Corrigé de la PC 4

18 février 2004

1. Le problème MAX2SAT

i) La façon la plus simple d'énumérer les différentes possibilités consiste à remarquer que la formule est symétrique en x, y, z. On peut donc se contenter de distinguer selon le nombre de valeurs 1 (= vrai) parmi ces trois littéraux.

nb. 1	t	nb sat.
0	0	6
0	1	4
1	0	7
1	1	5
2	0	7
2	1	5
3	0	6
3	1	7

Il s'ensuit en particulier que le nombre de clauses satisfaires est au plus 7, et qu'il existe une valeur de t telle que ce nombre soit 7 si et seulement si la clause $x \vee y \vee z$ est satisfaite.

ii) On va réduire polynomialement 3SAT à MAX2SAT. Soit une formule de la forme $f_1 \wedge f_2 \wedge \ldots \wedge f_n$, où chaque f_n dépend des variables $x_1, \ldots, x_m$ et est de la forme $x_{i_1} \vee x_{i_2} \vee x_{i_3}$.

On crée alors des variables t_i, et l'on remplace f_i par g_i donnée par

$$x_{i_1} \wedge x_{i_2} \wedge x_{i_3} \wedge t_i \wedge (\overline{x_{i_1}} \vee \overline{x_{i_2}}) \wedge (\overline{x_{i_2}} \vee \overline{x_{i_3}}) \wedge (\overline{x_{i_3}} \vee \overline{x_{i_1}}) \wedge (x_{i_1} \vee \overline{t_i}) \wedge (x_{i_2} \vee \overline{t_i}) \wedge \overline{(}x_{i_3} \vee \overline{t_i}).$$

Il est clair à partir de la question 1 que si la formule initiale est satisfiable, il existe un choix des valeurs des t_i tel qu'exactement $7n$ des clauses de $g_1 \wedge \ldots \wedge g_n$ soient satisfaites. À l'inverse, si l'on dispose d'un choix de valeurs des variables x_k et t_i tel que $7n$ clauses soient satisfaites, la question 1 montre que chacune des clauses f_i est satisfaite par cette affectation de variables. Résoudre 3SAT sur la formule initiale équivaut donc à résoudre MAX2SAT sur le problème construit.

Qui plus est, le nombre de variables et de clauses dans le problème ainsi construit est linéaire, donc polynomial, en le nombre de clauses et de variables initial. On a donc bien construit une réduction polynomiale de 3SAT à MAX2SAT.

Comme de plus MAX2SAT est dans NP (la donnée d'une affectation de valeurs aux variables fournit un certificat vérifiable en temps linéaire), il s'ensuit que MAX2SAT est NP-complet.

2. Ordonnancement de tâches

Le problème CLIQUE consiste à décider s'il existe un sous-ensemble des sommets de taille k et tel que toutes les arêtes soient présentes entre toute paire de sommets de ce sous-ensemble. Ce problème est clairement dans NP, et est NP-complet pr réduction à partir de STABLE (via passage au graphe complémentaire).

i) Soit $e = \{u, v\} \in E$. La décomposition est une partition de $X \cup E$, donc u figure dans l'un des ensembles de la décomposition, disons S_i, ainsi que e, disons dans S_j. On a $(u, e) \in A$, donc nécessairement par définition $1 \leq i < j \leq$. En particulier, $j \neq 1$ et $i \neq 3$, ce qui signifie $S_1 \cap E = S_3 \cap X = \emptyset$.

La même inégalité montre que $j = 2 \Rightarrow i = 1$, donc $S_2 \cap E$ est un sous-ensemble de $\{\{u, v\}, u \in S_1, v \in S_1\}$; cet ensemble est de cardinal $k(k-1)/2$, d'où $|S_2 \cap E| \leq k(k-1)/2$. Comme de fait $S_2 \cap E = \{\{u, v\}, u \in S_1, v \in S_1\} \cap E$, on voit que l'égalité n'a lieu que si $\{\{u, v\}, u \in S_1, v \in S_1\} \subset E$, ce qui signifie que toutes les arêtes entre éléments de $S_1 \cap X$ sont dans E, ou encore que $S_1 \cap X$ est une clique de G.

ii) On suppose Y, Z et T disjoints ; à partir de là, le calcul est simple : le nombre d'arcs est $|A| + pq + qr$. Or A contient deux éléments par arête de G, donc $|A| = 2m$.

Soit C une clique de taille k de G. On peut alors construire la décomposition compatible de Γ : $S_1 = C$, $S_2 = (C \times C) \cup (X - C)$, $S_3 = E - (C \times C)$; on peut en déduire une décomposition compatible de Γ' sous la forme $S_1 \cup Y$, $S_2 \cup Z$, $S_3 \cup T$. Dans ce cas, les cardinaux des ensembles sont $k + p, k(k-1)/2 + n - k + q, m - k(k-1)/2 + r$, et pour tout $l \geq \max(k, k(k-1)/2 + n - k, m - k(k-1)/2)$ on peut choisir p, q, r de sorte que la décomposition compatible de Γ' soit en trois sous-ensembles de même taille l.

