3-Coloration de graphes

K. Chai, M. Lacote, D. Lesnoff, I. Martayan, C. Zeng

Ecole d'été de mathématique et d'informatique

30 août 2019

Mathinfoly

Définitions

Définition

Un *graphe* est la donnée d'un couple (V, E) composé de :

- V d'un ensemble de *sommets* ou *noeuds* ;
- $E \subseteq \{\{x, y\} | (x, y) \in V^2 \wedge x \neq y\}$, d'un ensemble d'arêtes, qui sont des paires de sommets.

Un graphe *orienté* possède des arêtes ayant une orientation.

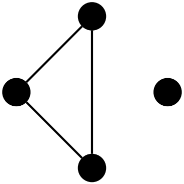

Figure 1 – Un graphe avec $|V| = 4$ et $|E| = 3$

k-coloration

Problème de k-coloration

Etant donné un graphe non orienté $G = (V, E)$, et un entier $k \geq 1$, existe-t-il une *coloration* $c : V \to \{1, 2, \cdots, k\}$ telle que : $\forall \{u, v\} \in E, c(u) \neq c(v)$?

On se place dans le cas $k = 3$.

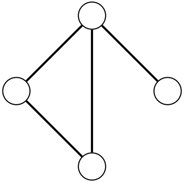

Figure 2 – Exemple de graphe à colorer

3-coloration : exemples

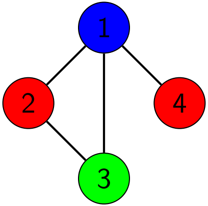

Figure 3 – 3-coloriage avec $|V| = 4$ et $|E| = 4$

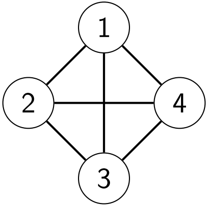

Figure 4 – 3-coloriage impossible avec $|V| = 4$ et $|E| = 6$

Formalisation en logique du premier ordre

Notations

Un graphe *non orienté* $G = (V, E)$:

- $n := |V|$ le nombre de noeuds ;
- $p := |E|$ le nombre d'arêtes ;

Signature

Nous considérons la signature $\Sigma = \{e^{r2}, c^{r2}, n^{f0}, p^{f0}\}$ ayant la sémantique suivante :

- $e(x, y)$: x est voisin de y ;
- $c(x, j)$: x a la couleur j.

Formalisation en logique du premier ordre

Contraintes

1. Tout sommet du graphe est coloré ;

Expressions en logique du premier ordre

1. $\forall x \in [\![1, n]\!], \exists j \in [\![1, k]\!], c(x, j)$;

Formalisation en logique du premier ordre

Contraintes

1. Tout sommet du graphe est coloré ;
2. Tout sommet possède au plus une couleur ;

Expressions en logique du premier ordre

1. $\forall x \in [\![1, n]\!], \exists j \in [\![1, k]\!], c(x, j)$;
2. $\forall x \in [\![1, n]\!], \exists i \in [\![1, k]\!], (c(x, i) \implies (\bigwedge_{j \neq i} \neg c(x, j)))$;

Formalisation en logique du premier ordre

Contraintes

1. Tout sommet du graphe est coloré ;
2. Tout sommet possède au plus une couleur ;
3. Deux sommets voisins ont des couleurs différentes.

Expressions en logique du premier ordre

1. $\forall x \in [\![1, n]\!], \exists j \in [\![1, k]\!], c(x, j)$;
2. $\forall x \in [\![1, n]\!], \exists i \in [\![1, k]\!], (c(x, i) \implies (\bigwedge_{j \neq i} \neg c(x, j)))$;
3. $\forall x, y \in [\![n]\!], \forall i \in [\![1, k]\!], \neg(e(x, y) \wedge (c(x, i) \wedge c(y, i)))$.

Format et variables

Fichier d'entrée

- n et p sur la première ligne
- p lignes décrivant les paires de sommets reliées par une arête

Variables

- $3n$ variables $x_{(i-1)n+(j-1)}$ où $i \in [\![1, n]\!]$, $j \in [\![1, 3]\!]$

Nombre de clauses

- $27n + 3p$ clauses

Fonctionnement du programme

Programme OCaml

- Lecture du fichier d'entrée
- Génération de la forme normale conjonctive

Fonctionnement du programme

Programme OCaml

- Lecture du fichier d'entrée
- Génération de la forme normale conjonctive

Programme Python

- Appel du programme OCaml
- Appel du SAT solveur
- Affichage du graphe à partir de la solution

