Le théorème de COOK

Le théorème de Cook

NOM : SAT (satisfiabilité)

DONNEES : une formule sous FNC

QUESTION : est-ce que la formule est satisfiable ?

Théorème (de Cook, 1970) :

SAT **est** NP**-complet.**

Un exercice en logique

Soient $v_1, v_2, \ldots, v_m$ **des variables logiques. Nous souhaitons utiliser la formule logique suivante :** *exactement_un*$(v_1, v_2, \ldots, v_m)$ **qui est vraie si et seulement si une (et une seule) des variables est vraie.**

Proposition : *exactement_un*$(v_1, v_2, \ldots, v_m)$ **peut s'écrire en forme normale conjonctive, avec une longueur** $O(m^2)$**.**

Preuve :

Notons que

$exactement\_un(v_1, v_2, \ldots, v_m) =$

$au\_moins\_un(v_1, v_2, \ldots, v_m) \wedge au\_plus\_un(v_1, v_2, \ldots, v_m)$

et $au\_moins\_un(v_1, v_2, \ldots, v_m) = \vee_{i=1,\ldots,m} v_i$

et $au\_plus\_un(v_1, v_2, \ldots, v_m) = \wedge_{1 \leq i < j \leq m}(\neg v_i \vee \neg v_j)$

Ce qui donne une longueur totale de

$$m+m(m-1)/2=m(m+1)/2$$

CQFD

Preuve du th. de COOK

Nous avons deux tâches :

1. **Prouver que SAT $\in$NP**
2. **Prouver que tout problème $\Pi\in$NP,**

$$\Pi \propto \text{SAT}$$

c.a.d. que *SAT est NP-difficile*

Pour conclure, que SAT est un problème « le plus difficile » dans NP.

SAT $\in$ NP

Machine non-déterministe à deux bandes

i) **on associe une valeur de vérité aux variables de façon non-déterministe (en temps O(n^2) - ce qui correspond à un passage pour chaque variable)**

ii) **on évalue la formule ainsi obtenue en temps linéaire**

iii) **si la formule est vraie, on passe dans l'état final d'acceptation.**

La NP-difficulté

Soit $\Pi \in$ NP. Il faut prouver que ce problème peut être réduit polynomialement à SAT

Par la définition de NP, il existe une Machine de Turing M, non-déterministe, qui accepte une donnée de longueur *n* en temps *p*(*n*). Nous pouvons supposer qu'il s'agit d'une machine à une bande.

La NP-difficulté (2)

Soient

q_0, q_1, ... , q_s les états de la machine, avec

q_0 l'état initial

q_r, q_{r+1}, ... , q_s finaux (états d'acceptation)

Soient

l_0, l_1, ... , l_u les lettres de l'alphabet de M, avec

l_0 le *blanc*

La NP-difficulté (3)

Pour simplifier notre travail, nous modifions la machine de sorte qu’une fois un état d’acceptation atteint, elle reste «stationnaire» dans cet état. Ainsi, il n’est plus utile de vérifier après chaque transition si la machine a terminé, il suffit de faire $p(n)$ étapes de calcul et vérifier ensuite.

La NP-difficulté (4)

La table de transition de la machine nous donne une valeur m des nombres de lignes de la table

Soit $x_1, x_2, \ldots , x_n$ une instance de longueur n du problème Π.

Nous allons construire une formule logique $F(x_1, x_2, \ldots , x_n)$, de longueur polynomiale en n, qui est satisfiable si et seulement si la machine **M** accepte l'instance $x_1, x_2, \ldots , x_n$.

Les variables utilisées

Contenu de la bande : **O**$((p(n))^2)$

$s_{i,j,t}$ **avec**

$$-p(n) \leq i \leq p(n)$$

$$0 \leq j \leq u$$

$$0 \leq t \leq p(n)$$

Si $s_{i,j,t}$ **est vraie, c'est que la case** i **de la bande contient la lettre** l_j **au temps** t**.**

Les variables utilisées

L'état de la machine : **O($p(n)$)**

$z_{i,t}$ avec

$$0 \leq i \leq s$$

$$0 \leq t \leq p(n)$$

La vérité de $z_{i,t}$ signifie qu'au temps t la machine se trouve dans l'état q_i.

Les variables utilisées

La tête de lecture/écriture : **O((*p(n)*)^2)**

$h_{i,t}$ **avec**

$$-p(n) \leq i \leq p(n)$$

$$0 \leq t \leq p(n)$$

Si $h_{i,t}$ **est vraie, c'est que la tête de lecture/écriture se trouve sur la case** i **de la bande au temps** t**.**

Les variables utilisées

Les transitions : **O($p(n)$)**

$b_{i,t}$ avec

$$1 \leq i \leq m$$

$$1 \leq t \leq p(n)$$

Si $b_{i,t}$ est vraie, c'est que pour passer du temps t-1 au temps t nous avons choisi la $i^{\text{ième}}$ ligne de la table de transition.

Les variables utilisés

Contenu de la bande :	$s_{i,j,t}$	$O((p(n))^2)$
L'état de la machine :	$z_{i,t}$	$O(p(n))$
La tête de lecture/écriture :	$h_{i,t}$	$O((p(n))^2)$
Les transitions :	$b_{i,t}$	$O(p(n))$

Total $O((p(n))^2)$

Une transition

Une transition représente donc une ligne de la table des transitions, de la forme :

ligne	lecture	état	écriture	déplacement	nouvel état
i	$lect_i$	$état_i$	$écrit_i$	$dépl_i$	$nétat_i$

avec $1 \leq i \leq m$, $lect_i$ **et** $écrit_i$ **dans l'intervalle** $[0...u]$, $état_i$ **et** $nétat_i$ **dans l'intervalle** $[0...s]$ **et** $dépl_i$ **dans** $\{-1,0,1\}$.

