ALGORITHMIQUE DISTRIBUÉE

MASTER2 INFORMATIQUE

10 décembre 2025

C. Gavoille - 1h20

La langue pour vos réponses peut-être le français ou l'anglais, mais pas les deux! Document: une feuille A4 recto-verso autorisée.

Question de cours

Question 1. *Rappelez les principales caractéristiques des modèles LOCAL, CONGEST et b-CONGEST.*

SOLUTION. **[3 pts |∑3.0]** Les trois modèles sont synchrones, les sommets ont des identifiants, processeurs et liens n'ont pas d'erreurs. Pour LOCAL, les messages sont de taille arbitraires. Pour CONGEST, ils sont limités à $O(\log n)$ bits, où n est le nombre de sommets du graphe G sur lequel est exécuté l'algorithme. Pour b-CONGEST, ils sont limités à b bits.

Question 2. *Expliquez à quoi correspond le problème d'algorithmique distribué de la « détection d'un H » dans le graphe G, pour un graphe H fixé ? [NB : On ne demande pas sa solution, mais les spécificités en terme d'entrées et de sorties.]*

SOLUTION. **[3 pts |∑6.0]** Le problème est le suivant. En entrée on dispose d'un graphe G. Initialement les sommets connaissent leur identifiant et le graphe H (qui est fixé). Il faut trouver un algorithme distribué qui, pour chaque sommet u, calcule une sortie booléenne DETECT(u) avec la propriété suivante : il existe une copie (au moins) de H dans G si et seulement si DETECT(u) = VRAI pour au moins un sommet u de G.

Question 3. *Donnez le principe d'un algorithme dans le modèle LOCAL permettant de détecter un C_6, un cycle de longueur six, dans un graphe G, Précisez le nombre de rondes.*

SOLUTION. **[3 pts |∑9.0]** Trois rondes suffisent.

À la ronde 1, les sommets s'échangent leurs identifiants. Ainsi, après cette ronde, chaque sommet u connaît la liste $L_1(u)$ de ses voisins, c'est-à-dire des sommets qu'il peut atteindre par un chemin de longueur un.

À la ronde 2, les sommets s'échangent l'ensembles de leurs voisins, c'est-à-dire les listes L_1. Ainsi, après cette ronde, chaque sommet u connaît la liste $L_2(u)$ des sommets qu'il peut atteindre par un chemin de longueur deux. Il faut aussi stocker dans $L_2(u)$ le prédécesseurs dans $L_1(u)$ de chaque $u_2 \in L_2(u)$ pour pouvoir reconstruire les chemins $u - u_1 - u_2$.

À la ronde 3, ils s'échangent l'ensemble des voisinages de leurs voisins, c'est-à-dire les listes L_2. Ainsi, après cette ronde, chaque sommet u connaît la liste $L_3(u)$ des sommets qu'il peut atteindre par un chemin de longueur trois. De même, il faut aussi stocker dans $L_3(u)$ le prédécesseurs dans $L_2(u)$ de chaque $u_3 \in L_2(u)$ pour pouvoir reconstruire les chemins $u - u_1 - u_2 - u_3$.

Il existe un cycle C_6 passant par u si et seulement si il existe deux chemins partant de u, disjoints sauf à leurs extrémités, et atteignant un sommet $w \in L_3(u)$. Plus précisément, il existe $u - u_1 - u_2 - w$ et $u - v_1 - v_2 - w$ avec $u_i \neq v_i \in L_i(u)$ et $w \in L_3(u)$, ce que peut vérifier assez facilement u une fois les ensembles $L_1(u)$, $L_2(u)$, $L_3(u)$ déterminés.

