1 Réduction de 3-SAT à CNF-SAT

Une formule *CNF* (pour *Conjunctive Normal Form*) est la donnée d'un ensemble de variables $x_1, \dots, x_n$ et d'un ensemble de clauses $C_1, \dots, C_k$ où chaque clause est une disjonction de littéraux (un littéral est soit une variable x_i soit la négation d'une variable $\neg x_i$). Pour tout $1 \leq j \leq k$, on a donc $C_j = \bigvee_i v_i^j$ où $v_i^j = x_l$ ou $v_i^j = \neg x_l$ pour un certain l.

Une formule CNF est satisfiable s'il existe une assignation des variables $x_1, \dots, x_n$ à vrai ($\top$) ou faux ($\bot$) telle que chaque clause soit satisfaite (c'est à dire que pour chaque clause au moins un de ses littéraux est vrai). Du point de vue logique une formule CNF est donc une formule de la forme : $\exists x_1, \dots x_n \in \{\top, \bot\}^n : \bigwedge_j \left(\bigvee_i v_i^j\right)$.

Définition 1. *Le problème* CNF-SAT *est celui de décider si une formule CNF est satisfiable.*

Définition 2. *Le problème* k-SAT *est le problème CNF-SAT où l'on impose que chaque clause contienne au plus k littéraux.*

Exercice 1. Montrer que les problèmes *CNF-SAT* et *k-SAT* (pour tout $k \in \mathbb{N}$)) sont dans $\mathcal{NP}$.

Solution 1. On considère comme certificat la valuation de chacune des variables. Vérifier la valuation est faisable en temps polynomial.

Exercice 2. Montrer que la formule $(v_1 \vee v_2 \vee v_3 \vee v_4)$ est satisfaite seulement quand il existe l tel que $(v_1 \vee v_2 \vee l) \wedge (\neg l \vee v_3 \vee v_4)$ est aussi satisfaite (où l est une variable qui n'apparaît pas dans les littéraux $v_1, \dots, v_4$).

Solution 2. Soit $(v_1 \vee v_2 \vee v_3 \vee v_4)$ est satisfiable alors l'un des $v_1, \dots, v_4$ est vrai. Si v_1 ou v_2 est vrai alors $l = \bot$ rend la second formule satisfiable, sinon on a v_3 ou v_4 qui est vrai et $l = \top$ rend la formule satisfiable.

Exercice 3. Montrer que 3-SAT est $\mathcal{NP}$-complet. Pour cela, partir du problème CNF-SAT (qui est supposé $\mathcal{NP}$-complet) et montrer que pour toute formule CNF φ on peut trouver une formule φ' qui est satisfiable si et seulement si φ l'est avec $|\varphi'|$ de taille polynomiale en $|\varphi|$ et où chaque clause de φ' contient au plus 3 littéraux.

Solution 3. Étant donnée une formule CNF, si on a une clause $v_1 \vee \dots \vee v_k$ avec $k > 3$ alors on la transforme $v_1 \vee v_2 \vee l$ et $\neg l \vee v_3 \dots v_k$ qui sont deux formules plus petites. En répétant ce processus, on obtient une formule CNF équivalente mais dont toutes les clauses ont au plus 3 littéraux.

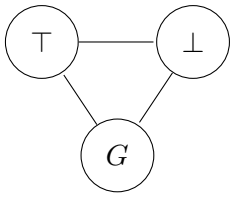
Définition 3. *Une formule DNF (pour* Disjunctive Normal Form*) est la donnée d'un ensemble de variables $x_1, \dots, x_n$ et d'une disjonction de clauses où chaque clause est une conjonction de littéraux. En forme logique une formule DNF est donc $\exists x_1, \dots, x_n \in \{\bot, \top\}^n : \bigvee_j (\bigwedge_i v_i^j)$*

Exercice 4. Le problème DNF-SAT est celui de la satisfiabilité d'une formule DNF. Le problème DNF-SAT est-il dans $\mathcal{NP}$? co-$\mathcal{NP}$? $\mathcal{P}$?

