Tutorat de Complexité - 1

La totalité des problèmes posés sont tirés des références suivantes :

- Le cours *Introduction à la complexité* de Paul Rozière :
- Le cours *Fondements de l'Informatique : Logique, modèles, calculs* d'Olivier Bournez :
- *Computational Complexity* de Christos Papadimitriou

1 Réductions

Définition 1.1 (`MAXSAT`). On s'intéresse au problème :

- Entrée : Soit $\mathcal{C}$ un ensemble de clauses et $n \in \mathbb{N}$ avec $n \leqslant \mathrm{Card}(\mathcal{C})$.
- Question : Existe-t-il une valuation satisfaisant au moins n clauses de $\mathcal{C}$?

Question 1. Montrer que `MAXSAT` est NP-complet.

Définition 1.2 (`NAESAT`). On s'intéresse au problème :

- Entrée : Soit $\mathcal{C}$ un ensemble de clauses.
- Question : Existe-t-il une valuation v telle que v et $\bar{v}$ satisfont $\mathcal{C}$

Question 2. Montrer que `NAESAT` est NP-complet.

Définition 1.3 (2-clause). Une *2-clause* est une clause comptant au plus 2 littéraux.

Définition 1.4 (`MAX2SAT`). On s'intéresse à la variante de `MAXSAT` suivante :

- Entrée : Soit $\mathcal{C}$ un ensemble de 2-clauses et $n \in \mathbb{N}$ avec $n \leqslant \mathrm{Card}(\mathcal{C})$.
- Question : Existe-t-il une valuation satisfaisant au moins n clauses de $\mathcal{C}$?

Question 3. Soit v une valuation. En fonction des images de x, y et z par v, étudier le nombre de 2-clauses satisfaisables parmi

$$\{x, y, z, w, \neg x \vee \neg y, \neg x \vee \neg z, \neg y \vee \neg z, x \vee \neg w, y \vee \neg w, z \vee \neg w\}$$

Question 4. Montrer que `MAX2SAT` est NP-complet.

Définition 1.5 (`TABLERONDE`). On s'intéresse au problème :

- Entrée : Soit E un ensemble de chevaliers et l une liste de paires de chevaliers qui se détestent.
- Question : Le Roi Arthur (qui fait partie de E) peut-il faire un plan de table tel que deux chevaliers se détestant ne sont pas côte-à-côte ?

Question 5. Montrer que `TABLERONDE` est NP-complet.

2 Clauses de Horn

Notations. On notera la clause $p_1 \vee \cdots \vee p_n \vee \neg q_1 \vee \cdots \vee \neg q_m$ par le couple de deux ensembles finis de variables propositionnelles

$$\{q_1, \ldots q_m\} \to \{p_1, \ldots, p_n\}$$

La clause vide, $\to$, s'interprète sémantiquement par l'absurde.

Définition 2.1 (Partie positive et négative). La *partie positive* de $\Gamma \to \Delta$ désigne Δ, la *partie négative* Γ.

Définition 2.2 (Clauses de Horn). Une clause de Horn est une clause dont la partie positive contient au plus une variable propositionnelle.

Définition 2.3 (Coupures). La règle de *coupure* associe à deux clauses une nouvelle clause selon le schéma suivant :

$$\frac{\Gamma_1 \cup \{p\} \to \Delta_1 \qquad \Gamma_2 \to \Delta_2 \cup \{p\}}{\Gamma_1 \cup \Gamma_2 \to \Delta_1 \cup \Delta_2}$$

Définition 2.4 (Déduction et réfutation par coupures). On définit inductivement la *déduction par coupures*. Une clause c se *déduit par coupures* d'un ensemble de clauses S si et seulement si $c \in S$ ou c est obtenue par une règle de coupure à partie de clauses qui se déduisent par coupures de S.

Un ensemble de clauses est *réfutable par coupures* signifie que la clause vide se déduit de S.

Question 6. Montrer que si un ensemble de clauses n'est pas satisfaisable il contient nécessairement une clause positive.

Question 7. Montrer que le fait d'être une clause de Horn est stable par coupures.

Soit S un ensemble de clauses de Horn. On suppose que pour toute variable p, p n'apparaît à la fois positivement et négativement dans aucune clause de S.

Question 8. Soit $\to p$ une clause positive de S. On note S_p l'ensemble des clauses de S contenant p. Soit $\mathrm{res}^+(S,p)$ l'ensemble des clauses obtenues par une seule règle de coupure entre $\to p$ et une clause de S qui contient p en position négative. Montrer que S est satisfaisable ssi $(S \setminus S_p) \cup \mathrm{res}^+(S,p)$ est satisfaisable.

Question 9. En déduire un algorithme pour la satisfaisabilité d'un ensemble de clauses de Horn en temps quadratique en fonction de l'entrée.