Algorithmique Distribuée (2)

- $disj_n(x,y) = \begin{cases} 1 \Leftrightarrow x \text{ et } y \text{ disjoints} \\ 0 \text{ sinon} \end{cases}$ (pas de 1 en commun)
- $script_\Pi(x,y) = (m_0, \dots, m_b)$
- Matrice de f

	y	y'
x	...	
x'		1
x	1	

si m scripts

Propriété des rectangles

- **Prop. 4.3** Si F_0, F_1 sont des ens. discriminants alors le nombre de scripts pour TOUT protocole est au moins $|F_0| + |F_1|$.

Ens. Discriminant $F_v \subseteq \{0,1\}^n \times \{0,1\}^n$
$\forall (x,y),(x',y') \in F_v,\ f(x,y) = f(x',y') = v$
et $\forall z \in \{f(x,y), f(x',y')\}$

- Pour $disj_n$, $2^n + n$ scripts sont nécessaires.
(Ens. discriminants $|F_0| \geq n$ et $|F_1| \geq 2^n$)

Objectif : Lien entre # scripts et complexité de communication (coût de communication)

Rappel : coût de com. de Π $= \max_{(x,y)} \sum_{i=0}^{b} |m_i|$

⚠ Complexité de com. ne peut pas être définie comme = coût minimum d'un protocole de com. (pour f)

$\exists \Pi_g$ qui a un coût de com. = 2 bits $\forall g\ \forall (x,y)$.

Π_g : Alice et Bob s'échangent tour à tour le mot ε, pendant $x \in [0, 2^n[$ rondes. Puis Alice envoie 0 quand elle a fini. Bob, qui a compté le # rondes, déduit x et transmet $g(x,y)$ à Alice.

$script_{\Pi_g}(x,y) = (\underbrace{\overset{A}{\varepsilon}\overset{B}{\varepsilon}\overset{A}{\varepsilon}\overset{B}{\varepsilon}, \dots, \overset{B}{\varepsilon}}_{2x}, \underbrace{\overset{A}{0}}_{1}, \underbrace{\overset{B}{g(x,y)}}_{1})$

coût $(\Pi_g) = \max_{x,y} \sum |m_i| = 2$.

⚠ Le nombre de scripts pour Π_g vaut 2^{n+1}.

Il faut imposer une propriété sur les messages. Ils doivent être SANS-PREFIXES.

Alice → Bob : 100 (ou m_0 ou 1001 ou m_0') est préfixe de 1001

Alice ← Bob : 11

Est-ce le protocole est fini ?

Propriété SANS-PREFIXE. Π est sans-préfixe si

$\forall$ ronde $i \in \mathbb{N}$, l'ens. des mots

$\mathcal{M}_\Pi(i) = \bigcup_{(x,y)} \{ m_0(x,y) \circ m_1(x, \Pi(x,y)) \circ m_2(x, \Pi(x,y)) \circ \dots \circ m_{\#\text{rondes pour } (x,y)}(x, \Pi(x,y)) \}$

exemple : $\mathcal{M}_\Pi(0)$ = Ens. de tous les messages qu'Alice peut envoyer au départ à Bob.

doit être sans-préfixe.

Ex : $\{0,10,11\}$ est sans-préfixe ‖ $\{0,01,11\}$ pas sans-préfixe
$\{0,1\}$ est sans-préfixe (car mot de taille différente)

Prop. Un message formé de mot $\in$ code ($=$ ens. de mots sans-préfixe) peut être décodé de manière unique (sans ambiguïté)

$M = \underline{10}\,\underline{00}\,\underline{11}\,\underline{0}\,\underline{10}\,\underline{0}\,\underline{10}$

Si $\mathcal{M}_\Pi(0)$ est sans-préfixe alors Bob pourra toujours décoder tout $m_0 \in \mathcal{M}_\Pi(0)$ d'Alice (à la ronde 0).

Exercice : Montrer qu'Alice ou Bob peuvent détecter lorsque le protocole sans-préfixe doit s'arrêter.

Réponse :
- $s_i = m_0, m_1, m_2 \dots m_i$ (Alice m_i → Bob)
- Bob peut calculer $\mathcal{M}_\Pi(i)$ et $\mathcal{M}_\Pi(i+1)$
- Par déf. si $s_i \in \mathcal{M}_\Pi(i)$. Si $s_i \in \mathcal{M}_\Pi(i+1)$?
- Si $s_i \in \mathcal{M}_\Pi(i+1)$ c'est que Π est terminé.
- Sinon $s_i \circ m_{i+1} \in \mathcal{M}_\Pi(i+1)$ et Π continue.

Définition La complexité de communication d'une fonction f est le coût de communication minimum d'un protocole sans-préfixe pour f.

NB : Π_g n'est pas sans-préfixe car ε est préfixe de tout mot. Et donc $\mathcal{M}_{\Pi_g}(2) = \{\overset{\varepsilon}{\varepsilon\varepsilon\varepsilon}\overset{\varepsilon}{\varepsilon\varepsilon}\overset{\varepsilon}{0}, 00, 01\}$ n'est pas sans-préfixe. $\mathcal{M}_{\Pi_g}(0) = \{\varepsilon, 0\}$

Proposition 4.5

Tout protocole de com. pour f de coût k et utilisant $\leq t$ rondes peut être transformé en un protocole sans-préfixe pour f de coût $2(k+t)$ et avec le même nombre de scripts.

Idée : transformation δ → τ [double chaque bit sauf le dernier qu'on inverse]

$\delta(10100) \to \underline{11}\,\underline{00}\,\underline{11}\,\underline{00}\,\underline{01}$ (si mot $\neq \varepsilon$)
$\quad\ 1\ 0\ 1\ 0\ 0.$

$\tau(m_i) = \begin{cases} 1 \circ m_i & \text{si } i = \text{dernière ronde} = r(x,y) \\ 01 & \text{si } i < r(x,y) \text{ et } m_i = \varepsilon \\ 00 \circ \delta(m_i) & \text{sinon} \end{cases}$ □

Proposition 4.6 Soit $\delta(f) = \min_\Pi \left\{ \max_{(x,y) \in \{0,1\}^{2n}} |script_\Pi(x,y)| \right\}$

La complexité de com. de f $CC(f) \geq \log_2 \delta(f)$

Idées :
- On peut supposer que Π minimum est sans-préfixe.
- Inégalité de Kraft sur le codage des mots sans-préfixe.

Ex : $X = \{\underset{\omega_1}{10}, \underset{\omega_2}{11}, \underset{\omega_3}{001}\}$ ⟶ $\sum_{\omega_i \in X} 2^{-l(\omega_i)} \leq 1$

$l(\omega_i)$ = longueur du code de ω_i □

Proposition 4.7

$$CC(disj_n) \geq \log_2(\underbrace{\delta(disj_n)}_{\geq 2^n + n}) \geq \log_2(2^n + n) > n.$$

$\Rightarrow$ $CC(disj_n) \geq n+1$. OPTIMAL à cause du protocole trivial.

Reste à montrer la réduction du calcul distribué de détection de C_3 et C_4 à partir de $CC(disj_n)$.

(X,Y) ⟶ Graphe $G_{X,Y}$ si **A** détecte C_3/C_4 alors on déduit un protocole pour $disj_n$