Graphes de flots

Un graphe de flot est une structure permettant de modéliser tout types de flux : réseau de transport routier, circulation d'eau ou de fluides dans des canalisations, distribution d'électricité, ... Sa définition ressemble à celle d'un graphe pondéré, où les poids seront considérés comme des capacités de débit.

1 Graphe de flot et flot maximal

Définition

On appelle **graphe de flot** un quintuplet $G = (S, A, c, s, t)$ où :
- (S, A) est un graphe orienté sans boucle ;
- $c : A \to \mathbb{R}_+$ est une fonction dite de **capacité** ;
- $s \in S$ est une **source** de (S, A) et $t \in S$ est un **puits** de (S, A).

Par convention, si $(u, v) \in S^2 \setminus A$, on pose $c(u, v) = 0$.

La figure 1 représente un graphe de flot G_0. Les capacités sont représentées sur les arêtes.

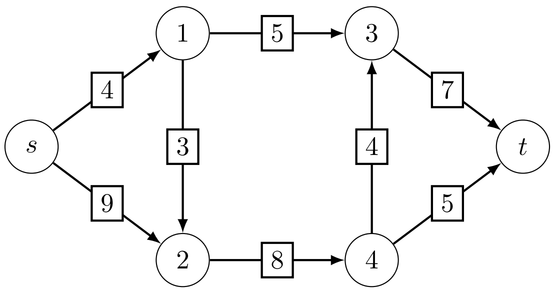

Figure 1 – Le graphe de flot G_0.

Définition

Soit $G = (S, A, c, s, t)$ un graphe de flot. On appelle **flot** une fonction $f : S^2 \to \mathbb{R}$ vérifiant les propriétés suivantes :
- antisymétrie : pour $(u, v) \in S^2$, $f(u, v) = -f(v, u)$;
- respect de la capacité : pour $(u, v) \in S^2$, $f(u, v) \leqslant c(u, v)$;
- conservation : pour $u \in S \setminus \{s, t\}$, $\sum\limits_{v \in S} f(u, v) = 0$.

On appelle **débit** de f la valeur $|f| = \sum\limits_{u \in S} f(s, u)$.

La figure 2 représente le graphe de flot G_0 et un flot f compatible avec G. Pour chaque arête (u, v), on représente sur l'arête $f(u, v)/c(u, v)$. Les autres valeurs nulles et négatives de f peuvent se déduire des valeurs positives.

Question 1 Existe-t-il un flot f de G_0 de débit 13 ? Justifier.

Pour la suite du sujet, on considère $G = (S, A, c, s, t)$ un graphe de flot et f un flot de G.

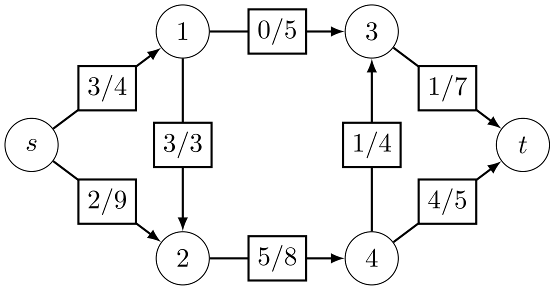

FIGURE 2 – Le graphe de flot G_0 et un flot f.

Définition

Si $u \in S$, on appelle :

- **flux sortant de** u la valeur $\phi_+(u) = \sum_{v \in S, f(u,v)>0} f(u,v)$;
- **flux entrant de** u la valeur $\phi_-(u) = \sum_{v \in S, f(v,u)>0} f(v,u)$;
- **flux net de** u la valeur $\phi(u) = \phi_+(u) - \phi_-(u)$.

Question 2 Montrer que la condition de **conservation** du flot f est équivalente à la propriété suivante :

$$\text{Pour tout } u \in S \setminus \{s,t\}, \phi(u) = 0$$

Question 3 Montrer que $|f| = \phi(s) = -\phi(t)$.

On considère le problème d'optimisation `Flot maximal` :

* **Instance** : un graphe de flot $G = (S, A, c, s, t)$.
* **Solution** : un flot f de G.
* **Optimisation** : Maximiser $|f|$.

Question 4 Représenter graphiquement une solution à `Flot maximal` sur le graphe G_0 de la figure 1. Justifier qu'il s'agit bien d'une solution maximale.

2 Algorithme de Ford-Fulkerson

Définition

On dit qu'une arête $(u,v) \in A$ est **saturée** si $f(u,v) = c(u,v)$. On dit qu'un chemin de s à t est **saturé** s'il contient une arête saturée.

Si un chemin de s à t n'est pas saturé, on définit l'action de **saturation du chemin** comme une modification de f qui consiste, pour chaque arête (u,v) du chemin, à augmenter $f(u,v)$ de la même valeur (et diminuer ($f(v,u)$ d'autant), jusqu'à ce que l'une d'entre elle soit saturée. La figure 3 montre le résultat de la saturation du chemin $(s,2,4,3,t)$ dans le graphe G_0 à partir du flot f. On a augmenté le débit de 3 le long du chemin. Les arêtes saturées sont $(2,4)$ et $(4,3)$.

