
*This **individual** homework must be uploaded on the [moodle](#) page of the lecture no later than **December 13th, 2021**. The deposit must be composed of a unique file, either in the plain text format or in pdf format (no .zip or .docx files). Important: all questions must be fully argued, and this will take into account for the scoring of your homework. The language for your answers can be either French or English. Please avoid to mix both languages.*

Layer Coloring

We shall consider only deterministic and distributed algorithms in the LOCAL model. We use the same notations as in the lecture notes. In particular, for a graph G , let $\Delta = \max \{\deg(u) : u \in V(G)\}$ denote the maximum degree of the graph G .

Question 1. *For the $(\Delta + 1)$ -coloring problem, what is the motivation for the choice of this number of colors? That is, why do we choose $\Delta + 1$ colors rather than another number of colors, say $\lceil \Delta/2 \rceil$ or n colors for instance?*

SOLUTION. On peut toujours colorier en $\Delta + 1$ couleurs, mais pas forcément en moins comme le montre une clique avec $\Delta + 1$ sommets qui nécessite $\Delta + 1 > \lceil \Delta/2 \rceil$ dès que $\Delta \geq 0$. Et puis avoir plus de couleurs est inutile car moins on en a mieux seront efficaces les applications utilisant se nombre de couleurs.

Let $f(\Delta, n)$ be the time complexity of some fast distributed algorithm providing, for every graph G of n vertices, an $(\Delta + 1)$ -coloring.

Question 2. *Without any proof, give an upper bound on $f(\Delta, n)$ that is described in the lecture notes.*

SOLUTION. $f(\Delta, n) = \log^* n + 2^{O(\Delta)}$.

We will introduce now a new coloring method that applies to graphs having some nice decomposition. More precisely, a (k, t) -decomposition of a graph G is a sequence $(G_1, G_2, \dots, G_t)$ of t induced subgraphs of G such that:

each vertex u of G

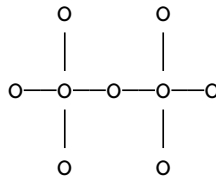
- appears in a unique subgraph G_i of the decomposition, and we will say that u is of *level* i .
- has at most k neighbors of level that is greater or equal to its own level.

For convenience, we denote by G_i^+ the subgraph of G induced by all the vertices of level $\geq i$. Of course, $G_1^+ = G$.

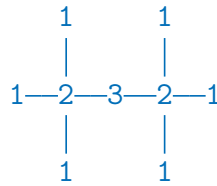
A (sequential) way to obtain a (k, t) -decomposition for G , if it exists, is to place in G_1 all the vertices of G_1^+ (i.e., G) with degree $\leq k$, and to remove them from G in order to obtain the subgraph G_2^+ . Then, we continue similarly by removing all the vertices of degree $\leq k$ from G_2^+ , and move them into G_2 , in order to obtain G_3^+ . And so on, t times, until to get the complete sequence $(G_1, \dots, G_t)$. In practice one can stop whenever one get an empty subgraph G_i^+ because if G has a (k, t) -decomposition¹ $(G_1, \dots, G_{t-1}, \emptyset)$, then $(G_1, \dots, G_{t-1})$ is a $(k, t - 1)$ -decomposition of G as well.

Question 3. *Give a $(1, 3)$ -decomposition of the following tree, by simply reporting the level of each vertex.*

¹By denoting by $\emptyset$ the empty graph, so the graph with no vertices.



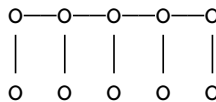
SOLUTION.



Question 4. What is the condition that must fulfill a graph G with n vertices in order to have a (k, n) -decomposition?

SOLUTION. Il y a des conditions nécessaires comme avoir au moins un sommet de degré au plus k ou avoir au plus kn arêtes. Mais il faut et il suffit que tout sous-graphe de G possède un sommet de degré au plus k . En effet, si c'est bien le cas alors G possède une (k, n) -décomposition. Sinon, il existe un sous-graphe H avec aucun sommet de degré $\leq k$. Et H ne pourra plus être re-décomposé.