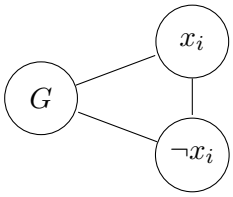
Solution 4. Le problème DNF-SAT est dans ces trois classes. En effet, pour satisfaire une formule DNF, il suffit de satisfaire l'un de ses clauses donc on traite clause par clause. Et une clause est satisfiable si et seulement si elle ne comporte pas un littéral et son contraire.

2 Réduction de 3-SAT à 3-coloriage

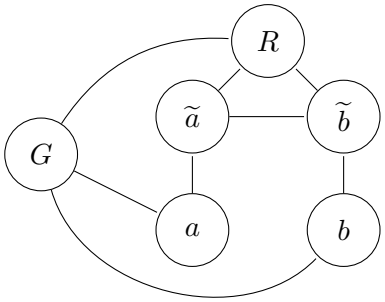
Définition 4. *Étant donné un graphe $G = (V, E)$, G est dit k-coloriable si l'on peut colorier ses nœuds avec k couleurs de telle façon que deux nœuds reliés n'ont jamais la même couleur. Formellement, la k-coloriabilité de G s'exprime de la façon logique suivante $\exists (c : V \to \{1, \dots, k\})\ \forall (i, j) \in E\ : c(i) \neq c(j)$.*



(a) Gadget de couleur



(b) Gadget de littéraux



(c) Gadget OR

Exercice 5. Montrer que pour tout k fixé, le problème du k-coloriage est dans $\mathcal{NP}$.

Solution 5. Ce problème est dans $\mathcal{NP}$ car on prend le graphe et un certificat correspond à choisir une couleur pour chaque nœud tandis que le vérificateur correspond à vérifier que le coloriage est bon.

Exercice 6. Montrer que si le problème de la $k+1$-coloriabilité est dans $\mathcal{P}$ alors celui de la k-coloriabilité l'est aussi.

Solution 6. Si l'on sait résoudre la $k+1$-coloriabilité en temps polynomial alors il suffit d'ajouter un nœud relié à tous les autres et $k+1$-colorier le graphe obtenu. En retirant le nœud ajouté on obtient une k-coloration du graphe.

Nous allons maintenant monter que 3-SAT se réduit à 3-coloriage. Pour cela nous allons introduire plusieurs gadgets.

Gadget de couleurs Dans notre graphe, les trois couleurs vont représenter soit vrai ($\top$), faux ($\bot$) ou autre (G pour Ground). Pour encoder ces couleurs notre premier gadget est d'inclure le graphe de la figure **??**. Nous brancherons ensuite souvent des nœuds à G ou $\bot$ pour interdire ces couleurs, il s'agira toujours de ces mêmes nœuds G et $\bot$ (le gadget n'est présent qu'une seule fois même si ses nœuds sont utilisés dans les autres gadgets).

Exercice 7. Trouver un gadget qui impose qu'un nœud soit colorié de la même couleur que $\top$.

Gadget littéraux Pour chaque variable x_i nous allons créer deux nœuds : x_i et $\neg x_i$ que nous allons brancher à G et entre eux selon le schéma de la figure **??**.

Gadget OR Supposons que a et b représentent des valeurs booléennes (ce que l'on force en les branchant à G), le gadget OR est celui de la figure **??** (le G est celui du premier gadget).

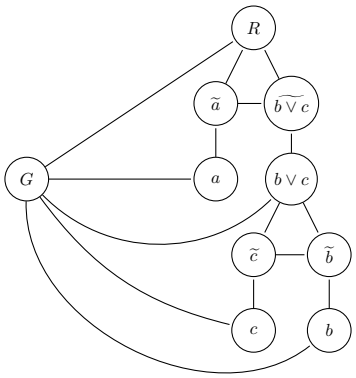
Exercice 8. Montrer que dans tout coloriage du gadget, la couleur de R est la même que celle de a ou la même que celle de b. Et montrer que si on limite le graphe à ce gadget, alors R peut être colorié par la couleur de a ou par celle de b.

