



Complexité de communication

Question 1. Dans le modèle b -CONGEST, expliquez ce qu'est, pour un graphe H fixé, le problème de la « détection d'un H » ? [NB : On ne demande pas sa solution.]

SOLUTION. 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 $\text{DETECT}(u)$ qui est vrai si et seulement il existe une copie (au moins) de H dans G . L'algorithme doit fonctionner dans un mode synchrone où chaque message contient au plus b bits (b -CONGEST).

Question 2. 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. 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, et le dernier bit doit être la réponse. Bien sûr, chacun ne connaît que son entrée et le protocole. L'objectif est de minimiser le nombre total de bits échangés.

Question 3. À quel problème d'algorithmique distribué vu en cours la complexité de communication nous a-t-elle servi ? Citez le résultat.

SOLUTION. On s'est servi de la complexité de communication pour démontrer que dans le modèle b -CONGEST, il fallait $\Omega(\sqrt{n}/b)$ rondes dans le pire cas pour détecter un C_4 dans un graphe à n sommets.

On considère la fonction booléenne $\text{gt}_n(x, y) \in \{0, 1\}$ qui étant donnés deux mots binaires x, y , chacun de n bits, renvoie 1 si et seulement si « $x > y$ » (*greater-than*), c'est-à-dire si l'entier codé en binaire par x est strictement supérieur à l'entier codé par y . Par exemple, $\text{gt}_2(11, 01) = 1$ car « $11 > 01$ ». En effet, $11_{\text{deux}} = 3_{\text{dix}} > 1_{\text{dix}} = 01_{\text{deux}}$.

Question 4. Donnez la matrice de la fonction gt_n , pour $n = 2$.

SOLUTION.

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

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 . Cela revient donc à dire qu'on ne peut pas avoir $f(x, y') = f(x', y) = v$.

Question 5. *Donnez un ensemble discriminant F_0 pour gt_2 le plus grand possible. Même question pour un ensemble discriminant F_1 .*

SOLUTION. Soient $F_0 = \{(00, 00), (01, 01), (10, 10), (11, 11)\}$ et $F_1 = \{(01, 00), (10, 01), (11, 10)\}$. On a $|F_0| = 4$ et $|F_1| = 3$.

Question 6. *Démontrez que tout ensemble discriminant F_v , pour une fonction f et valeur v données, ne peut avoir deux entrées qui sont situées sur une même ligne ou même colonne de la matrice de f .*

SOLUTION. Supposons que $(x, y), (x', y') \in F_v$ avec $x' = x$ (même ligne). Alors par définition de F_v , $f(x, y) = f(x', y') = v$ et $\bar{v} \in \{f(x, y'), f(x', y)\} = \{f(x', y'), f(x, y)\} = \{v\}$: contradiction car $v \neq \bar{v}$. On obtient la même contradiction si $y' = y$ (même colonne).

Question 7. *Généralisez la question 5 à tout n , c'est-à-dire construisez des ensembles discriminants F_0 et F_1 les plus grands possibles pour gt_n . Vous devez justifier les constructions ainsi que leur optimalité.*

SOLUTION. De manière générale, $F_0 = \{(x, x) : x \in \{0, 1\}^n\}$ et $F_1 = \{(x + 1, x) : x \in [0, 2^n - 1]\}$. On a que F_0 est bien discriminant car F_0 correspond à une diagonale (de valeurs à 0) et les éléments inférieurs à cette diagonale sont tous égaux à 1. Les éléments de F_1 correspondent à une sous-diagonale (de valeurs à 1) où tous les éléments supérieurs sont égaux à 0.

Concernant la taille, on a $|F_0| = 2^n$ et $|F_1| = 2^n - 1$. L'optimalité de la taille se déduit de la question 6 pour $|F_0|$ puisque si $|F_v| > 2^n$, on aurait deux entrées sur la même ligne ou même colonne. Donc $|F_0|, |F_1| \leq 2^n$. De plus, $|F_1| = 2^n$ est impossible car la première ligne est rempli de valeur nulle, interdisant d'avoir 2^n valeurs à 0. Donc $|F_1| < 2^n$.

