Leçon 7.
Théoréme de Cook

Denis Trystram

October 3, 2012

Motivation : C'est le premier problème qui a été montré comme appartenant à la classe $\mathcal{NP}$-complet. Cook a établi l'existence d'un problème maximal dans $\mathcal{NP}$ au sens de la réduction polynomiale.

Nous allons montrer dans cette section l'existence d'un premier problème de $\mathcal{NP}$-complet. Ce premier problème a avoir été démontré $\mathcal{NP}$-complet, par Cook, est un problème de logique propositionnelle : Satisfiability (SAT). Rappelons l'énoncé de ce problème :

SAT
Instance. Un ensemble de clauses $C = \{C_1, \ldots, C_m\}$
Question. La formule $C_1 \wedge \ldots \wedge C_m$ est-elle satisfiable?

(Cook's Theorem) SAT $\in \mathcal{NP}$-complet

Il est facile de vérifier que la donnée d'une fonction d'interprétation pour les variables booléennes satisfait ou non la formule logique, donc SAT $\in \mathcal{NP}$, il nous faut maintenant prouver que :
$\forall L \in \mathcal{NP}$ il existe une réduction polynomiale de L vers SAT.

La difficulté vient du fait qu'il faut prouver l'existence d'une réduction pour tous les langages L de $\mathcal{NP}$. La seule indication dont on dispose est l'existence d'une TM non déterministe qui accepte L en temps polynomial.

Reprécisons tout d'abord la notion de temps d'exécution sur une NTM non déterministe sur un mot $w \in L$: c'est le minimum des temps d'exécution parmi toutes les exécutions acceptant w. L'idée de la preuve du théorème de Cook est de **coder l'exécution d'une NTM T reconnaissant L comme**

une formule logique. Nous allons montrer pour cela comment coder les données de L, toutes les informations sur le ruban de T, mais aussi les états et les transitions.

Montrons l'existence d'une transformation qui à chaque mot w et tout langage (problème) $L \in \mathcal{NP}$ associe une instance L_{SAT} qui est positive si et seulement si $w \in L$. On considère une TM non déterministe $(Q, \Gamma, \Sigma, \Delta, \square, q_0, q_A)$ dont la complexité est bornée par un polynôme $p(n)$ et w un mot de Σ^* de longueur n. Cette machine accepte le mot w si et seulement si il existe une exécution de TM sur w de longueur au plus $p(n)$ qui mène vers un état accepteur. Une telle exécution peut être représentée par $p(n)$ configurations successives de TM. Chaque configuration est caractérisée par l'état, le contenu du ruban et la position de la tête de lecture. On propose de la représenter par un ensemble de tableaux :

- Un tableau R à deux dimensions de taille $(p(n)+1)$ par $(p(n)+1)$ qui indique pour chaque configuration et chaque case du ruban le symbole de Γ se trouvant dans cette case. On peut noter ici que le nombre de cases visitée est au plus $p(n) + 1$ car on visite au plus une nouvelle case à chaque transition.

- Un tableau Q mono-dimensionnel de taille $(p(n) + 1)$ donnant l'état de chaque configuration i.

- Un tableau P mono-dimensionnel de taille $(p(n)+1)$ donnant la position de la tête de lecture dans chaque configuration i.

- Enfin, un tableau C mono-dimensionnel de taille $(p(n) + 1)$ donnant le choix non déterministe effectué par la machine à chaque étape. On note r le nombre maximum de ces choix.

  Ce tableau a un statu un peu particulier, il n'est pas strictement nécessaire à la description logique de l'exécution, mais sera utile pour vérifier que le contenu des tableaux définit bien une exécution valide.

La transformation produit une formule logique qui est satisfaite que pour un contenu de ces tableaux qui définit une exécution acceptant w. On introduit maintenant une variable propositionnelle par case des tableaux (il y en a un nombre polynomial $O(p(n)^2)$:

- $r_{ij\sigma}$ pour $0 \leq i, j \leq p(n)$ et $\sigma \in \Gamma$. Cette valeur est à 1 si le symbole σ est dans la case (i, j).

- q_{ik} pour $0 \leq i \leq p(n)$ et $k \in Q$. Cette valeur est 1 si l'état de la configuration i est k et 0 sinon.
- p_{ij} pour $0 \leq i, j \leq p(n)$. Cette valeur est à 1 si la tête de lecture pointe sur la case j.
- c_{is} pour $0 \leq i \leq p(n)$ et $1 \leq s \leq r$ si la fonction de transition s'effectue sur le choix s à l'étape i.

