INF581: Du calcul probabiliste au calcul quantique	**Hiver 2011**
Cours 5 — 21 octobre	
Enseignant : Frédéric Magniez	*Rédacteur : Gaspard FEREY*

5.1 Max Cardinality Graph Matching

Problème :

> **Input Stream:** Arêtes d'un graphe à n sommets (n connu) dans un ordre arbitraire.
> Exemple : $(2,3),(1,4),(5,2),...$
>
> **Output:** Couplage M de plus grande taille possible.

Mémoire : $O(n \times \text{polylog}(n)) = O(n \times \log^{O(1)}(n))$

Définition 5.1. *M est une C-approximation de Max Cardinality Graph Matching si $|M| \geq C \times |M^*|$ où M^* est un couplage de plus grande taille du graphe.*

Théorème 5.2. *Il existe un algorithme déterministe à 1 passe, de mémoire $O(n \log n)$ qui renvoie une $\frac{1}{2}$-approximation de Max Cardinality Graph Matching.*

Théorème 5.3. (2012) *Il n'existe pas d'algorithme en 1 passe et mémoire $O(n \times \text{polylog}(n))$ qui retourne mieux qu'une $\frac{2}{3}$-approximation (même probabiliste).*

Théorème 5.4. (2013) *Idem : pas mieux qu'une $(1-1/e)$-approximation.*

Remarque : Cette borne reste vraie dans le modèle "vertex arrival order".

Algorithme Glouton

```
M ← ∅
Tant que Stream non vide
    e ← lire arête du Stream
    Si M ∪ {e} est encore un couplage
        M ← M ∪ {e}
Retourner M
```

Mémoire : $O(n \log n)$ car $|M| \leq \frac{n}{2}$.

Lemme 5.5. *M est maximal pour l'inclusion.*
C'est à dire que $\forall e \in G$:

- *Soit $e \in M$*
- *Soit $M \cup \{e\}$ n'est pas un couplage.*

Preuve: Vrai par construction de M.
On prouve par induction qu'à chaque instant, M est maximal pour les arêtes déjà lues. □

Lemme 5.6. *Soit M^* un couplage de plus grande taille. Soit M un couplage maximal pour l'inclusion. Alors*

$$|M| > \frac{1}{2}|M^*|$$

Notations On note M^* un couplage de plus grande taille du graphe G.
Pour M couplage quelconque, on note $V(M)$ l'ensemble des sommets couverts par M.

Preuve: (Par charge)
1) Chaque arête de M^* reçoit une charge.
2) Chaque arête deM^* transmet cette charge sur une de ses extrémités dan $V(M)$.

Remarque : C'est toujours possible car pour une arête $e = (a, b) \in M^*$, si ni a, ni b ne sont dans $V(M)$, alors $M \cup \{e\}$ est encore un couplage (absurde par maximalité de M).

On a par définition :

$$\sum_{v \in V(M)} \text{charge}(v) = |M^*|$$

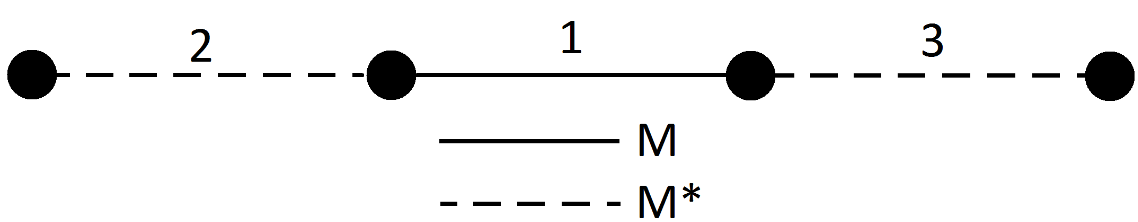
De plus, comme M^* est un couplage, chaque sommet reçoit au plus 1 charge. On en conclut que :

$$|M^*| = \sum_{v \in V(M)} \text{charge}(v) \leq |V(M)| = \frac{1}{2}|M|$$

□

La combinaison des deux lemmes précédents prouve le **Théorème 5.2**.