Solution 8. Soit a et b sont de la même couleur et alors $\widetilde{a}$ et $\widetilde{b}$ doivent être coloriés en les deux couleurs complémentaires donc R est forcément colorié de cette couleur commune.

Soit a et b sont coloriés en $\top$ et $\bot$ tandis que R est forcément colorié en l'une de ces couleurs puisque R est lié à G.

Exercice 9. Trouver un gadget qui réalise le OR entre trois valeurs booléennes a, b, c (représentées par des nœuds coloriés de la même couleur que $\top$ ou $\bot$).

Solution 9.



Preuve finale **Exercice 10.** Trouver un gadget pour vérifier chacune des clauses.

Solution 10. Pour chaque clause $v_1 \vee v_2 \vee v_3$ on utilise le 3-OR avec $a = v_1$, $b = v_2$, $c = v_3$ puis on relit R à G et $\bot$. Ce gadget ne peut être colorié que si un des v_i est colorié avec la couleur de $\top$.

Exercice 11. Présenter une réduction polynomiale de 3-SAT au 3-coloriage.

Solution 11. Nos gadgets littéraux introduisent 2 nœuds et 3 arêtes par variable tandis que nos gadgets clauses introduisent 7 nœuds et 12 arêtes. Dans tous les cas la formules est transformé en graphe de taille linéaire et a fortiori polynomial.

3 Miscellanées

Exercice 12. Le problème du 2-coloriage est-il dans $\mathcal{NP}$? co-$\mathcal{NP}$? $\mathcal{P}$?

Solution 12. Le 2-coloriage est dans $\mathcal{P}$ car l'algorithme glouton suivant fonctionne.

On répète la procédure suivante sur chaque composante connexe du graphe. On choisit un nœud au hasard et on le colorie dans la première couleur et de parcourir récursivement ses voisins. À chaque fois soit le voisin n'est pas colorié et on le colorie (de la couleur opposée) récursivement, soit il est colorié et on vérifie sa couleur.

Exercice 13. Montrer que SAT se réduit à 3-SAT (c'est à dire la satisfiabilité de formules booléennes composées de $\vee$, $\wedge$, $\neg$ et de variables).

Solution 13. On traduit récursivement les formules avec t en CNF. Tout d'abord on pousse les négations en bas en utilisant les formules de De Morgan puis $t(\varphi \wedge \psi) = t(\varphi) \wedge t(\psi)$; et pour $t(\varphi \vee \psi)$ on ajoute x à chaque clause de $t(\varphi)$ et $\neg x$ à chaque clause de $t(\psi)$ et la conjonction des deux nous donne ce que l'on veut.

Cette réduction est polynomiale car si on prend l'arbre de la formule, le nombre de clauses est le nombre de feuilles tandis que le nombre de littéraux dans une clause est le nombre de or rencontrés plus un.

Exercice 14. Dans quelles classes se trouve le problème 2-SAT ?

Solution 14. 2-sat est dans $\mathcal{P}$. On regarde le graphe dont les nœuds sont les littéraux possibles et on met une arête entre v et $\neg w$ s'il existe une clause $v \vee w$. La formule est satisfiable si et seulement si une variable et sa négation appartiennent à une composante fortement connexe de ce graphe.

4 Problème de clique

Définition 5. *Le problème de la clique consiste à décider si étant donné un graphe G et un entier k que le graphe G contient une k-clique (un ensemble de k nœuds tous reliés les uns aux autres).*

Exercice 15. Montrer que le problème de la clique est $\mathcal{NP}$-complet.

Suggestion : trouver une réduction avec 3-SAT. Étant donné $\bigvee_j(\bigwedge_i v_i^j)$ on créé un nœud par v_i^j (si un même littéral apparaît dans plusieurs clauses, alors il est présent plusieurs fois).

Solution 15. Dans le graphe on relit v_i^j à v_l^k s'il ne sont pas dans la même clause (i.e. $k \neq j$) et s'il l'un n'est pas la négation de l'autre. Si l'on trouve une k-clique alors il y est possible de satisfaire k clauses (en prenant k comme le nombre de clause on a la satisfiabilité de la formule 3-CNF).