Inversement, supposons que S admette une décomposition compatible en trois sous-ensembles de taille l. Un argument analogue à celui de la question i) montre que nécessairement $Y \subset S_1$, $Z \subset S_2$ et $T \subset S_3$. Par suite, $|S_1 \cap X| = l - p = k$, et $|S_2 \cap X| + |S_2 \cap E| = l - q = k(k-1)/2 + n - k$, d'où, vu $S_3 \cap X = \emptyset$, $|S_2 \cap E| = k(k-1)/2$ et $S_1 \cap X$ est une clique de taille k de G.

iii) ORD-TACHES est manifestement dans NP : la donnée d'une décomposition compatible, qui est de taille polynomiale, fournit un certificat vérifiable en temps polynomial, puisqu'il suffit, pour chaque i et chaque sommet de S_i d'énumérer les arêtes (x, y) issues de x et de vérifier que y est dans un ensemble S_j avec $j > i$; le coût est, au pire, un parcours de A et un parcours de S pour chaque sommet, donc $O(n^2 m)$. La question ii) montre que CLIQUE se réduit à un cas particulier de ORD-TACHES, celui où l'on fixe $t = 3$ et $l = |S|/3$; cette instance, et donc le problème ORD-TACHES plus généralement, est donc bien NP-complet.

Remarque : l'ajout des ensembles Y, Z, T peut paraître artificiel, mais il permet de se ramener à l'énoncé initial d'ORD-TACHES. La construction de Γ permettait de montrer qu'il existait une clique de taille k si et seulement s'il existait une décomposition compatible avec $|S_1| = k$ et $|S_2| = k(k-1)/2 + n - k$, ce qui ne se réduit pas de façon aisée à l'existence d'une décomposition compatible dont tous les ensembles sont de taille $\leq l$. Ajouter Y, Z et T a fourni le degré de liberté nécessaire.

3. Soit $(G = (V, E), k)$ une instance de COUVRANT-MIN (le graphe G est non orienté). Posons $S = X$, $C = E$. La question HITTING-SET(S, C, k) est vraie si et seulement s'il existe $S' \subset S$, $|S'| \leq k$, tel que $\forall x \in E, x = \{u, v\}$ on a $u \in S'$ ou $v \in S'$, ce qui est exactement COUVRANT-MIN(G, k). On a donc réduit COUVRANT-MIN à HITTING-SET. Comme ce dernier est manifestement dans NP (la donnée de S' constitue un certificat de taille $|S'| \leq |S|$, donc polynomiale, qui se vérifie en énumérant les ensembles de C et en parcourant S' pour chaque élément, soit en temps $O(\text{taille}(C)|S'|))$, il est donc bien NP-complet.

4. Soit $(G = (V, E), k)$ une instance de STABLE (le graphe G est non orienté). Construisons le graphe $H = (\{1, \ldots, k\}, \emptyset)$. La question ISO-SOUS-GRAPHE(H, G) revient à demander s'il existe $\phi : \{1, \ldots, k\} \to V$ tel que pour tout k $(\phi(u), \phi(v)) \notin E$, c'est-à-dire à STABLE$(G, k)$. Comme l'isomorphisme de sous-graphe est manifestement dans NP (la donnée du graphe de ϕ fournit un certificat de taille $O(|H|)$, vérifiable en temps $O(|H||E|)))$, et qu'on lui a réduit STABLE, ce problème est donc NP-complet.

5. i. L'assertion $x = y$ équivaut à $(x \vee \overline{y}) \wedge (y \vee \overline{x})$; l'assertion $x = \overline{y}$ équivaut à $(x \vee y) \wedge (\overline{y} \vee \overline{x})$. L'assertion $x = y \vee z$ se réécrit donc $(x \vee (\overline{y} \wedge \overline{z})) \wedge (\overline{x} \vee y \vee z) = (x \vee \overline{y}) \wedge (x \vee \overline{z}) \wedge (\overline{x} \vee y \vee z)$. Enfin, $x = y \wedge z$ devient $(x \vee \overline{y} \vee \overline{z}) \wedge (\overline{x} \vee (y \wedge z)) = (x \vee \overline{y} \vee \overline{z}) \wedge (\overline{x} \vee y) \wedge (\overline{y} \vee z)$.

ii. Étant donné une instance de CIRCUIT-SAT, on construit une instance de SAT de la façon suivante : on associe une variable a_s à chaque sommet de l'instance de CIRCUIT-SAT.

On associe à chaque constante représentée par une variable x_i la clause x_i si la constante vaut 1, et la clause $\overline{x_i}$ si la clause vaut 0.

On associe ensuite à chaque arc une clause du type :

- si l'arc relie un sommet d'étiquette $\vee$ représenté par une variable x à deux sommets représentés par des variables y et z, la clause $x = y \vee z$, réécrite comme en i) ;
- si l'arc relie un sommet d'étiquette $\wedge$ représenté par une variable x à deux sommets représentés par des variables y et z, la clause $x = y \wedge z$, réécrite comme en i) ;
- si l'arc relie un sommet d'étiquette $\neg$ représenté par une variable x à un sommet représenté par une variable y, la clause $x = \overline{y}$, réécrite comme en i).

Par construction, si l'on substitue aux feuilles correspondant à des littéraux des valeurs arbitraires, les clauses représentent le calcul d'évaluation de la fonction booléenne. La conjonction de toutes ces clauses est donc une représentation, sous forme CNF (i.e., de type SAT), de la fonction boolénne calculée par le circuit. On a donc bien réduit CIRCUIT-SAT à SAT.