Couplage maximum dans un graphe biparti

Quentin Fortier

December 1, 2022

Couplage

Dans ce cours, $G = (V, E)$ est un graphe non orienté et non pondéré.

Définition

Un **couplage** de G est un ensemble d'arêtes $M \subseteq E$ tel qu'aucun sommet ne soit adjacent à 2 arêtes de M, c'est-à-dire :

$$\forall e_1, e_2 \in M,\ \ e_1 \neq e_2 \implies e_1 \cap e_2 = \emptyset$$

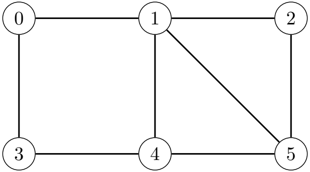
Définition

Un sommet $v \in V$ est **couvert** par M s'il appartient à une arête de M. Sinon, v est **libre** pour M.

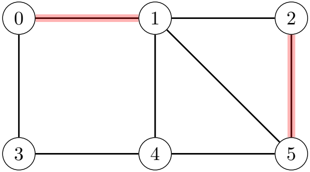
Applications :

- SCEI : associe à chaque élève au plus une affectation
- Mariage : chaque personne est mariée à au plus une autre personne

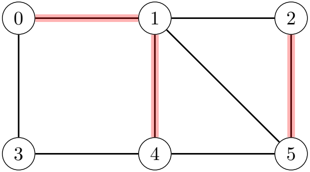
Couplage



Un graphe G



Un couplage de G (en rouge)



Pas un couplage

Couplage

Exercice

Écrire une fonction
`est_couplage : int array array -> (int*int) list -> bool`
déterminant si un ensemble d'arêtes forme un couplage d'un graphe.

```
let est_couplage g m =
let n = Array.length g in
let libre = Array.make n false in (* libre.(v) = sommet déjà vu *)
let rec aux m =
    match m with
    |[] -> true
    |(x, y)::q -> if g.(x).(y) = 1 && libre.(x) && libre.(y)
                then begin
                    libre.(x) <- false ;
                    libre.(y) <- false ;
                    aux q
                    end
                else false
    in
    aux m
```

Couplage

Soit M un couplage d'un graphe G.

Définitions

- La **taille** de M, notée $|M|$, est son nombre d'arêtes.
- M est un couplage **maximum** s'il n'existe pas d'autre couplage de taille strictement supérieure.
- M est un couplage **maximal** s'Il n'existe pas de couplage M' tel que $M \subsetneq M'$.
- M est un couplage **parfait** si tout sommet de G appartient à une arête de M.

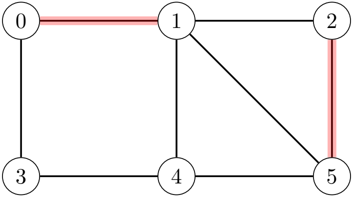
Question

Quelle(s) implication(s) a t-on entre couplage maximum et couplage maximal ?

Couplage

Exercice

1. Le couplage ci-dessous est-il parfait ?
2. Quels sont les sommets couverts par ce couplage ? Et ceux libres ?
3. Le graphe ci-dessous admet-il un couplage parfait ?



Couplage

On va s'intéresser au problème suivant :

Problème : Couplage maximum

Entrée : Graphe G non orienté, non pondéré.
Sortie : Un couplage maximum de G.

Chemin augmentant

Soit M un couplage d'un graphe G.

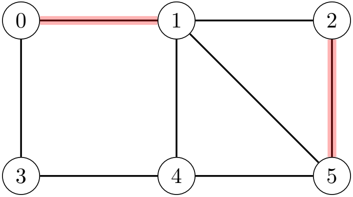
Définition

- Un chemin de G est M-**alternant** si ses arêtes sont alternativement dans M et dans $E \setminus M$.
- Un chemin de G est M-**augmentant** s'il est M-alternant et si ses extrémités sont libres pour M.

Question

Donner un exemple de chemin augmentant pour le couplage ci-dessous.

c = [4, 5, 2, 1, 0, 3]



Chemin augmentant

Définition

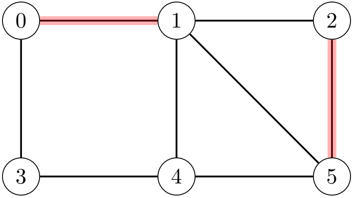
Si A et B sont des ensembles, $A \Delta B = (A \setminus B) \cup (B \setminus A)$.