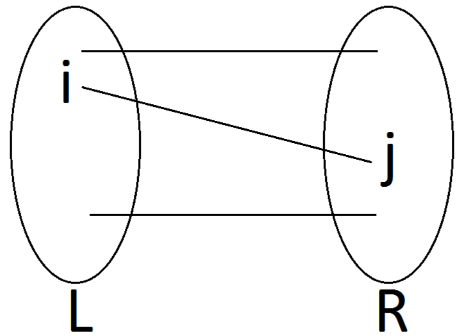
Exemple: G tel que l'algorithme glouton qui ne donne pas mieux qu'une $\frac{1}{2}$-approximation.
$n = 4$



$$|M| = \frac{1}{2}|M^*|$$

5.2 Matching in "vertex arrival order"

Input: Graphe biparti sur n sommets.



Stream: $j \in R$ "arrive" avec la liste des sommets avec qui j est connecté.

Exemple: $(1, \underline{1}), (3, \underline{1}), (17, \underline{1}), (4, \underline{2}), (3, \underline{2}), (17, \underline{3}), (1, \underline{3}), ...$

Théorème 5.7. *Il existe un algorithme probabiliste en 1 passe de mémoire en $O(n \log n)$ qui calcule un couplage M tel que :*

$$\mathbb{E}\left[|M|\right] \geq (1 - \frac{1}{e})|M^*|$$

(où l'espérance porte sur les choix probabilistes de l'algorithme.)

Motivation : "adwords" (online)
L : advertisers
R : key words

Modèle simple: $(i, j) \in G$ si i veut bien payer pour le mot clef j.
i paye au plus une fois.
Les mots clefs "arrivent" (stream), il faut choisir à la volée quelle publicité afficher pour maximiser son gain. On n'affiche qu'un publicité par mot clef.

Modèle plus complexe: i veut bien payer V_i par mot clef.

Modèle adwords: i a en plus un budget B_i et veux bien payer $V_{i,j}$ pour le mot clef j.

Remarque : Tout ces problèmes dérivent du précédent.

Remarque : Pour une détermination "online", on ne peux pas faire mieux qu'une $\frac{1}{2}$-approximation avec l'algorithme précédent.

Algorithme Ranking

Pour tout $i \in L$

  Choisir au hasard un réel $Y_i \in [0,1]$

$M \leftarrow \emptyset$

Tant que Stream non vide

  $j \leftarrow$ lire sommet du Stream (et ses arêtes)

  Trouver i parmi les voisins de j tel que $i \notin V(M)$ et Y_i minimal.

  Si on trouve un tel i

    $M \leftarrow M \cup \{(i,j)\}$

Retourner M

5.2.1 Analyse de 2013

Programmation linéaire pour un couplage de taille maximale.

Intuition: $x_{i,j} = 1$ si $(i,j) \in M$, 0 sinon.

$$\text{Opt} = \max \sum_{(i,j)\in E} x_{i,j}$$

Contraintes:

$$\begin{aligned}
\forall i \in L, \sum_{j|(i,j)\in E} x_{i,j} &\leq 1 \\
\forall j \in R, \sum_{i|(i,j)\in E} x_{i,j} &\leq 1 \\
\forall i \in L, \forall j \in R, x_{i,j} &\geq 0
\end{aligned}$$

La solution maximale n'est pas nécessairement entière.

Lemme 5.8. *$Opt(G) \geq |M^*|$*

Écriture du dual.

Minimisation:

$$\text{Opt} = \min \sum_{i\in L} \alpha_i + \sum_{j\in R} \beta_j$$

Contraintes:

$$\begin{aligned}
\forall i \in L, \alpha_i &\geq 0 \\
\forall j \in R, \beta_j &\geq 0 \\
\forall (i,j) \in E, \alpha_i + \beta_j &\geq 1
\end{aligned}$$

On va montrer que l'algorithme de ranking peut nous fournir une solution $(\alpha_i)_{i\in L}$, $(\beta_j)_{j\in R}$ qui vérifie les propriété suivantes :

Propriété 1 : $\bar{\alpha_i} = \mathbb{E}(\alpha_i), \bar{\beta_j} = \mathbb{E}(\beta_j)$ vérifient les contraintes du dual.
Propriété 2 : $|M| = \left(\sum_{i\in L}\alpha_i + \sum_{j\in R}\beta_j\right)\cdot\left(1-\frac{1}{e}\right)$

Preuve: (Théorème 5.7)
D'après la première propriété

$$\sum_{i\in L}\bar{\alpha_i} + \sum_{j\in R}\bar{\beta_j} \geq \mathrm{Opt}(G) \geq |M^*|$$

