Université de Lille

FST- Département Informatique

ACT

2018–2019
Master1 d'Informatique

Noyau d'un graphe

Exercice 1 : Noyau d'un graphe

Le *noyau* d'un graphe dirigé est un ensemble de sommets qui est:

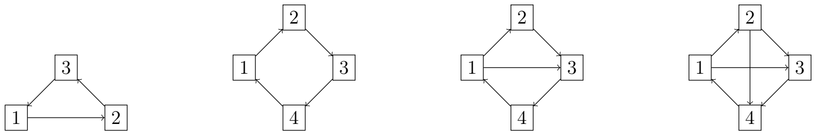
- indépendant: aucun arc ne relie deux sommets du noyau
- et dominant: tout sommet est soit dans le noyau soit a un successeur dans le noyau.

Le noyau a par exemple des applications pour certains jeux à deux joueurs (Nim, échecs, ...): un sommet correspond à une configuration du jeu, un arc relie une configuration à ses successeurs, une configuration perdante est une configuration qui n'a pas de successeur; si le graphe est acyclique (pas de partie infinie), quand un joueur peut jouer un coup qui amène dans une position du noyau, il a une stratégie gagnante (Pourquoi?).

Comme le noyau est indépendant, l'autre joueur sort du noyau obligatoirement: comme chaque sommet hors noyau a un successeur dans le noyau, le premier joueur rejoue un coup qui amène dans le noyau, ainsi de suite... Le deuxième joueur sera bloqué après un certain nombre de coups dans une configuration sans successeur, donc perdante.

Le problème que nous appellerons ici le problème du noyau sera donc:
Entrée: $G = (S, A)$ - -un graphe dirigé
Sortie: Oui, Ssi il existe un noyau de G.

Q 1. Parmi les exemples suivants, lesquels ont un noyau?



Il n'y a pas de noyau pour le premier (comme pour tout cycle de longueur impaire).
Il y a deux noyaux pour le second: 1, 3 ou 2,4
Il y a un seul noyau pour le troisième: 2,4.
Pour le dernier: il n'y a pas de noyau, les seuls ensembles indépendants sont de cardinal 1, et un sommet a au plus deux arcs entrants, donc un somment ne peut être le successeur des trois autres.

Q 2. Montrer que le problème du noyau est NP.

Soit n le cardinal de S.
Le certificat est un sous-ensemble de sommets.
On peut le représenter par un tableau de n booléens par exemple; alors la taille sera n, avec une taille de la donnée au moins n: donc la taille du certificat est bien polynomialement bornée par celle de l'instance.
La vérification consiste à vérifier que le noyau est bien indépendant, i.e. qu'il n'y a pas d'arcs entre deux sommets du noyau et que tout sommet hors noyau a un successeur dans le noyau.

```
pour chaque sommet s {
    si Cert(s) // s est dans le noyau {
    //vérifier qu'il n'a pas de lien avec les autres sommet  du noyau
       pour chaque sommet s'
          si   (Cert(s') && (s'\=s ) && ((s,s') dans A )) return false ;
    }
    sinon { // s n'est pas  dans le noyau
        Boolean b= faux;
        pour chaque sommet s' {
           si   (Cert(s')  && ((s,s') dans A))  b=vrai;}
        si non b return Faux;  }  // s n'a pas de successeur dans le noyau
}
return Vrai;
```

La vérification est bien polynomiale, en $O(n^2)$, si le test ((s,s') dans A) est en O(1).

Q 3. Montrer que 3-Sat se réduit polynomialement dans le problème du noyau. Qu'en conclure pour le problème du noyau?

On va donc proposer une réduction qui transforme toute instance en 3-Sat, i.e. toute formule booléenne sous forme de conjonctions dd 3 littéraux en un graphe, tel que ce graphe ait un noyau Ssi la formule se départ était satisfiable.
Soit donc une formule de 3-SAT $\vee_1^p C_i$ avec $C_i = l_i^1 \wedge l_i^2 \wedge l_i^3$. La taille de la formule sera donc supérieure à n -on suppose que chaque variable apparaît au moins une fois, sinon on l'élimine-, et supérieure à p: il ya p clauses.
On va construire $G = (S, A)$ comme suit:

Sommets: I

un sommet par clause,

deux pour chaque variable -un pour le littéral x_i un pour le littéral $\bar{x}_i$,

deux noeuds supplémentaires A et B

Donc S a 2n+p+2 sommets si n est le nombre de variables.

Arcs:

de A vers B

de B vers chaque sommet-clause,

de chaque sommet-clause vers A

d'un sommet-clause vers un sommet-littéral si elle contient le littéral correspondant

entre les deux sommets associés à une variable (et dans les deux sens)

La réduction est bien polynomiale: S a 2n+p+2 sommets si n est le nombre de variables, donc la construction des sommets est linéairement bornée par rapport à la taille de la formule. La construction des arcs se fera bien aussi en temps polynomial: pour un couple de sommets (s,s'), tester si il on ajoute un arc de s vers s' peut se faire en 0(1).
Montrons qu'elle est correcte:

- Si la formule de départ est satisfiable, prenons comme éléments du noyau les sommets-littéraux correspondant à une valuation rendant vraie la formule.

  Cela forme bien un noyau:

  – il est indépendant,

  – de chaque sommet-clause il y a un arc vers un sommet-littéral du noyau car la formule est satisfaite par la valuation, de chaque sommet-littéral hors noyau un arc vers sa négation qui est dans le noyau, et il y a un arc de A vers B.

- Réciproquement, si il y a un noyau: dans G, remarquons d'abord qu'il contient B; sinon au moins une clause est dans le noyau car B doit avoir un successeur dans le noyau et ses seuls successeurs sont les commets-clauses. mais alors A n'est pas dans le noyau par indépendance et son seul successeur B n'est pas dans le noyau: c'est impossible.

  Donc B est dans le noyau et ni A, ni aucune clause n'y sont par indépendance.

  On peut remarquer que le noyau contient pour chaque variable un et un seul littéral correspondant- un seul car il est indépendant, au moins un, car si un sommet n'est pas dans le noyau, un de ses successeurs y est. On construit la valuation en mettant x à Vrai si le littéral x est dans le noyau, à Faux si c'est le littéral $\bar{x}$.

  Pour chaque clause, il y a un arc vers un sommet du noyau donc chaque clause est rendue vraie par la valuation et la formule est satisfiable,

Le pb est donc NP-dur, donc NP-complet avec la question précédente.