Question 4. *En déduire un algorithme de détection de C_6 dans le modèle CONGEST. Précisez sa complexité en temps (nombre de rondes) en fonction du degré maximum Δ du graphe G.*

SOLUTION. **[3 pts |∑12.0]** Dans l'algorithme précédant, seules les rondes 2 et 3 posent problèmes, lorsque u souhaite envoyer $L_1(u)$ puis $L_2(u)$ à ses voisins. On a $|L_1(u)| \leqslant \Delta$ et $|L_2(u)| \leqslant \Delta\cdot(\Delta-1)$. Donc u peut envoyer $L_1(u)$ en Δ rondes, puis $L_2(u)$ en $\Delta\cdot(\Delta-1)$ rondes. Cela fait un total de $1+\Delta+\Delta\cdot(\Delta-1) = 1+\Delta^2 = O(\Delta^2)$ rondes.

Complexité de communication

Question 5. *Expliquez ce qu'est un protocole de communication qui calcule une fonction booléenne f ? Quelles en sont les principales règles ? Quel est l'objectif recherché ?*

SOLUTION. **[3 pts |∑15.0]** C'est un algorithme qui permet à deux participants, Alice et Bob, d'échanger des messages jusqu'à obtenir la valeur $f(x, y)$ où x, y sont respectivement les entrées d'Alice et Bob. Ils s'échangent des messages chacun à leur tour, Alice commençant, le dernier bit étant la réponse. Bien sûr, chacun ne connaît que son entrée, la fonction f, et le protocole. L'objectif est de minimiser le nombre total de bits échangés.

Question 6. *Expliquez le principe du protocole « trivial » pour le calcul de f. Quel est son coût ?*

SOLUTION. **[3 pts |∑18.0]** Il s'agit d'un protocole de communication particulier où Alice transmet un seul message, son entrée x, et où Bob renvoie la réponse $f(x, y)$ attendue. Son coût est de $n + 1$ où n est la taille des entrées.

On considère la fonction $\text{aleq}_n(x, y) \in \{0, 1\}$ (pour *ALmost EQual*) qui étant donnés deux mots binaires x, y, chacun de n bits, renvoie 1 si et seulement si « $|x - y| \leqslant 1$ », c'est-à-dire si les entiers représentés en binaires par les mots x et y diffèrent d'au plus une unité. Par exemple, $\text{aleq}_2(11, 10) = 1$ car $|3 - 2| \leqslant 1$. En revanche, $\text{aleq}_2(01, 11) = 0$ car $|1 - 3| \not\leqslant 1$. Bien évidemment, $\text{aleq}_n(x, x) = 1$ pour tout $x \in \{0, 1\}^n$.

Question 7. *Donnez la matrice de la fonction* aleq_n *pour* $n = 2$.

SOLUTION. **[3 pts |∑21.0]**

x \ y	00	01	10	11
00	1	1	0	0
01	1	1	1	0
10	0	1	1	1
11	0	0	1	1

On rappelle qu'un ensemble discriminant pour une fonction booléenne f et une valeur $v \in \{0, 1\}$ est un ensemble F_v d'entrées (x, y) pour f tel que, pour tout $(x, y) \neq (x', y') \in F_v$, $f(x, y) = f(x', y') = v$ et $\bar{v} \in \{f(x, y'), f(x', y)\}$, où $\bar{v} = 1 - v$ est la valeur complémentaire de v.

Question 8. *Donnez un ensemble discriminant* F_1 *pour* aleq_2 *le plus grand possible et donnez sa cardinalité. Même question pour un ensemble discriminant* F_0.

SOLUTION. **[3 pts |∑24.0]**

x \ y	00	01	10	11
00	1	**1**	**0**	0
01	1	1	**1**	**0**
10	**0**	1	1	**1**
11	0	**0**	1	1

Pour $v = 1$, on peut poser $\mathbf{F_1} = \{(00, 01), (01, 10), (10, 11)\}$. La diagonale inférieure était aussi possible, de même qu'on pouvait remplacer dans F_1 la première entrée par $(00, 00)$ ou la dernière par $(11, 11)$. On remarque que cette diagonale de 1 a des zéros au-dessus. C'est donc bien un ensemble discriminant avec $|\mathbf{F_1}| = 3$. On remarque que tout ensemble F_0 à plus de trois entrées aurait du contenir deux uns du carré 2×2 central de la matrice, ou deux valeurs verticales ou horizontales, rendant non-discriminant l'ensemble.