D'après la seconde :

$$\mathbb{E}\left[|M[\right] = \left(1-\frac{1}{e}\right)\left(\sum_{i\in L}\bar{\alpha_i} + \sum_{j\in R}\bar{\beta_j}\right)$$

On en conclut

$$\mathbb{E}\left[|M[\right] \geq \left(1-\frac{1}{e}\right)\cdot|M^*|$$

□

Construction des α_i, β_j :
Soient $g : [0,1] \to [0,1]$ et $F \in [0,1]$.
Dans la pratique, on verra qu'il faut choisir $g(y) = e^{y-1}$ et $F = \left(1-\frac{1}{e}\right)$.
Lorsque i et couplé avec j par l'algorithme, on pose :

$$\begin{aligned}\alpha_i &= \frac{1}{F}g(Y_i)\\ \beta_j &= \frac{1}{F}(1-g(Y_i))\end{aligned}$$

Pour i non couvert, on pose $\alpha_i = 0$.
Pour j non couvert, on pose $\beta_j = 0$.

Intuition : i a un budget $\frac{1}{F}$ et il paye une fraction $(1-g(Y_i)$ à j.

Preuve: Montrons (α_i, β_j) solution de la **Propriété 2**.
Pour chaque arête $(i,j) \in M$, $\alpha_i + \beta_j = F$ et si $i \notin V(M)$, $\alpha_i = 0$ et si $j \notin V(M)$, $\beta_j = 0$.
Donc :

$$\left(\sum_{i\in L}\alpha_i + \sum_{j\in R}\beta_j\right) = \frac{1}{F}\cdot|M|$$

□

Preuve: Montrons la **Propriété 2**.
La seule chose à vérifier est :

$$\forall (i,j) \in E, \quad \mathbb{E}[\alpha_i] + \mathbb{E}[\beta_j] \quad \geq \quad 1 \tag{5.1}$$

Fixons (i,j) et notons $Y^{-i} = (Y_k)_{k \neq i}$.
On va montrer (5.1) lorsque uniquement Y_i est aléatoire.
Considérons l'algorithme sur G privé de i.

- Si j est couplé avec i', posons $v = Y_{i'}$ et $\beta_j^{-i} = \frac{1-g(v)}{F}$.
- Si j n'est pas couplé, posons $v = 1$.

Lemme 5.9. *(non prouvé)*

1. *Si $Y_i < v$ alors i est couplé avec quelqu'un.*
2. *On a toujours $\beta_j \geq \beta_j^{-i}$.*

Conséquences :

$$\begin{aligned}
\mathbb{E}_{Y_i}(\beta_j) &\geq \beta_j^{-i} = \frac{1-g(v)}{F} \\
\mathbb{E}_{Y_i}(\alpha_i) &\geq \int_0^v \frac{g(y)}{F}\mathrm{d}y \\
\mathbb{E}_{Y_i}(\alpha_i + \beta_j) &= \frac{1}{F}\left[1 - g(v) + \int_0^v g(y)\mathrm{d}y\right]
\end{aligned}$$

- Si on pose $g(v) = e^{v-1}$, il faut choisir

$$F \leq 1 - e^{v-1} + (e^{v-1} - \frac{1}{e}) = 1 - \frac{1}{e}$$

- Si on pose $g(v) = v$, il faut choisir

$$F \leq 1 - v + \frac{v^2}{2}$$

pour tout v donc $F \leq \frac{1}{2}$.

$\square$

5.3 Correction des exercices

5.3.1 Exercice 1

Input: Stream de n entiers dans $[1, n]$ avec exactement un doublon α (β manquant).

Output: Trouver α.

Contraintes: Mémoire en $O(n \log n)$, une passe.

ALGORITHME SOLUTION

Calculer en une passe :
— $S_1 = \sum_{=1}^{n} a_i$
— $S_2 = \sum_{=1}^{n} a_i^2$

On a :

$$\begin{aligned} S_1 &= \frac{n(n+1)}{2} + \alpha - \beta \\ S_2 &= \frac{n(n+1)(2n+1)}{6} + \alpha^2 - \beta^2 \end{aligned}$$

Retourner l'unique (car $\alpha \neq \beta$) solution du système.