Un chemin est vu comme un ensemble d'arêtes.

Théorème

Soit M un couplage de G et P un chemin M-augmentant dans G. Alors $M \Delta P$ est un couplage de G.

Exemple : Dessiner $M \Delta P$ pour le couplage ci-dessous et le chemin $P = 3 - 0 - 1 - 4$.



Démonstration
Si $G = (V, E)$, soit $v \in V$
Supposons que :

- $v \in e_1$ où $e_1 \in M \Delta P$
- $v \in e_2$ où $e_2 \in M \Delta P$

On ne peut pas avoir $e_1 \in M$ et $e_2 \in M$ (car M couplage) ni $e_1 \in P \setminus M$ et $e_2 \in P \setminus M$ (car ce ne serait pas alternant). Par symétrie, supposons $e_1 \in M \setminus P$ et $e_2 \in P \setminus M$

1. Si v est extrémité de P: **absurde** car v est libre (P augmentant) mais v adjacent à $e_1 \in M$
2. $\exists e_3 \neq e_2 \in P$ adjacent à v, P est augmentant et $e_2 \notin M$ donc $e_3 \in M$ **absurde** car v adjacent à 2 arêtes de M

Chemin augmentant

Soit M un couplage d'un graphe G.

Théorème

M est un couplage maximum de G

$\Longleftrightarrow$

Il n'existe pas de chemin M-augmentant dans G

Preuve :

$\Longrightarrow$ Soit M un couplage maximum.
Supposons qu'il existe un chemin M-augmentant P.
Alors $M \Delta P$ est un couplage de G et $|M \Delta P| > |M|$: absurde.

Chemin augmentant

Théorème

M est un couplage maximum de G

$$\Longleftrightarrow$$

Il n'existe pas de chemin M-augmentant dans G

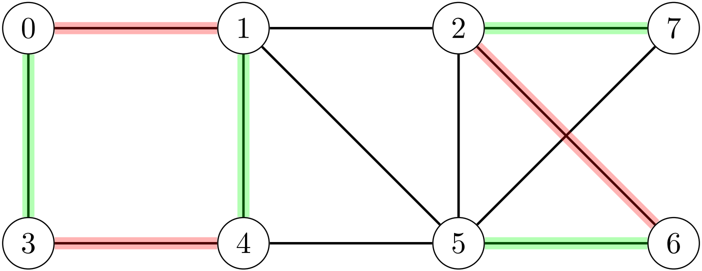
Preuve :

$\Longleftarrow$ Supposons qu'il existe un couplage M^* vérifiant $|M^*| > |M|$. Considérons $G^* = (V, M \Delta M^*)$. Alors :

1. Les degrés des sommets de G^* sont au plus 2, donc G^* est composé de cycles et de chemins uniquement.
2. Chacun de ces cycles et chemins alternent entre des arêtes de M et des arêtes de M^*.
3. Comme $|M^*| > |M|$, un de ces chemins contient plus d'arêtes de M^* que de M: c'est un chemin M^*-augmentant.

Chemin augmentant

Illustration de la preuve précédente :



Un couplage M et un couplage maximum M^*

Chemin augmentant

Couplage maximum par chemin augmentant

Entrée : Graphe $G = (V, E)$
Sortie : Couplage maximum M de G

$M \leftarrow \emptyset$
Tant que il existe un chemin M-augmentant P dans G **:**
  $M \leftarrow M \Delta P$

Question

Comment trouver un chemin M-augmentant ?

- Dans un graphe quelconque, avec le Blossom algorithm (très compliqué et HP).
- Plus facilement dans un graphe **biparti**.

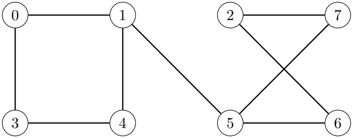
Graphe biparti

Définition

Un graphe $G = (V, E)$ est **biparti** s'il existe une partition $V = A \sqcup B$ telle que toute arête de E a une extrémité dans A et une extrémité dans B.

Question

Montrer que le graphe ci-dessous est biparti, en donnant une partition de ses sommets.



Graphe biparti

Définition équivalente :

Définition

On appelle k-**coloration** de G une fonction $c : V \longrightarrow \{1, 2, \dots, k\}$ telle que pour tout arc $(u, v) \in E$, on a $c(u) \neq c(v)$.

