

# ALGORITHMES DISTRIBUÉS

## MASTER2 INFORMATIQUE

Décembre 2013

---

*Ce devoir est à rendre par courriel à gavoille@labri.fr au plus tard **lundi 6 janvier 2013** minuit. Il doit se présenter sous la forme d'un fichier de quelques pages au format .pdf ou .ps comprenant vos nom, prénom et spécialité de Master. Répondez de manière concise tout en justifiant vos réponses. La notation tiendra compte, en premier lieu, de l'argumentation de vos réponses.*

---

Tous les algorithmes distribués que l'on considèrera dans le devoir seront dans le modèle LOCAL. On rappelle que chaque sommet du graphe procède à des calculs locaux et des rondes synchrones de messages échangés avec ses voisins dont la taille importe peu. Le sommet  $u$  possède un identifiant  $ID(u)$  codé par un entier unique dans le graphe qui peut éventuellement être utilisé par l'algorithme.

On suppose que le graphe  $G$  est un cycle orienté muni d'une 6-coloration. Tout sommet  $u$  possède une variable  $c(u) \in \{0, \dots, 5\}$  représentant la couleur du sommet  $u$ . L'orientation est définie en  $u$  par les relations prédécesseurs et successeurs, notées respectivement  $PRED(u)$  et  $SUCC(u)$ .

**Question 1.** Montrer qu'en une ronde il est possible de calculer pour  $G$  une 4-coloration en donnant l'algorithme correspondant. En déduire qu'il est possible de calculer une 3-coloration pour  $G$  en  $\lceil \frac{1}{2} \log^* m \rceil + 2$  rondes, une amélioration par rapport à l'algorithme vu en cours.

L'objectif maintenant est de concevoir un algorithme de coloration le plus efficace possible (toujours en terme de nombre de rondes) basé sur la construction d'un ensemble indépendant maximal. On supposera que cet ensemble est déjà calculé.

On rappelle qu'un ensemble de sommets  $M$  d'un graphe  $G$  est dit *indépendant* s'il n'y a pas d'arête de  $G$  entre deux sommets quelconques de  $M$ . Un ensemble indépendant  $M$  est *maximal* si dans  $G$  il n'y a pas d'autre ensemble indépendant dans  $G$  contenant strictement  $M$ . Notez bien que dans une coloration, l'ensemble des sommets d'une couleur donnée forme un ensemble indépendant (pas forcément maximal).

On suppose donc que  $G$  possède un ensemble indépendant maximal  $M$  défini par une variable booléenne  $M(u) \in \{0, 1\}$  qui vaut 1 si et seulement si le sommet  $u \in M$ .

**Question 2.** En supposant que  $G$  est un cycle orienté muni d'un ensemble indépendant maximal  $M$ , donner un algorithme qui construit une 3-coloration pour  $G$ . Donner le nombre de rondes effectuées par votre algorithme.

**T.S.V.P.**

On suppose maintenant que  $G$  est muni d'une  $m$ -coloration et a un degré maximum  $\Delta$  pour lequel on souhaite construire une  $(\Delta + 1)$ -coloration. Chaque sommet  $u$  possède une variable  $c(u) \in \{0, \dots, m - 1\}$  représentant la couleur de  $u$ , et on supposera que  $2 \leq \Delta < m$ .

**Question 3.** Montrer comment en une ronde il est possible d'orienter les arêtes de  $G$  afin de produire une  $\Delta$ -orientation pour  $G$ , c'est-à-dire que chaque sommet  $u$  possède au plus  $\Delta$  variables  $\text{PÈRE}_1(u), \dots, \text{PÈRE}_\Delta(u)$  représentant les arcs sortant de  $u$  dans cette orientation.

Notez que chaque arête doit être orienté dans un seul sens. En particulier, on ne peut avoir simultanément  $\text{PÈRE}_i(u) = v$  et  $\text{PÈRE}_j(v) = u$ . Bien sûr, il est possible que  $\text{PÈRE}_k(u) = \perp$  pour certains indices  $k$ .

Dans la suite on note  $F_k$  le graphe 1-orienté induit par les arcs  $(u, \text{PÈRE}_k(u))$ . On pose  $G[i, j] = \bigcup_{k=i}^j F_k$  le sous-graphe de  $G$  composé de l'union  $F_i \cup F_{i+1} \cup \dots \cup F_j$ . Notez que  $G[i, j]$  est une  $(j - i + 1)$ -orientation, et bien sûr que  $G[1, \Delta] = G$ .

Pour construire la  $(\Delta + 1)$ -coloration, on procède comme suit. D'abord on construit en parallèle sur chaque 1-orientation  $F_k$  une 3-coloration comme l'algorithme vu en cours. En particulier on obtient une  $(\Delta + 1)$ -coloration pour  $G[1, 1] = F_1$ . Puis, successivement pour chaque  $k = 2, \dots, \Delta$ , on fusionne la  $(\Delta + 1)$ -coloration de  $G[1, k - 1]$  avec la 3-coloration de  $F_k$  afin d'obtenir une  $3(\Delta + 1)$ -coloration pour  $G[1, k]$ . On effectue ensuite une réduction de palette pour finalement obtenir une  $(\Delta + 1)$ -coloration pour  $G[1, k]$  et on recommence avec  $k + 1$ .

**Question 4.** Détailler l'algorithme résumé précédemment, montrer qu'il est correct et calculer sa complexité en nombre de rondes.

On propose une variante de la construction précédente. Pour cela on décide, après avoir calculé la 3-coloration en parallèle sur chaque  $F_k$  de : (a) construire récursivement une  $(\Delta + 1)$ -coloration pour les sous-graphes  $G[1, k]$  et  $G[k + 1, \Delta]$  où  $k = \lfloor \Delta/2 \rfloor$ , (b) de fusionner les deux colorations, et enfin (c) réduire le nombre de couleurs à  $\Delta + 1$ .

**Question 5.** Détailler ce nouvel algorithme, puis discuter de sa validité, de sa complexité et de son intérêt.

**FIN.**