Type de données et algorithmes

```
type formule =
  |Var of int
  |Non of formule
  |Et of formule * formule
  |Ou of formule * formule;;
```

1. Traduction récursive en forme normale conjonctive
2. Réécriture avec des fonctions récursives terminales
3. Généralisation pour le problème de *k*-coloration

Performances du programme
Amélioration du modèle

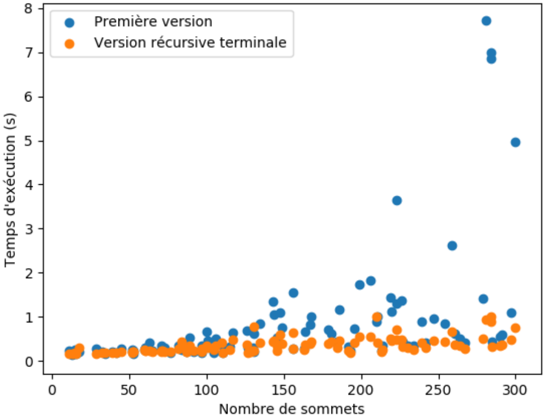

Figure 5 – Temps d'exécution en fonction de n

Performances du programme

Amélioration du modèle

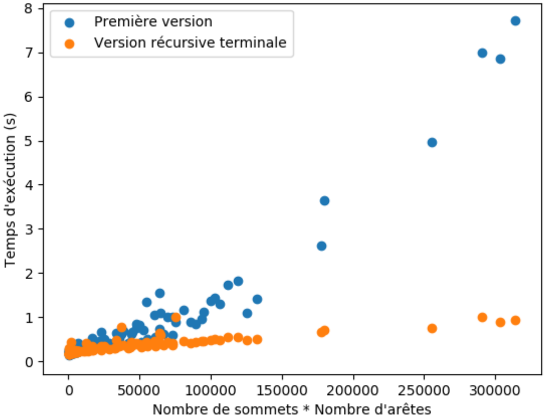

Figure 6 – Temps d'exécution en fonction de *np*

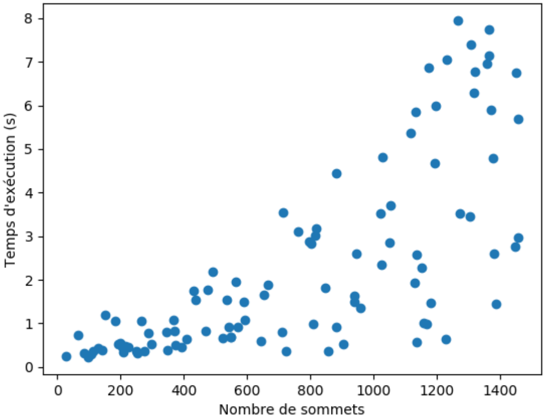

Figure 7 – Temps d'exécution en fonction de n

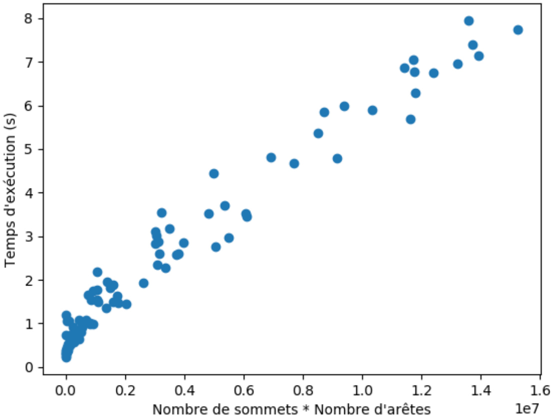

Figure 8 – Temps d'exécution en fonction de *np*

Résultats du programme

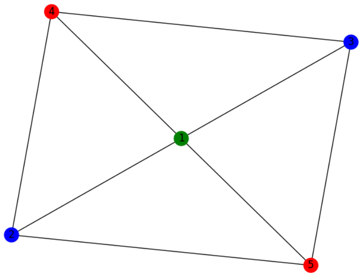

Figure 9 – Une solution pour $n = 5$ et $p = 8$

Conclusion et autre application

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

Figure 10 – Une grille de sudoku

- 9 couleurs
- 81 sommets
- lignes, colonnes et blocs reliés

Bibliographie I

P. Lafourcade.
Cours de logique.
Ecole d'été de mathématique et d'informatique - Mathinfoly, 2019.

D. Babic.
Satisfiability Suggested Format.
Domagoj-Babic-Website, 1993.

Wikipedia
Boolean satisfiability problem.
Wikipedia page, 2019.