Pour $v = 0$, on peut poser $\mathbf{F_0} = \{(00, 10), (01, 11), (10, 00), (11, 01)\}$. Pour les deux premières entrées $(00, 10), (01, 11)$ (ou, par symétrie, les deux dernières), on observe qu'on a une valeur 1 mal placée. Pour les entrées $(00, 10), (10, 00)$ (ou leurs symétriques) on a une valeur 1 placée en bas à droite (ou en haut à gauche).

C'est donc bien un ensemble discriminant avec $|\mathbf{F_0}| = 4$. On ne pouvait pas faire plus grand, puisqu'au delà de 2^n deux valeurs verticales ou horizontales apparaissent, rendant non-discriminant l'ensemble.

Question 9. *Expliquez la structure de la matrice de la matrice* aleq_n *pour* $n \geqslant 3$.

SOLUTION. **[3 pts** $|\sum 27.0]$ De manière générale, la matrice de aleq_n possède trois diagonales descendantes de 1 : la diagonales principales (cas $y = x$), puis une juste au-dessus (cas $y = x + 1$) en l'autre juste en dessous (cas $y = x - 1$). En particulier, chaque ligne et chaque colonne possède trois 1 consécutifs, sauf les premières ou les dernières où il y en a que deux.

Question 10. *Rappelez la définition de* $\text{CC}(f)$*, la complexité de communication d'une fonction booléenne* f.

SOLUTION. **[3 pts** $|\sum 30.0]$ C'est le coût minimum d'un protocole de communication **sans-préfixe** calculant f.

Question 11. *Généralisez la question 8 à tout* $n \geqslant 3$*, c'est-à-dire construisez des ensembles discriminants* F_1 *et* F_0 *les plus grands possibles pour* aleq_n *et donnez leurs cardinalités.*

SOLUTION. **[3 pts** $|\sum 33.0]$

Pour $v = 1$, on peut poser $F_1 = \{(x, x+1) : x \in \{0, \dots, 2^n - 2\}\}$. Le triangle supérieur est rempli de zéro car il correspond à des entrées (x, y) avec $y > x + 1$. F_1 est donc discrimant pour la valeur $v = 1$. On a $|F_1| = 2^n - 1$.

Pour $v = 0$, on peut poser, on pouvait garder le même ensemble F_1 que pour la question 8. Mais, pour $n \geqslant 3$, on pouvait aussi choisir $F_0 = \{(x, x+2) : x \in \{0, 1, 2, 3\}\}$ avec $|F_1| = 4$. Il s'agit d'une diagonale contenant quatre 1 avec un triangle inférieur de 0. Attention au fait que chaque colonne ou ligne de la matrice contient au plus trois valeurs à 0, et donc la matrice ne contient aucun triangle de 0 de taille > 3 ce qui limite la taille des ensembles discriminants.

Question 12. *Donnez un minorant, fonction de n, sur le nombre de scripts de tout protocole de communication calculant* aleq_n*. Justifiez. En déduire la valeur exacte de* $\text{CC}(\text{aleq}_n)$.

SOLUTION. **[3 pts** $|\sum 36.0]$ D'après une proposition du cours, le nombre de scripts doit vérifier $\sigma(\text{aleq}_n) \geqslant |F_0| + |F_1| = 2^n - 1 + 4 > 2^n$. Donc $\text{CC}(\text{aleq}_n) \geqslant \log_2 \sigma(\text{aleq}_n) > n$. Donc $\text{CC}(\text{aleq}_n) \geqslant n + 1$, puisque CC() est un entier. À cause du majorant dû au protocole trivial (question 6), on doit avoir l'égalité, soit $\text{CC}(\text{aleq}_n) = n + 1$.

