Algorithmique Avancée 2018 École Centrale-Supélec

PC7 : Algorithmes d'approximations

Christoph Dürr, Nguyễn Kim Thắng

1 Couverture par ensembles

On vous donne un univers $U = \{u_1, \ldots, u_n\}$ et m ensembles $S_1, \ldots, S_m \subseteq U$ ainsi qu'une pondération $w_i \in \mathbb{N}$ pour tout chacun des ensembles S_i. Une *couverture par ensembles* est une collection $\mathcal{C}$ de ces ensembles telle que $\bigcup_{S_i \in \mathcal{C}} S_i = U$. L'objectif est de trouver une couverture telle que le poids total $\sum_{i:S_i \in \mathcal{C}} w_i$ soit minimal.
Ce problème est NP-complet. Nous allons chercher un algorithme d'approximation. Considérez l'algorithme suivant.

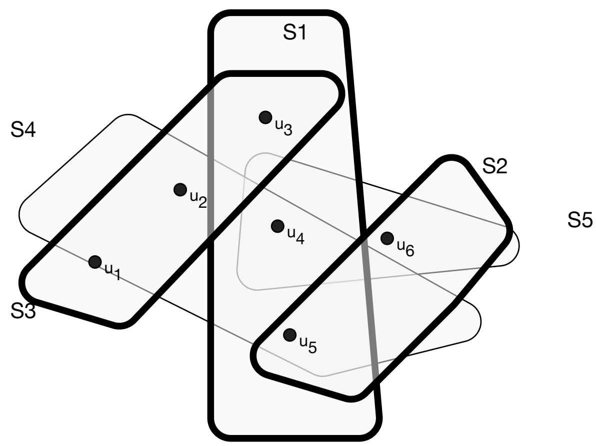

Figure 1: En gras une solution possible

Algorithm 1 Algorithme glouton pour Couverture des ensembles.

Initialement, $R \leftarrow U$ and $\mathcal{C} \leftarrow \emptyset$.
while $R \neq \emptyset$ **do**
  Soit S_i un ensemble qui minimise $\frac{w_i}{|S_i \cap R|}$.
  $\mathcal{C} \leftarrow \mathcal{C} \cup S_i$.
  $c_u \leftarrow \frac{w_i}{|S_i \cap R|}$ pour tous $u \in S_i \cap R$. {cette définition est pour l'analyse seulement.}
  $R \leftarrow R \setminus S_i$.
Retourner $\mathcal{C}$.

1. Donner une interprétation de c_u.

2. Montrer que $\sum_{S_i \in \mathcal{C}} w_i = \sum_{u \in U} c_u$.

3. Notons OPT le coût d'une solution optimale. Renommons les éléments de l'univers U dans l'ordre dans lequel l'algorithme les a couverts, donc pour $i < j$, u_i a été couvert en même temps ou avant u_j. Montrer que pour tout $i = 1, \ldots, n$,
$$c_{u_i} \leq OPT/(n - i + 1).$$

4. Déduire que l'algorithme glouton ci-dessus est une $O(\log n)$-approximation.

2 Couverture par sommets

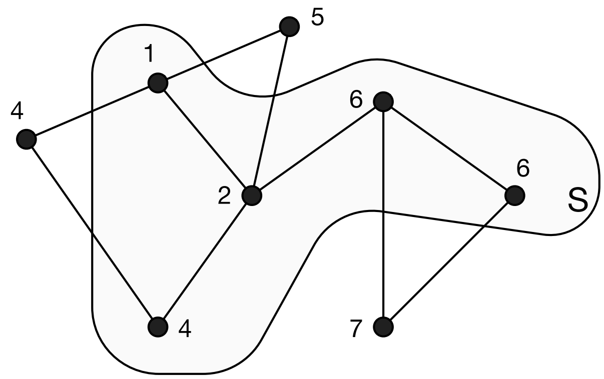

Figure 2: Le problème de couverture par sommets.

On vous donne un graphe $G(V, E)$ avec $n = |V|, m = |E|$ et chaque sommet $v \in V$ est associé avec un poids w_v. Une *couverture par sommets* est un sous-ensemble des sommets $S \subseteq V$ tel que chaque arête $e \in E$ a au moins une extrémité dans S. L'objectif est de trouver une couverture des arêtes telle que le poids total $w(S) = \sum_{v \in S} w_v$ soit minimal.
Ce problème est NP-complet. Nous allons chercher un algorithme d'approximation. Intuitivement, nous allons associer un prix p_e à chaque arête. Les prix p_e (pour $e \in E$) sont *équitables* si $\sum_{e \ni v} p_e \leq w_v$ pour chaque sommet v; c.a.d., les arêtes adjacentes à v ne payent pas plus que le poids du sommet v. Un sommet v est dit *saturé* si $\sum_{e \ni v} p_e = w_v$. Considérez l'algorithme suivant.

Algorithm 2 Algorithme pour Couverture des arêtes.

1: Initialement, $p_e \leftarrow 0$ pour toutes $e \in E$.
2: **while** il y a une arête $e = (u, v)$ telle que soit u ou v n'est pas saturé **do**
3: Choisir une telle arête e arbitraire.
4: Augmenter p_e au maximum sans violer la propriété équitable.
5: Retourner l'ensemble S des sommets saturés.

1. Montrer que pour chaque couverture des arêtes S, on a $\sum_e p_e \leq w(S)$.
2. Prouver que l'ensemble S et les prix p_e retournés par l'algorithme satisfait $w(S) \leq 2 \sum_{e \in E} p_e$.
3. Montrer que l'algorithme est 2-approximation.