Question 8. *Donnez un minorant, fonction de n , sur le nombre de scripts minimum de tout protocole de communication calculant gt_n . Justifiez. En déduire la complexité de communication de gt_n .*

SOLUTION. C'est au moins $|F_0| + |F_1| = 2^n + 2^n - 1$ à cause de la propriété des rectangles. On remarque que $|F_0| + |F_1| > 2^n$ car pour $n \geq 1$, $2^n - 1 \geq 1$. La complexité de communication est au moins le logarithme en base deux du nombre minimum de scripts, ce qui fait $\geq \log_2(|F_0| + |F_1|) > \log_2(2^n) = n$, soit $n + 1$ bits. Ce n'était pas demandé, mais on note que ce minorant est optimal puisque la complexité de communication de toute fonction booléenne (sur des entrées de n bits) est au plus $n + 1$.

Bellman-Ford

Dans le cours, nous avons vu l'algorithme de Bellman-Ford distribué (BFD) permettant de calculer, dans le cas asynchrone, un arbre couvrant de plus courts chemins dans un graphe G depuis un sommet racine r_0 fixé. Le principe, pour un sommet u , est d'envoyer à tous ses voisins la longueur du chemin courant menant à r_0 (longueur représentée par une variable $\text{LAYER}(u)$), dès que cette longueur diminue, tout en mettant à jour son père.

On rappelle qu'un graphe (G, ω) est dit *valué* (on dit parfois *pondéré*) si chaque arête uv possède un certain poids $\omega(uv) \in \mathbb{R}^+$. Bien évidemment, un arbre couvrant de plus courts chemins dans un graphe valué doit tenir compte du poids de ses arêtes. Un graphe est dit *non valué* si le poids de chaque arête est unitaire (donc compte pour 1).

Question 9. *Donnez une description de BFD, c'est-à-dire le pseudo-code comme vu en cours pour un sommet u quelconque, lorsque le graphe G est valué. Vous supposerez que la racine r_0 est connue de u ainsi que $\omega(uv)$ pour chaque voisin v de u .*

SOLUTION.

Répéter :

- $d := \text{RECEIVE}()$, et soit v l'émetteur
(si $u = r_0$, poser $v := \perp$ et $d := \omega(u\perp) := 0$)
- Si $d + \omega(uv) < \text{LAYER}(u)$:
 1. $\text{LAYER}(u) := d + \omega(uv)$
 2. $\text{PARENT}(u) := v$
 3. $\text{SEND}(\text{LAYER}(u), w)$ pour tout $w \in N(u) \setminus \{v\}$

Question 10. *Modifiez votre pseudo-code de façon à ce que chaque sommet u possède, en plus de la variable $\text{PERE}(u)$, une variable $\text{FILS}(u)$ contenant tous ses fils dans l'arbre calculé par BFD.*

SOLUTION. À FINIR ...

Question 11. *Décrivez un graphe non valué à n sommets et un scénario catastrophe où BFD produit $\Omega(n^3)$ messages.*

SOLUTION. Il faut considérer une clique à n sommets et le scénario où un chemin le plus long possible, avec $n - 1$ arêtes, est d'abord construit. Une fois ce chemin établi, dans un mauvais scénario, il est alors possible que chaque sommet de ce chemin envoie un message de mise à jour à tous ces voisins (sauf son parent) et que le chemin diminue d'une arête à la fois. Chaque diminution de ce chemin engendre donc $L \cdot (\text{deg}(u) - 1)$ messages, où L est la longueur courante du plus long chemin. Comme $L = n - 1, n - 2, \dots, 1$, et $\text{deg}(u) = n - 1$ pour tout sommet u , le nombre de messages envoyés sera d'au moins :

$$\sum_{L=n-1}^1 L \cdot (n-2) \geq \sum_{L=n-1}^{n/2} n/2 \cdot (n-2) = \Omega(n^3).$$

Ce n'était pas demandé, mais c'est la pire complexité envisageable, car dans le cas non valué, on a vu que c'était au plus $O(nm) = O(n^3)$ puisque $m = O(n^2)$.