On considère la fonction d'égalité $eq_n(x, y) \in \{0, 1\}$ qui vaut 1 si et seulement si les mots binaires de n bits x et y sont égaux. On pourra admettre que $\text{CC}(eq_n) = n + 1$.

Question 13. *Donnez les détails d'un protocole de communication sans mot vide de coût* $n+2$ *calculant* eq_n *en* 4 *rondes de sorte qu'Alice transmet la moitié des bits de* x *à Bob et que Bob transmet la moitié des bits de* y *à Alice. Vous pourrez supposer* n *pair.*

SOLUTION. **[3 pts** $|\sum 39.0]$

À la ronde 1, Alice transmet les $n/2$ premiers bits de x, notés $x[0..n/2[$.

À la ronde 2, Bob transmet les $n/2$ derniers bits de y, $y[n/2..n[$.

À la ronde 3, Alice calcule $b = eq_{n/2}(x[n/2..n[, y[n/2..n[) \in \{0, 1\}$ quelle transmet à Bob.

À la ronde 4, Bob calcule $eq_{n/2}(x[0..n/2[, y[0..n/2[) \wedge b \in \{0, 1\}$ qu'il transmet à Alice.

En remplaçant b par son expression, on vérifie facilement que le dernier bit est effectivement $eq_n(x, y)$ puisque l'évaluation de cette fonction peut être découpée en segments (ici deux de longueur $n/2$) en appliquant and ET-logique entre ces segments. Il y a 4 rondes dans ce protocole, et son coût est $n/2+n/2+1+1 = n + 2$ comme souhaité.

On considère un processus de transmission particulier, noté $T_{1,2}$, qui consiste à envoyer des petits messages de longueur variable. Plus précisément, le participant (disons Alice) qui souhaite envoyer un mot binaire $S = s_0 s_1 s_2 \dots s_{t-1}$ de t bits à l'autre (Bob) le découpe en messages d'1 ou 2 bits. Lorsque c'est à son tour,

et que les bits $s_0 \dots s_{i-1}$ ont déjà été transmis, Alice envoie dans son message le bit $m = s_{i+1}$ si $s_i = 0$, ou envoie les deux bits $m = s_{i+1}s_{i+2}$ si $s_i = 1$. Notez que dans tous les cas, s_i n'est pas directement transmis. Cependant Bob, qui reçoit un tel petit message m, peut en déduire les bits $s_i s_{i+1}$ ou $s_i s_{i+1} s_{i+2}$ en fonction de la taille $|m| \in \{1, 2\}$. Si $i \in \{t-2, t-1\}$, les bits s_{i+1} et s_{i+2} peuvent ne pas exister. Dans ce cas Alice ajoute 1 ou 2 bits arbitaires, ce qui ne pose pas de problème à Bob qui ne tiendra pas compte de ces bits s'il connaît t.

Question 14. *En vous inspirant du protocole de la question 13, combinée avec le processus de transmission $T_{1,2}$, donnez un protocole de communication sans mot vide calculant eq_n. Donnez son coût. [Aide : Supposez d'abord qu'Alice et Bob terminent en même temps la transmission de leur partie. Puis adaptez le protocole pour qu'il fonctionne dans tous les cas.]*

SOLUTION. **[3 pts |∑42.0]** Il faut entrelacer les rondes 1 et 2 d'Alice et Bob du protocole de la question 13 en utilisant la transmission $T_{1,2}$. Alice, à partir de l'indice $i = 0$, envoie x_{i+1} ou $x_{i+1}x_{i+2}$ et Bob en déduit les 2 ou 3 bits de x à partir de x_i. Bob fait de même pour les 2 ou 3 bits de y à partir de $i = n/2$. Ils alternent ainsi jusqu'à ce qu'Alice et Bob aient envoyé toute leur moitié, respectivement $x[0..n/2[$ et $y[n/2..n[$.