Let consider the *comb* graph with n vertices, assuming n is even, be the graph obtained from a path of $n/2$ vertices each one having a pending vertex (a vertex of degree 1). Hereafter is a comb with 10 vertices.



Question 5. What is the minimum number t of levels in a $(1, t)$ -decomposition of a comb with 10 vertices?

SOLUTION. $t = 4$. Le somme central du chemin sans les feuilles, doit être de niveau 4.

Question 6. In general, what is the minimum number t of levels in a $(1, t)$ -decomposition of a comb with n vertices?

SOLUTION. On a $t = 1 + \lceil n/4 \rceil$.

Si $n/2 = 2i + 1$ est impair (comme pour $n = 10$), alors il n'y a qu'un seul sommet de niveau t , et ce niveau vaut $t = i + 2 = 1 + \lceil n/4 \rceil$.

Si $n/2 = 2i$ est pair (comme pour $n = 8$), alors il n'y a deux sommets de niveaux t , et ce niveau vaut $t = i + 1 = 1 + \lceil n/4 \rceil$.

Question 7. What is the minimum number t of levels in a $(2, t)$ -decomposition of a comb with n vertices?

SOLUTION. Il faut $t = 2$ niveaux pour $n > 4$. En effet, la suppression des sommets de degrés ≤ 2 , laisse un chemin non vide (donc composé que de sommets de degrés 1 ou 2). Lorsque $n \leq 4$, alors il faut $t = 1$ niveaux car le graphe est un chemin.

Question 8. Assume that G is a graph with n vertices and m edges, and that there exist two integers $k, b \geq 1$ such that $m \leq (k + 1) \cdot (1 - 1/b) \cdot n/2$. Show that G has at least n/b vertices of degree $\leq k$.

SOLUTION. Soit S l'ensemble des sommets de degré $\leq k$ dans G . Notons qu'il y a $n - |S|$ sommets de degré $\geq k + 1$ dans G . Ainsi $2m = \sum_u \deg(u) \geq (n - |S|)(k + 1)$ ce qui implique $m \geq (n - |S|)(k + 1)/2$. Soit $|S| \geq n/b$, et alors il n'y a rien d'autre à démontrer. Soit $|S| < n/b$. Mais alors $n - |S| > n - n/b =$

$(1 - 1/b)n$. Il suit que $m > (k + 1)(1 - 1/b)n/2$ en contradiction avec la définitions de k et b . Donc $|S| \geq n/b$.

In the remaining, you may assume that the property stated in Question 8 is indeed true.

Question 9. Show that every tree with n vertices has a $(3, \lceil \log_2 n \rceil - 1)$ -decomposition and also a $(2, \lceil \log_{3/2} n \rceil - 1)$ -decomposition, assuming n is large enough. [Hint: Think that the decomposition may stop whenever it remains at most $k + 1$ vertices. Why?]

SOLUTION. De manière générale, il y a $t = 1$ niveau si $b = 1$, et si $b > 1$, il y a environ $\log_{b/(b-1)} n$ niveaux car à chaque fois, en enlevant au moins n/b sommets, on se retrouve avec au plus $(1 - 1/b) \cdot n$ sommets. De manière générale, après i étapes il reste $n \cdot (1 - 1/b)^i$ sommets. Notons que la décomposition se termine juste après l'étape où il reste $\leq k + 1$ sommets, car à ce moment là tous les sommets sont de degré $\leq k$. Donc le nombre d'étapes est le plus petit t tel que (en posant $B = b/(b - 1)$)

$$\begin{aligned} n \cdot \left(\frac{b-1}{b} \right)^{t-1} &\leq k+1 \Leftrightarrow \frac{n}{k+1} \leq \left(\frac{b}{b-1} \right)^{t-1} = B^{t-1} \\ \Leftrightarrow \log_B \left(\frac{n}{k+1} \right) &\leq t-1 \Leftrightarrow t \geq \log_B \left(\frac{n}{k+1} \right) + 1 \end{aligned}$$

Le plus petit entier t recherché vaut $t = \lceil \log_B(n/(k+1)) \rceil + 1$.

