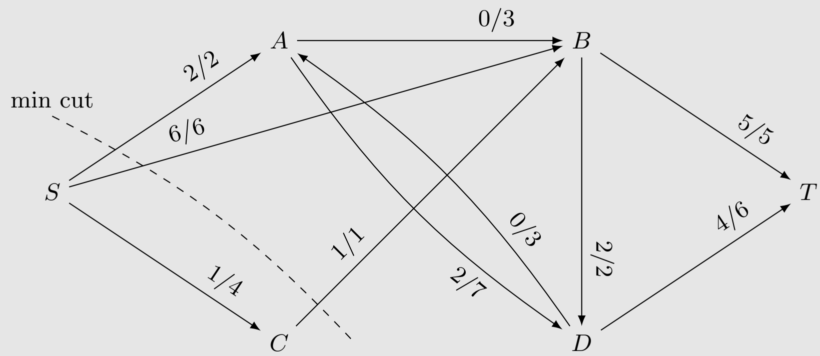
Élements de correction :



on applique par exemple un parcours en profondeur à chaque itération de l'algo et on trouve les chaînes augmentantes suivantes :

— $S \to A \to B \to T : 2$
— $S \to B \to T : 3$
— $S \to B \to D \to T : 2$
— $S \to B \leftarrow A \to D \to T : 1$
— $S \to C \to B \leftarrow A \to D \to T : 1$
— le dernier appel visite S et C et puis s'arrête.

Le flot max est 9, et la coupe minimale est $\{S, C\}$, et on obtient le graphe de flot final :



Question 3

Quel est le débit maximal ? Combien de temps prendra la transmission d'un fichier de taille de 100 Mo ($1Mo = 2^{23}bits$) ?

Élements de correction :

Flot max = 9, donc le débit maximal est $9Mbits/s$ d'après la question précédente. La transmission de ce fichier prendra :

$$\frac{100 \times 2^{23}[\text{bits}]}{9 \cdot 2^{20}[\text{bits/s}]} = 88{,}8 \text{ [s]}.$$

Exercice 2 : Couper un graphe en deux

Soit $G = (V, E)$ un graphe non-orienté connexe et deux sommets s et p du graphe.

Un sous-ensemble d'arêtes $F \subseteq E$ est un (s, p)**-séparateur** si toutes les chaînes entre s et p passent forcément par une arête de F.

Soit P un ensemble de chaînes reliant s à p. On peut dire que les chaînes de P sont **disjointes** (en anglais, *edge-disjoint*) si aucune paire de chaînes (P_i, P_j) avec $i \neq j$ n'a d'arêtes en commun.

Théorème de Menger : Le nombre maximal des chaînes disjointes entre s et p dans G est égal à la taille du plus petit (s, p)-séparateur de G.

Question 1

Démontrer le théorème de Menger.

Élements de correction :

Il faut transformer le graphe G en un graphe à flot puis utiliser l'égalité entre coupe-min et flot-max vue en cours.

Pour transformez votre graphe G, vous allez transformer chaque arête $\{u, v\}$ en deux arcs (u, v) et (v, u) sauf celles qui partent de s ou bien celles qui arrivent à p, celles-ci donnent lieu à un seul arc orienté dans ≪ le bon sens ≫. **Rédigez proprement cette transformation !**

On fixe la capacité de chaque arc à 1. On a alors :

- Le nombre maximum de chaînes *edge-disjoint* entre s et p est égal au flot maximum, puisque toute les capacités sont fixées à 1 (chaque arête ne peut donc être prise qu'une foie).
- La taille du plus petit (s, p)-séparateur dans G est alors égale à la capacité de la (s, p)-coupe minimale dans G'.

En utilisant l'égalité entre *min-cut* et *max-flow* vue en cours, on obtient :

$$\begin{aligned} \text{nombre max de chaînes } \textit{edge-disjoint} \text{ entre } s \text{ et } p &= \text{flot max} \\ &= \text{coupe min} \\ &= \text{taille du plus petit } (s,p)\text{-séparateur.} \end{aligned}$$