On considère l'algorithme glouton suivant :

Entrée : Graphe $G = (S, A, c, s, t)$ graphe de flot
Début algorithme
  Poser $f(u,v) = 0$ pour tout $(u,v) \in S^2$
  Tant que Il existe un chemin non saturé de s à t **Faire**
    Saturer ce chemin
  Renvoyer f

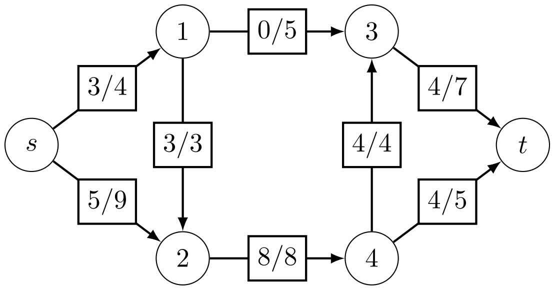

FIGURE 3 – Le graphe de flot G_0 et le flot f après saturation du chemin $(s, 2, 4, 3, t)$.

Question 5 Expliquer comment trouver un chemin non saturé de s à t dans un graphe de flot, étant donné un flot f donné.

Question 6 Terminer l'exécution de l'algorithme sur le graphe G_0 à partir du flot f de la figure 3 et représenter graphiquement le résultat.

La question précédente et la question 4 montrent que cet algorithme glouton ne renvoie pas toujours un flot maximal. Pour corriger le problème, il faut envisager de pouvoir faire « refluer » le flot en arrière dans une arête.

Définition

On définit le **graphe résiduel** $G_f = (S, A_f)$ où $A_f = \{(u, v) \in S^2 \mid c(u, v) - f(u, v) > 0\}$. Un chemin de s à t dans G_f est appelé **chemin améliorant pour** f.

Attention, le graphe G_f peut contenir des arêtes n'existant pas dans G.

Question 7 Représenter graphiquement le graphe résiduel de G_0 avec le flot obtenu à la question 6.

L'algorithme de Ford-Fulkerson est alors le suivant :

Entrée : Graphe $G = (S, A, c, s, t)$ graphe de flot
Début algorithme
  Poser $f(u, v) = 0$ pour tout $(u, v) \in S^2$.
  Tant que Il existe un chemin améliorant pour f **Faire**
    Saturer ce chemin.
  Renvoyer f

Question 8 Terminer l'exécution de l'algorithme de Ford-Fulkerson sur le graphe G_0 avec le flot obtenu à la question 6.

On représente en OCaml un graphe de flot $G = (S, A, c, s, t)$ à capacités entières par un couple formé d'un tableau de listes d'adjacence et d'une matrice d'entiers donné par le type suivant :

```ocaml
type graphe_flot = int list array * int array array;;
```

Ainsi, si `(g, c)` est une variable de type `graphe_flot` correspondant à G, alors :

- $S = [\![0, n-1]\!]$ où $n = |S|$;
- $s = 0$ et $t = n - 1$;
- pour $u \in S$, `g.(u)` contient la liste des voisins de u ;
- pour $u, v \in S^2$, `c.(u).(v)` est égal à $c(u, v)$.

On supposera que pour chaque arête (u, v), il existe l'arête (v, u), quitte à ce qu'elle soit de capacité 0.

Un flot f de G sera représenté par une matrice d'entiers de type :

```ocaml
type flot = int array array;;
```

telle que si `f` est une variable représentant f et $(u, v) \in S^2$, alors `f.(u).(v)` est égal à $f(u, v)$.

Question 9 Écrire une fonction `compatible : flot -> graphe_flot -> bool` qui vérifie si un flot donné en argument est compatible avec un graphe donné en argument (donc vérifie les trois propriétés de la définition d'un flot). On supposera que les dimensions des tableaux sont identiques.

Question 10 Écrire une fonction `chemin_ameliorant : flot -> graphe_flot -> int list` qui prend en argument un flot f et un graphe de flot G et renvoie une liste d'entiers correspondant à un chemin améliorant. Cette fonction renverra une liste vide s'il n'existe pas de tel chemin. Cette fonction devra avoir une complexité en $\mathcal{O}(|S| + |A|)$.

Question 11 En déduire une fonction `ford_fulkerson : graphe_flot -> flot` qui renvoie un flot maximal selon l'algorithme de Ford-Fulkerson.

Question 12 Montrer que si les capacités sont entières, l'algorithme de Ford-Fulkerson termine toujours et déterminer sa complexité temporelle.

3 Flot maximal, coupe minimale

Définition

Soit $G = (S, A, c, s, t)$ un graphe de flot. On appelle **coupe** de G un ensemble $X \subset S$ tel que $s \in X$ et $t \in \overline{X}$. La **capacité** d'une coupe X est $C(X) = \sum\limits_{(u,v) \in X \times \overline{X}} c(u, v)$.

Question 13 Dans le graphe G_0 donné figure 1, déterminer la capacité de la coupe $X = \{s, 1, 3\}$.

Question 14 Soit f un flot et X une coupe de G. Montrer que $|f| = \sum\limits_{(u,v) \in X \times \overline{X}} f(u, v) \leqslant C(X)$.

Question 15 Montrer l'équivalence entre les propriétés suivantes :

1. f est un flot maximal ;
2. il n'existe pas de chemin améliorant pour f ;
3. il existe une coupe X telle que $|f| = C(X)$.

Question 16 En déduire la correction de l'algorithme de Ford-Fulkerson s'il termine.

4 Réduction du couplage maximum

Question 17 Expliquer comment résoudre le problème du couplage maximum dans un graphe non orienté biparti en trouvant un flot maximal dans un graphe de flot dont on détaillera la construction. Justifier la correction et déterminer la complexité temporelle d'un tel algorithme.

⋆ ⋆ ⋆