Master DID - Complexité Algorithmique - Session 2

Vendredi 28 juin 2019. 09h00-12h00. Salle Z-011

La précision et la clarté de vos réponses sont *fondamentales*. Chaque réponse doit être justifiée. Barème indicatif et documents interdits. Durée 1h30.

Exercice 1. [3pts] Soit $k \in \mathbf{N}$ et soit $f_k : \mathbf{N} \to \mathbf{N}$ la fonction définie par $n \mapsto k$.

1. [2pts] Écrivez un programe sur la machine de Turing qui calcule la fonction f. On supposera que l'alphabet est binaire : $\Sigma = \{0, 1\}$.

2. [1pt] En déduire qu'il existe une infinité de fonctions que l'on peur calculer à l'aide d'une machine de Turing.

Exercice 2. [1pt] Soit n la taille des données traitées par un algorithme. Pour chacune des fonctions suivantes, citez un algorithme dont c'est la complexité :

(1) $O(\log n)$,

(2) $O(n)$,

(3) $O(n \log n)$,

(4) $O(n^2)$.

Exercice 3. [1pt] Si Π désigne un problème de décision, son *co-problème*, noté $\overline{\Pi}$, est défini par la même instance que celle de Π, mais dont la question est la négation de celle de Π. On considère le problème de décision Π dont l'instance est un entier naturel N et la question est : “N est-il pair ?” Ce problème de décision appartient-il à la classe

(1) P ?

(2) NP ?

(3) Co-NP (i.e. $\overline{\Pi}$ est dans NP) ?

(4) NP-complet ?

Exercice 4. [5pts] Soit Π et Π' deux problèmes de décision. On rappelle que $\Pi \propto \Pi'$ s'il existe une transformation polynomiale des instances I de Π en instances J de Π' et telle que $I \in Y_\Pi \Leftrightarrow J \in Y_{\Pi'}$. On rappelle qu'un problème de décision Π est dit NP-complet s'il appartient à la classe NP et que $\forall \Pi' \in$ NP, $\Pi' \propto \Pi$.

On rappelle qu'une clause $C = u_1 u_2 \dots u_k$ à k littéraux u_i définis sur un ensemble $X = \{x_1, x_2, \dots, x_n\}$ de variables booléennes, s'interprète comme la disjonction de ces littéraux :

$$u_1 \vee u_2 \vee \dots \vee u_k.$$

Une clause C est dite *satisfaisable* si l'on peut affecter une valeur de vérité à chacune des variables de X, i.e. fixer une *interprétation* $I : X \to \{\top, \bot\}$, telle que l'un au moins des littéraux u_i ait pour valeur $\top$ (*vrai*). On généralise cette notion à un ensemble $\mathscr{C} = \{C_1, C_2, \dots, C_m\}$ de clauses si leur conjonction

$$C_1 \wedge C_2 \wedge \dots \wedge C_m$$

est satisfaisable, autrement dit si chacune des clauses C_i est satisfaisable.

On considère à présent les deux problèmes de décision suivants :

> **Problème de satisfaisabilité** [SAT]
> *Instance :* Un ensemble de n variables booléennes $X = \{x_1, x_2, \dots, x_n\}$ et un ensemble de m clauses $\mathscr{C} = \{C_1, C_2, \dots, C_m\}$.
> *Question :* Existe-t-il $I : X \to \{\top, \bot\}$ telle que $\mathcal{C}$ est satisfaisable ?

> **Problème de 3-satisfaisabilité** [3-SAT]
> *Instance :* Un ensemble de n variables booléennes $X = \{x_1, x_2, \dots, x_n\}$ et un ensemble de m clauses $\mathscr{C} = \{C_1, C_2, \dots, C_m\}$ à 3 littéraux.
> *Question :* Existe-t-il $I : X \to \{\top, \bot\}$ telle que $\mathcal{C}$ est satisfaisable ?

Exemple 1 ▷ $X = \{x_1, x_2, x_3, x_4\}$ et $\mathscr{C} = \{x_1\overline{x_2},\ x_1x_2\overline{x_4},\ \overline{x_1}x_2x_3\overline{x_4}\}$ est une instance de SAT mais pas de 3-SAT (la 1ère clause ne contient que deux littéraux et la 3ème clause en contient 4). ◁

On se propose de démontrer que 3-SAT est NP-complet. Il faut d'une part prouver que 3-SAT appartient à la classe NP et d'autre part trouver une transformation polynomiale des instances I de SAT en instances J de 3-SAT telle que I est satisfaisable si et seulement si J est satisfaisable. Il faut donc construire un ensemble de variables X' et un ensemble de clauses $\mathscr{C}'$ à trois littéraux à partir du couple $(X, \mathscr{C})$. Pour cela, on va :

(1) conserver les variables de X dans X',

(2) conserver les clauses à 3 littéraux de $\mathscr{C}$ dans $\mathscr{C}'$,

(3) remplacer chaque clause C de $\mathscr{C}$ qui ne contient pas exactement 3 littéraux par un ensemble $f(C)$ de clauses de $\mathscr{C}'$ à trois littéraux en rajoutant si nécessaire de nouvelles variables booléennes à X'.

1. **1.** [0,5pt] Quelle est la réponse au problème SAT pour l'instance de l'exemple ?
2. **2.** [0,5pt] Démontrez que 3-SAT appartient à NP.
3. **3.** [1pt] Considérons une clause à un seul littéral $C = u$. On se donne deux nouvelles variables v et w et on considère l'ensemble des clauses

$$f(C) = \{uvw, uv\overline{w}, u\overline{v}w, u\overline{v}\overline{w}\}.$$

Montrez que C est satisfaisable si et seulement si $f(C)$ est satisfaisable. Indication : vous pouvez utiliser l'algèbre de Boole pour réduire l'expression logique à satisfaire.

4. **4.** [1pt] En vous inspirant de la question précédente, trouvez comment transformer une clause à deux littéraux $C = uv$ en un ensemble de clauses à trois littéraux $f(C)$ tel que C est satisfaisable si et seulement si $f(C)$ est satisfaisable.
5. **5.** [1pt] Considérons une clause $C = u_1u_2 \dots u_k$ à k littéraux, $k > 3$. On se donne $k - 3$ variables $v_1, v_2, \dots, v_{k-3}$ et l'ensemble $f(C)$ de $k - 2$ clauses à trois littéraux suivant :

$$\underbrace{u_1u_2\boldsymbol{v}_1}_{C_1},\ \underbrace{\overline{\boldsymbol{v}}_1u_3\boldsymbol{v}_2}_{C_2},\ \underbrace{\overline{\boldsymbol{v}}_2u_4\boldsymbol{v}_3}_{C_3},\ \dots, \underbrace{\overline{\boldsymbol{v}}_{k-4}u_{k-2}\boldsymbol{v}_{k-3}}_{C_{k-3}},\ \underbrace{\overline{\boldsymbol{v}}_{k-3}u_{k-1}u_k}_{C_{k-2}}.$$

Montrez que C est satisfaisable si et seulement $f(C)$ est satisfaisable.

6. **6.** [1pt] Montrez que la transformation $(X, \mathscr{C}) \mapsto (X', \mathscr{C}')$ ainsi construite est polynomiale et concluez.