Télécom SudParis Octobre 2021

Institut Polytechnique de Paris

Séminaire d'Initiation à la Recherche

TP 1 - Coloriage de graphes

Dans ce TP, nous nous intéressons à l'utilisation de l'intelligence artificielle pour résoudre des problèmes combinatoires pouvant être encodés dans le langage propositionnel classique. Plus précisément, nous nous intéressons au problème de la satisfiabilité propositionnelle (SAT) connu pour être le problème de décision NP-Complet de référence. Ce problème consiste à décider si une formule sous forme normale conjonctive (CNF) [1] est satisfiable ou non. Autrement dit, est ce qu'il existe une affectation des variables de la formule qui la rend vraie.

Voici un exemple d'une formule CNF Φ construite à partir de trois variables booléennes $x_1, x_2, x_3 \in \{vrai, faux\}$ et contenant quatre clauses. Cette formule est satisfiable car il existe une affection (appelé modèle) des variables x_1, x_2 et x_3 qui la satisfait en l'occurrence $x_1 = vrai$, $x_2 = vrai$ et $x_3 = vrai$. Toutefois, ajouter la clause $(\neg x_1 \vee \neg x_2 \vee \neg x_3)$ à Φ la rend non satisfiable.

$$\Phi = \begin{array}{ll} (\neg x_1 \vee x_2) & \wedge \\ (\neg x_2 \vee x_3) & \wedge \\ (\neg x_3 \vee x_1) & \wedge \\ (x_1 \vee x_2 \vee x_3) & \end{array}$$

Pour résoudre le problème SAT, plusieurs algorithmes ont été proposés dans la littérature. Dans ce TP, vous pouvez utiliser l'un des solveurs de l'état de l'art `minisat` [2] (implémenté en C++) ou `pysat` [3] (une librairie python pour utiliser des solveurs SAT implémentés en C++).
Le format d'entrée standard pour les solveurs SAT est le format `dimacs`. Pour notre formule Φ, voici le contenu du fichier qui la représente :

```
c ceci est une formule CNF sous format dimacs
c cette formule contient 3 variables et 4 clauses
p cnf 3 4
-1 2 0
-2 3 0
-3 1 0
1 2 3 0
```

Plusieurs problèmes combinatoires (e.g., vérification de modèles, planification, ordonnancement, etc.) peuvent être encodés en SAT en vue de leur résolution. Dans la suite, nous nous intéressons au problème de coloriage de graphes.

---

1. Une formule CNF (Conjunctive Normal Form en anglais) est une conjonction de clauses de la forme $c_1 \wedge c_2 \wedge \ldots \wedge c_n$ telle que chaque clause c_i ($1 \leq i \leq n$) est une disjonction de littéraux de la forme $c_i = x_1 \vee x_2 \vee \ldots \vee x_m$.
2. `http://minisat.se/MiniSat.html`
3. `https://pysathq.github.io/`

Problème de coloriage de graphe

Soit $\mathcal{G} = (V, E)$ un graphe tel que $|V| = n$ et k un entier représentant le nombre de couleurs. Le problème de coloriage de graphe consiste à vérifier s'il est possible de colorier le graphe avec au plus k couleurs de sorte que chaque deux sommets adjacents soient de couleurs différentes. La figure 1 illustre une solution du problème de coloriage de graphe pour $k = 3$. Notons que ce problème ne possède pas de solution pour $k = 2$.

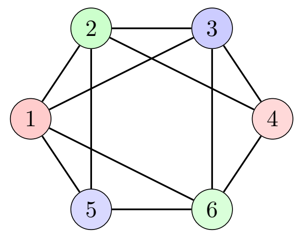

FIGURE 1 – Une solution de coloriage du graphe $\mathcal{G}$ avec $k = 3$

Pour résoudre ce problème, il est possible de l'encoder en SAT tel que la solution du problème correspond à un coloriage recherché. Pour cela, nous allons associer à chaque sommet i des variables booléennes $x_{i,1}, \ldots, x_{i,k}$ où $x_{i,j} = vrai$ stipule que la couleur j est attribuée au sommet i. Ensuite, nous allons ajouter les contraintes suivantes :

— un sommet a une seule couleur

$$\bigwedge_{1 \leq i \leq n} (\sum_{1 \leq j \leq k} x_{i,j} = 1) \equiv \bigwedge_{1 \leq i \leq n} [(x_{i,1} \vee \ldots \vee x_{i,k}) \wedge \bigwedge_{1 \leq l < p \leq k} (\neg x_{i,l} \vee \neg x_{i,p})] \tag{1}$$

— chaque deux sommets adjacents ne peuvent pas avoir la même couleur

$$\bigwedge_{(i,j) \in E} \bigwedge_{1 \leq l \leq k} (\neg x_{i,l} \vee \neg x_{j,l}) \tag{2}$$

Clairement, toute solution des deux contraintes (1) et (2) simultanément corresponds à un coloriage recherché.

1. À partir des équations (1) et (2), donner la formule CNF encodant le problème de coloriage de graphe pour le graphe de la figure 1 avec $k = 3$.
2. Écrire un parseur qui prend en entrée un graphe $\mathcal{G}$ et un entier positif k et qui renvoie un fichier contenant l'encodage du problème de coloration du graphe $\mathcal{G}$ avec k couleurs.
3. Exécuter votre programme pour différentes valeurs de k en téléchargeant le graphe `Karate` sur `https://networkx.org/documentation/stable/auto_examples/graph/plot_karate_club.html`.
4. Ajouter à votre programme une fonction qui génère aléatoirement un graphe ayant comme arguments le nombre de sommets et le nombre d'arêtes et qui cherche le nombre minimum de couleurs pour colorier ce graphe.