Le coût de ce protocole économise au moins la transmission d'1 bit sur 3. Donc Alice transmet environ $2/3 \times (n/2)$ et pareil pour Bob, soit $2n/3$ au total, auquel il faut ajouter 2 bits pour les deux dernières rondes.

Il y a cependant un problème dans ce protocole si l'un des participants termine la transmission de sa moitié avant l'autre. Par exemple, Alice pourrait transmettre à chaque fois 2 bits (si $x[0..n/2[= 1**1**1**\cdots$), alors que Bob n'en transmet qu'1 seul (si $y[n/2..n[= 0*0*0*\cdots$). Or il est crucial d'alterner les transmissions. Dans cet exemple, Alice aura fini après $(n/2)/3 = n/6$ rondes, alors que Bob, après $n/6$ rondes, n'aura transmis (avec le bit implicite supprimé) que $n/6 \times 2 = n/3$ bits. Bob doit encore transmettre $n/2 - n/3 = n/6$ bits à Alice. Notons que $n/6$ est le pire déficite entre Alice et Bob.

Il y a plusieurs façons de corriger le problème.

- Si l'un des participants termine avant l'autre, alors il transmet d'un bloc le reste à l'autre. Cela fait donc au pire $2n/3 + n/6 = 5n/6$ bits pour les deux premières rondes, auquel il faut ajouter les deux dernières d'1 bit, soit $5n/6 + O(1)$.
- Alice transmet normalement ses bits selon $T_{1,2}$ dans l'ordre en partant des premiers bits, alors que Bob transmet ses bits selon $T_{1,2}$ en partant des derniers. Chaque participant, à toute ronde, peut évaluer le nombre de bits de son entrée qu'il a transmis et qu'il a reçu (en incluant le bit déduit de la taille). L'alternance des transmissions $T_{1,2}$ s'arrête dès qu'un participant s'aperçoit qu'un total de n bits a été transmis. Ainsi la partie d'Alice transmise à Bob et la partie de Bob transmise à Alice ont une longueur potentiellement différente mais qui se somment à n. Le calcul de eq_n, à partir de ses deux segments (éventuellement de longueur inégaux), ne pose pas de problème. Le coût de ce protocole s'analyse comme suit. À chaque application de $T_{1,2}$ par Alice ou Bob, 1 bit n'est pas transmis. Donc le coût vaut $n - r$ où r est le nombre minimum de transmissions $T_{1,2}$ (ou de rondes) effectuées. Comme chaque transmission $T_{1,2}$ consomme au plus 3 bits parmi ces n, il suit que $r \geqslant n/3$, et donc le coût est au plus $n - n/3 = 2n/3$, auquel on peut ajouter 2 bits pour la fin du calcul.

Question 15. *En remarquant que le procotole de communication pour eq_n issu de la question 14 a un coût $\leqslant n$, expliquez la contradiction apparente avec le fait que* $\mathrm{CC}(eq_n) = n + 1$.

SOLUTION. **[3 pts |∑45.0]** Le protocole issu de la question 14 n'est pas sans-préfixe. En effet, le premier message d'Alice peut être n'importe lequel parmi $\{0, 1, 00, 01, 10, 11\}$ puisque selon $T_{1,2}$, elle peut envoyer n'importe suite s_{i+1} ou $s_{i+1}s_{i+2}$ d'1 ou 2 bits. Or 0 est préfixe de 00 par exemple.

La définition de $\mathrm{CC}(f)$ est le coût minimum d'un procotole sans-préfixe (cf. question 10). Un protocole qui utilise des messages qui ne sont pas sans-préfixe peut avoir un coût inférieur, comme celui de la question 14. Il n'y a donc pas de contradiction.

FIN.