La formule

Formule = Consistance $\wedge$ Initialisation $\wedge$ Transition $\wedge$ Terminaison

Consistance

Consistance = Cons(bande) $\wedge$ Cons(état) $\wedge$ Cons(tête) $\wedge$ Cons(transition)

Consistance(bande)

$$\bigwedge_{-p(n) \leq i \leq p(n)}$$

$$\left(\bigwedge_{0 \leq t \leq p(n)} \text{exactement\_un}(s_{i,0,t}, s_{i,1,t}, \ldots, s_{i,u,t})\right)$$

Longueur de la formule :

$\big(2p(n)+1\big)\big(p(n)+1\big)u(u+1)/2$ **c.a.d.** $O((p(n))^2)$

Consistance(état)

$$\bigwedge_{0\leq t \leq p(n)} \textbf{exactement\_un}(z_{0,t}, z_{1,t}, \ldots, z_{s,t})$$

Longueur de la formule :

$\left(p(n)+1\right)s(s+1)/2$ **c.a.d.** $\mathbf{O}\left(p(n)\right)$

Consistance(tête)

$$\bigwedge_{0\leq t \leq p(n)} \text{exactement\_un}(h_{-p(n),t}, h_{-p(n)+1,t}, \ldots, h_{p(n),t})$$

Longueur de la formule :

$\big(p(n)+1\big)\big(2p(n)+1\big)\big(p(n)+1\big)$ **c.a.d.** $O((p(n))^3)$

Consistance(transition)

$$\bigwedge_{1\leq\, t\, \leq p(n)} \text{exactement\_un}(b_{1,t}\,,\, b_{2,t}\,,\, \ldots,\, b_{m,t}\,)$$

Longueur de la formule :

$\big(p(n)+1\big)m(m+1)/2$ **c.a.d.** $\mathrm{O}\big(p\,(n)\big)$

Consistance

Cons(bande)	$O((p(n))^2)$
Cons(état)	$O(p(n))$
Cons(tête)	$O((p(n))^3)$
Cons(tran)	$O(p(n))$
Consistance	$O((p(n))^3)$

Initialisation (longueur $O(p(n))$)

- les n cases de la bande contenant les données soit correctes (c.a.d. la case i contient x_i qui est la lettre l_i)
- les autres cases contiennent des blancs
- la tête se trouve sur la case du milieu (0)
- l'état initial soit q_0

$$\bigwedge_{-p(n) \leq i \leq 0} s_{i,0,0} \ \bigwedge_{1 \leq i \leq n} s_{i,l_i,0} \ \bigwedge_{n+1 \leq i \leq p(n)} s_{i,0,0} \ \wedge$$

$$\wedge\ z_{0,0} \ \wedge\ h_{0,0}$$

Terminaison

$$z_{r,p(n)} \vee z_{r+1,p(n)} \vee \cdots \vee z_{s,p(n)}$$

Longueur **O(1)**

Transition

$$\text{Transition} = \bigwedge_{1 \leq t \leq p(n)} \text{Trans}(t)$$

Trans(t)

$$\bigwedge_{-p(n) \le i \le p(n)}$$

$$\left(\bigwedge_{0 \le j \le u} (h_{i,t-1} \vee \neg s_{i,j,t-1} \vee s_{i,j,t}) \wedge BTrans(i,t)\right)$$

BTrans(i,t)

$$\bigwedge_{1 \le k \le m}$$

$$\Big\{\big(h_{i,t-1} \wedge b_{k,t}\big)$$

$$\Rightarrow$$

$$\big(z_{\text{état}_k,t-1} \wedge z_{\text{nétat}_k,t} \wedge s_{i,\text{lect}_k,t-1} \wedge s_{i,\text{écrit}_k,t} \wedge h_{i+\text{dépl}_k,t}\big)\Big\}$$

BTrans(i)

$$\bigwedge_{1 \leq k \leq m}$$

$$\{\neg\, h_{\mathrm{i,t\text{-}1}} \vee \neg\, b_{\mathrm{k,t}} \vee \left(z_{\mathrm{t\text{-}1,état}_k} \wedge z_{\mathrm{t,nétat}_k} \wedge s_{\mathrm{i,lect}_k\mathrm{,t\text{-}1}} \wedge s_{\mathrm{i,écrit}_k\mathrm{,t}} \wedge h_{\mathrm{i+dépl}_k\mathrm{,t}}\right)\}$$

Un problème

La dernière formule n'est pas en FNC !!!!!!

En effet, nous avons

$$\wedge\left\{a \vee b \vee \left(c \wedge d \wedge e \wedge f \wedge g\right)\right\}$$

Ce qui se traduit en

$$\wedge\left\{ \left(a \vee b \vee c\right) \wedge \left(a \vee b \vee d\right) \wedge \left(a \vee b \vee e\right) \wedge \left(a \vee b \vee f\right) \wedge \left(a \vee b \vee g\right)\right\}$$

Transition

La longueur totale des formules de transition est en $O((p(n))^2)$.

Longueur totale

Consistance	$O((p(n))^3)$
Initialisation	$O(p(n))$
Terminaison	$O(1)$
Transition	$O((p(n))^2)$
Formule	$O((p(n))^3)$

Ceci complète la preuve du théorème de COOK