Algorithmique Avancée 2018 — École Centrale-Supélec

PC6 : NP-complétude

Christoph Dürr, Nguyễn Kim Thắng

1 Coloriage des graphes

Un *k-coloriage* d'un graphe $G(V, E)$ est une fonction $f : V \to \{1, 2, \ldots, k\}$ telle que $f(u) \neq f(v)$ pour toute arête $e = (u, v) \in E$. Prouver:

1. Si le graphe $G(V, E)$ est bipartite alors G est 2-coloriable.
2. Etant donné un graph $G(V, E)$, décider si G est 3-coloriable est NP-complet.
   Indice: Réduction vers 3-SAT.

2 Subset Sum

Etant donné un ensemble de n entiers $S = \{a_1, a_2, \ldots, a_n\}$ et un entier V. Le but est décider s'il existe un sous-ensemble de S tel que la somme des entiers de ce sous-ensemble soit égal à V. Montrer que SUBSET-SUM est NP-complet.

3 2-Partition

En entrée, nous avons un ensemble de n entiers $S = \{a_1, a_2, \ldots, a_n\}$. En sortie, nous devons renvoyer un sous-ensemble Tde S tel que la somme des entiers de T soit égale à la somme des entiers de $S \backslash T$. Montrer que 2-PARTITION est NP-complet.

4 Couverture par sommets

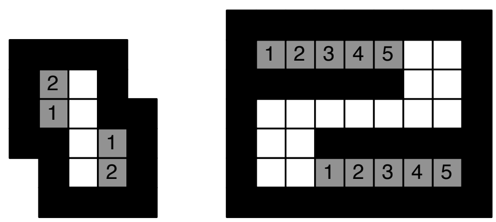
En entrée, nous avons un graphe et un entier K. En sortie, nous devons renvoyer un sous-ensemble de sommet C de taille K tel que pour tout arête (i, j) du graphe i ou j appartient à C. Montrer que COUVERTURE PAR SOMMETS est NP-complet.

5 Jeu de Cubic

Définition. Cubic *est un jeu à un joueur qui se joue sur une grille de carrés. Il existe trois types de carrés:*

- *les murs (représentés en noir): ceux-ci ne peuvent pas être déplacer,*
- *les cases vides (représentées en blanc),*
- *les blocs de couleur (représentés en gris, la couleur est représentée par un nombre).*

A chaque tour, le joueur peut déplacer un bloc de couleur d'une case vers la gauche ou d'une case vers la droite, si les cases correspondantes ne sont pas déjà occupées par un mur ou un autre bloc de couleur. Si un bloc de couleur se trouve au dessus d'une case vide, il "tombe" jusqu'à rencontrer un mur ou un bloc de couleur. Si deux blocs de la même couleur partage un côté, ils disparaissent. Éventuellement, les blocs de couleur se trouvant au dessus-eux tombent. Le joueur gagne s'il arrive à faire disparaître tous les blocs de couleur.



jeu 1 jeu 2

Figure 1: Deux configurations initiales du jeu *Cubic*.

1. Résolvez les problèmes de *Cubic* de la figure 1.

Étant donné une instance de *Cubic*, nous souhaitons déterminer si celle-ci admet une solution. Nous allons montrer que ce problème est NP-complet. Pour cela, nous allons effectuer une réduction à partir de SAT.

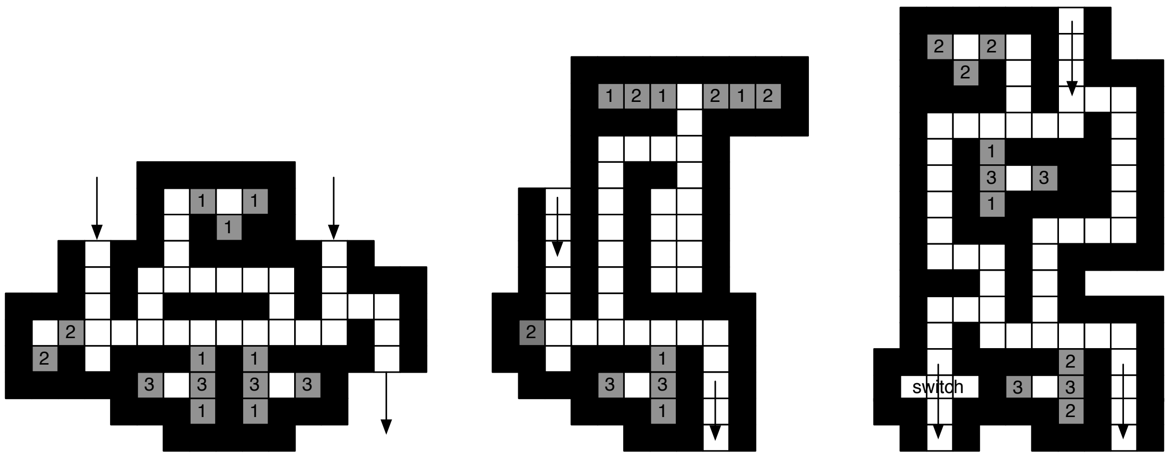
encodage d'une variable encodage de la sortie

Figure 2: Encodage des variables et de la sortie.

2. Exprimez le problème SAT sous la forme de circuit (graphe orienté où les sommets représentent des opérateurs logiques et tous les arcs sont orientés du haut vers le bas).

Nous allons transformer une instance de SAT encodée sous forme de circuit en une instance de *Cubic* qui admet une solution si et seulement si le circuit admet une affectation des variables qui donne une sortie VRAI. On encode les variables et les sorties comme indiqué dans la figure 2.

3. Expliquez comme fonctionne cette encodage. De quels autres gadgets a-t-on besoin?



porte AND   porte SWITCH   porte mystère

Figure 3: Différents gadgets.

4. Expliquez le fonctionnement de la porte AND de la figure 3.

5. Expliquez comment faire une porte NOT et une porte OR.

6. A quoi sert la construction SWITCH de la figure 3.

7. En utilisant la structure SWITCH, créez une structure faisant un croisement de fils.

8. Que fait la construction mystère de la figure 3.

9. Montrez que *Cubic* est NP-complet.

[https://www2.stetson.edu/˜efriedma/papers/spiral.ppt]