Pour un graphe G sur lequel s'exécute BFD, on notera D le diamètre du graphe, n le nombre de sommets et m le nombre d'arêtes. Pour tout sommet u , on note $\mu(u)$ le nombre de fois où le sommet u met à jour sa variable locale $\text{LAYER}(u)$ durant l'exécution de BFD dans le pire des scénarios. On a vu dans le cours que le nombre de messages produits par l'algorithme était plus petit que $\sum_{u \in V(G)} \mu(u) \cdot \text{deg}(u) \leq 2m\mu$, où $\mu = \max_{u \in V(G)} \mu(u)$. Nous avons aussi vu, dans le cas non valué, que $\mu(u) < n$ pour tout sommet u de G .

Question 12. *Tout en justifiant, donnez un majorant sur le nombre de messages de BFD sur un graphe non valué dans le cas d'un scénario synchrone.*

SOLUTION. Il vaut $2m$ au plus car $\mu(u) = 1$ pour chaque sommet (sauf la racine où $\mu(r_0) = 0$). En effet, dans le cas synchrone, BFD calcule un BFS, c'est-à-dire un plus court chemin dans le cas non valué. Et une fois établi, le chemin ne peut plus être raccourci. La variable de chaque sommet est donc mise à jour une seule fois.

Pour tout réel $\alpha \in [0, 1]$, on dira qu'un scénario est α -synchrone si chaque message prend un temps au moins α pour traverser une arête. Dit autrement, le temps de traversée de chaque message prend un temps imprévisible appartenant à $[\alpha, 1]$. Le cas 1-synchrone correspond au cas synchrone et 0-synchrone au cas asynchrone.

Question 13. *Donnez un majorant du nombre de messages de BFD lors d'un scénario α -synchrone dans le cas non valué, en fonction des paramètres n, m, D et bien sûr α que vous pourrez supposer non nul¹ [Aide : Bornez la longueur du chemin construit de u à r_0 en fonction de la distance de u à r_0 .]*

SOLUTION. Soit u un sommet à distance $d(u)$ de r_0 . Le plus long chemin possible construit par BFD est $L(u) = \max \{d(u)/\alpha, n - 1\}$. Le cas $L(u) = n - 1$ est lorsque $\alpha = 0$. Notons que $L(u) \leq d(u)/\alpha + n - 1$. Il suit que $\mu(u) \leq \max L(u) - d(u), 1 \leq L(u) - d(u) + 1 = \max \{d(u)/\alpha, n - 1\} - d(u) + 1$. Comme $d(u) \leq D$, $\mu(u) \leq d(u) \cdot (1/\alpha - 1) + n \leq D(1/\alpha - 1) + n$. Il suit que le nombre de messages est au plus $O(mD(1/\alpha - 1) + nm)$.

Question 14. *En supposant que les poids des arêtes du graphe G sont des entiers de $\{1, \dots, W\}$, donnez un majorant sur le nombre de messages générés par BFD pour G dans ce cas.*

SOLUTION. Le plus long chemin à pour coût $(n - 1)W$. Comme ils diminuent d'une unité au moins à chaque mise à jour de la variable $LAYER(u)$, il suit que $\mu(u) \leq \mu \leq (n - 1)W$ et donc que le nombre de message est en $O(mnW)$ d'après la formule de la somme donnée en introduction.

Question 15. *Même question si, de plus, le scénario est α -synchrone.*

SOLUTION. Pour le cas du scénario α -synchrone, il faut remplacer le terme n par $D(1/\alpha - 1) + n$, soit au total $O(mnW + mD(1/\alpha - 1)W)$ messages.