Ces formules n'ont de sens que s'il n'existe qu'un seul symbole de Γ à un moment donné sur une case, qu'un seul état et que la tête de lecture ne pointe que sur une seule case. Il est donc nécessaire de restreindre le nombre de fonctions d'interprétation possibles.
Tout d'abord, exprimons que chaque case ne contient qu'un seul symbole de Γ :
$r_{ij\sigma} \Longrightarrow \overline{r_{ij\sigma'}}\ \forall i, j$ pour $\sigma \neq \sigma'$. ce qui se traduit en forme normale conjonctive par la formule $(\bigwedge_{\sigma \neq \sigma'}(\overline{r_{ij\sigma}} \vee \overline{r_{ij\sigma'}})$. de plus, $(\bigvee_{\sigma \in \Gamma} r_{ij\sigma})$ exprime que le contenu de la case est bien un symbole de Γ, ainsi, il suffit d'exprimer pour toutes les cases les deux conditions à la fois :

$$\bigwedge_{0 \leq i,j \leq p(n)} \left(\bigwedge_{\sigma \neq \sigma'} (\overline{r_{ij\sigma}} \vee \overline{r_{ij\sigma'}}) \wedge \left(\bigvee_{\sigma \in \Gamma} r_{ij\sigma} \right) \right)$$

On vérifie aisément que cette formule est sous forme normale conjonctive et que sa longueur est polynomiale, précisément, elle est dans $O(p(n)^2)$.
Il faut exprimer de la même manière que l'on a qu'une seule configuration à la fois, que la tête de lecture ne pointe que sur une seule case et qu'il n'y a qu'un seul choix possible à un moment donné pour une transition. On donne ci-dessous la formule logique qui concerne le tableau P des positions de la tête de lecture dont la longueur est dans $O(p(n)^3)$:

$$\bigwedge_{0 \leq i \leq p(n)} \left(\bigwedge_{0 \leq j,j' \leq p(n) j \neq j'} (\overline{p_{ij}} \vee \overline{p_{ij'}}) \wedge \left(\bigvee_{0 \leq i \leq p(n)} p_{ij} \right) \right)$$

Il reste maintenant à exprimer sous forme logique que toute fonction d'interprétation définit bien une exécution de la machine acceptant le mot w :

- **La première configuration est la configuration initiale**. Ceci s'exprime par les conditions que les n caractères du mot d'entrée w figure sur le ruban au départ (le reste étant le mot blanc), que la position de la tête de lecture soit sur la première case du mot et que

l'état soit l'état initial. Ici, nous avons supposé pour simplifier (sans perte de généralité) que le mot était rangé à partir de la position $j = 0$, ce qui est un peu abusif car les déplacements de la tête de lecture peuvent se faire aussi bien vers la droite que vers la gauche. Ceci ne change évidemment pas fondamentalement la formule donnée (on pourrait par exemple considérer un tableau R plus large ($-p(n)$ à $+p(n)$)i).

$$[\bigwedge_{0\leq j\leq n-1} r_{0jw_{j+1}} \wedge \bigwedge_{n\leq j\leq p(n)} r_{0j\square}] \wedge q_{00} \wedge p_{00}$$

- **Le passage d'une étape à la suivante est bien conforme à la fonction de transition**.

  Il nous faut exprimer deux conditions : les cases du ruban qui ne sont pas placées sous la tête de lecture ne sont pas modifiées et (ce qui est le plus délicat) que le passage d'une configuration à la suivante est bien conforme à la transition (symbole, état et choix) :

  Tout d'abord, précisons la formule qui exprime que les cases non placées sous la tête de lecture ne sont pas modifiées :

$$\bigwedge_{c0\leq i\leq p(n)0\leq j\leq p(n)\sigma\in\Gamma} [(r_{ij\sigma} \wedge \overline{p_{ij}}) \Longrightarrow r_{(i+1)j\sigma}]$$

  que l'on transforme en forme normale conjonctive :

$$\bigwedge(\overline{r_{ij\sigma}} \vee p_{ij} \vee r_{(i+1)j\sigma})$$

  Détaillons maintenant la formule qui atteste de la conformité du passage de la configuration i à $i+1$ (qui passe de l'état k à l'unique état k', du symbole σ à σ'). La variable d représente le déplacement de la tête de lecture ($d = -1, 0, +1$) donné de façon unique par la relation de transition.

$$\bigwedge_{c0\leq i,j\leq p(n)\sigma\in\Gamma 1\leq s\leq r} [(r_{ij\sigma} \wedge q_{ik} \wedge p_{ij} \wedge c_{is} \wedge \overline{q_{(i+1)k'}})$$

$$\wedge(r_{ij\sigma} \wedge q_{ik} \wedge p_{ij} \wedge c_{is} \wedge \overline{r_{(i+1)j\sigma'}}) \wedge (r_{ij\sigma} \wedge q_{ik} \wedge p_{ij} \wedge c_{is} \wedge \overline{p_{(i+1)(j+d)}})$$

- **Finalement, on exprime que l'on atteint bien l'état accepteur au bout de $p(n)+1$ étapes :** $\bigvee_{i \neq p(n)} q_{i,q_A}$

Toutes ces transformations sont polynomiales. La formule finale est obtenue comme la conjonction de toutes les formules précédentes.