Lemme

G admet une 2-coloration

$$\Longleftrightarrow$$

G est biparti

Graphe biparti

Exercice

Écrire une fonction `est_biparti : int list array -> bool` pour déterminer si un graphe est biparti, en complexité linéaire.

Exercice

Modifier la fonction précédente pour renvoyer un 2-coloriage.

```
let est_biparti g =
    let n = Array.length g in
    let color = Array.make n (-1) in
    (* color.(v) sera la couleur du sommet v *)
    let rec aux c v =
        if color.(v) = -1 then (
            color.(v) <- c;
            List.for_all (aux (1 - c)) g.(v);
        )
        else color.(v) = c
```

Graphe biparti

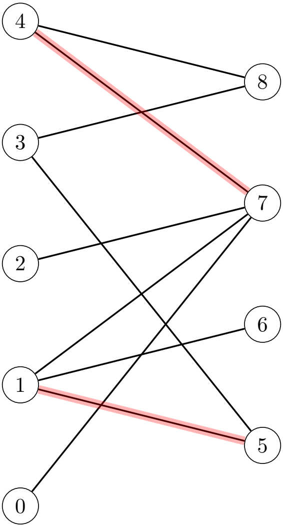
Soit $G = (V, E)$ un graphe biparti, avec $V = A \sqcup B$, et M un couplage de G.
On définit $\overrightarrow{G_M} = (V_M, \overrightarrow{E_M})$ où :

- $V_M = V \cup \{s, t\}$, où s et t sont deux nouveaux sommets.
- $\overrightarrow{E_M} = \{(s, u) \mid u \in A$ et u est libre$\} \cup \{(v, t) \mid v \in B$ et v est libre$\} \cup \{(u, v) \mid \{u, v\} \in E \setminus M\} \cup \{(v, u) \mid \{u, v\} \in M\}$.

Autrement dit :

- On ajoute deux nouveaux sommets s et t.
- On mets des arcs depuis s vers chaque sommet libre de A.
- On mets des arcs depuis chaque sommet libre de B vers t.
- On oriente les arcs de M de B vers A.
- On oriente les arcs de $E \setminus M$ de A vers B.

Graphe biparti



Graphe biparti

Théorème

$$\vec{P} \text{ est un chemin de } s \text{ à } t \text{ dans } \overrightarrow{G_M}$$
$$\iff$$
$$P \cap E \text{ est un chemin } M\text{-augmentant dans } G$$
(où P est obtenu à partir de P en enlevant les orientations)

Il suffit donc de trouver un chemin de s à t dans $\overrightarrow{G_M}$ pour trouver un chemin M-augmentant dans G.

Complexité :

1. Construction de $\overrightarrow{G_M}$: $\mathrm{O}(|V| + |E|)$.
2. Recherche d'un chemin de s à t dans $\overrightarrow{G_M}$: $\mathrm{O}(|V| + |E|)$ (DFS ou BFS).

Total : $\boxed{\mathrm{O}(|V| + |E|)}$.

Graphe biparti

Couplage maximum par chemin augmentant

Entrée : Graphe $G = (V, E)$
Sortie : Couplage maximum M de G

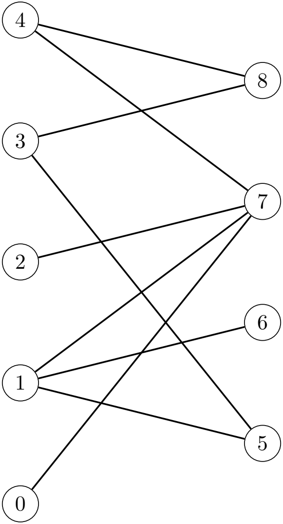
$M \leftarrow \emptyset$
Tant que il existe un chemin M-augmentant P dans G **:**
   $M \leftarrow M \Delta P$

Complexité :
Il y a au plus $|E|$ d'itération du « Tant que », car on ajoute une arête au couplage à chaque fois.
Sur un graphe biparti, on obtient alors une complexité $\mathrm{O}(|E|(|V| + |E|))$ ($= \boxed{\mathrm{O}(|V||E|)}$ si G est supposé connexe).

Graphe biparti

Question

Appliquer l'algorithme précédent au graphe ci-dessous.



Graphe biparti

Exercice (TP pendant les vacances)

Écrire une fonction `couplage_max` telle que, si `g` est un graphe biparti représenté par liste d'adjacence et `a`, `b` partitionne les sommets de `g`, alors `couplage_max g a b` renvoie un couplage maximal de `g`.