Étant donné un graphe G et un sommet r_0 , soit $\Gamma(G, r_0)$ l'ensemble de tous les chemins simples² de G issus de r_0 . Si le graphe est valué par ω , on notera $\omega(P)$ le coût d'un chemin P , la somme des poids de ses arêtes, en posant $\omega(P) = 0$ si $P = \{r_0\}$ n'a pas d'arête. Notez que les chemins de $\Gamma(G, r_0)$ ne dépendent pas de ω et possèdent entre 0 et $n - 1$ arêtes. Il est facile de voir, par exemple, que si G est un chemin et r_0 une de ses extrémités, alors $|\Gamma(G, r_0)| = |V(G)| = n$.

Question 16. *Combien vaut $|\Gamma(G, r_0)|$ au maximum si G est un arbre de racine r_0 , de hauteur h et avec k feuilles ? Combien au minimum ? Donnez un exemple qui atteint ce minimum (pour chaque h et k donnés).*

SOLUTION. C'est au plus $1 + (h - 1)k$, le chemin sans arête étant commun à toutes les branches. C'est au moins $h - 1 + k$, atteint pour un chemin de hauteur $h - 1$ avec k feuilles au bout.

L'objectif est de généraliser la question 14 en tenant compte plus finement des coûts des chemins possibles. Soit $\mathcal{W}(G, \omega, r_0) = \{\omega(P) : P \in \Gamma(G, r_0)\}$, c'est-à-dire l'ensemble des coûts possibles pour les chemins de $\Gamma(G, r_0)$. Notez que $|\mathcal{W}(G, \omega, r_0)| \leq |\Gamma(G, r_0)|$, puisqu'au pire les chemins ont des coûts tous différents. Il y égalité dans l'exemple précédent, où G est un chemin et r_0 une de ses extrémités. On utilisera la notation allégée $|\mathcal{W}|$ et $|\Gamma|$ lorsque les paramètres G, ω, r_0 sont sous-entendus.

Question 17. *Calculez $|\Gamma|$ et $|\mathcal{W}|$ dans le cas d'un cycle G à cinq sommets où les poids des arêtes sont uniformes.*

SOLUTION. D'après le cas général ci-après, on a $|\Gamma| = 1 + 2(n - 1) = 9$ et $|\mathcal{W}| = n = 5$.

1. En effet, on pourra supposer que α peut approcher 0 arbitrairement près sans jamais l'atteindre.
2. C'est-à-dire des chemins ne passant pas deux fois par le même sommets.

Question 18. Généralisez à un cycle G à n sommets avec des poids uniformes. En déduire que $|\Gamma|/|\mathcal{W}| \rightarrow 2$ lorsque $n \rightarrow +\infty$.

SOLUTION. On a $|\Gamma| = 1 + 2(n - 1)$ (chemin vers la droite ou vers la gauche) et $|\mathcal{W}| = n$ (les chemins ayant entre 0 et $n - 1$ arêtes, cela fait exactement n coûts possibles différents).

Question 19. Décrivez un exemple de graphe G à n sommets avec un sommet r_0 et des poids uniformes où $|\mathcal{W}| = O(1)$ alors que $|\Gamma| = \Omega(n^2)$. [Aide : Pensez à un graphe de petit diamètre.]

SOLUTION. On peut prendre un $K_{2,n}$ (un graphe biparti complet, composé d'un stable avec n sommets et deux sommets de degré n tout d'eux connectés à ce stable) et r_0 un des deux sommets de degré n . Les poids sont unitaires. Le nombre de chemins est n pour ceux d'une arête et n pour ceux de deux arêtes, et $n(n - 1)$ pour ceux de trois arêtes. Donc $|\Gamma| = \Omega(n^2)$. Les coûts possibles peuvent être seulement parmi $\{0, 1, 2, 3\}$. Et donc $|\mathcal{W}| = 4 = O(1)$.

Question 20. Majorez le nombre de messages pour BFD appliqué à un graphe valué général (G, ω) depuis r_0 en fonction des paramètres précédents, notamment $|\Gamma|$ et/ou $|\mathcal{W}|$. Justifiez.

SOLUTION. Il est clair que $\mu < |\mathcal{W}|$ à cause de l'amélioration stricte des chemins dans l'algorithme BFD. Il suit que le nombre de messages est en $O(m|\mathcal{W}|)$.