Pour avoir une $(3, \dots)$ -décomposition, il faut prendre $k = 3$ et $b = 2$. On vérifie que $m \leq 4 \cdot (1 - 1/2) \cdot n/2 = n$ ce qui est bien vrai dans tout arbre (et aussi toute forêt). On a alors $B = 2$ et $t = \lceil \log_B(n) - 2 \rceil + 1 = \lceil \log_B(n) \rceil - 1$.

Pour avoir une $(2, \dots)$ -décomposition il faut prendre $k = 2$ et $b = 3$. On vérifie aussi que $m \leq 3 \cdot (1 - 1/3) \cdot n/2 = 3 \cdot 2/3 \cdot n/2 = n$ ce qui est encore vrai pour tout arbre. On a alors $B = 3/2$, et $t = \lceil \log_B(n) - \log_B(3) \rceil + 1 = \lceil \log_B(n) - 2.70\dots \rceil + 1 \leq \lceil \log_B(n) - 2 \rceil + 1 = \lceil \log_B(n) \rceil - 1$.

NB: Dans le cas des arbres, on peut en fait faire un peu mieux, et montrer qu'il existe toujours $> n/2$ sommets de degré ≤ 2 . Et donc en prenant $k = b = 2$, on obtient $t = \lceil \log_2(n) - \log_2(3) \rceil + 1 = \lceil \log_2(n) - 1.58\dots \rceil + 1 \leq \lceil \log_2(n) \rceil$, soit une $(2, \lceil \log_2 n \rceil)$ -décomposition.

En voici la preuve. Soit n_i le nombre de sommets de degré i dans un arbre. En comptant le nombre d'arête et de sommets, il vient:

$$2(n-1) = \sum_{i \geq 1} i \cdot n_i = n_1 + 2n_2 + 3n_3 + \dots \quad (1)$$

$$n = \sum_{i \geq 1} n_i = n_1 + n_2 + n_3 + \dots \quad (2)$$

En faisant (1)-(2), on obtient $n - 2 = n_2 + \sum_{i \geq 3} (i-1) \cdot n_i \geq n_2 + 2 \sum_{i \geq 3} n_i$. On conclut que $\sum_{i \geq 3} n_i \leq n/2 - 1 - n_2/2 < n/2$. De (2) on déduit que $n_1 + n_2 = n - \sum_{i \geq 3} n_i > n/2$.

We shall present now a method to construct a $(k+1)$ -coloring of a graph G having a (k, t) -decomposition $(G_1, \dots, G_t)$. We assume that k and t are parameters known by all the vertices, and that colors are integers taken from $[0, k]$. We will make the use of the function FIRSTFREE seen in the lecture and defined by $\text{FIRSTFREE}(X) = \min(\mathbb{N} \setminus X)$, for every subset $X \subseteq \mathbb{N}$.

The three main steps of the method are the following:

1. Compute the level of every vertex of G .
2. Compute (in parallel) a $(k+1)$ -coloring of every G_i .
3. For each $i = t-1, t-2, \dots, 1$ in this order, update the coloring of vertices only in G_i thanks to FIRSTFREE applied by colors and for the graph G_i^+ .

Question 10. Show that this method indeed provides a valid $(k+1)$ -coloring for G .

SOLUTION. Cette méthode donne bien une $(k + 1)$ -coloration en remarquant que chaque sommet u de niveau i qui se re-colorie (avec FIRSTFREE) est de degré $\leq k$ dans G_i^+ laissant une couleur libre pour u . De plus, on peut vérifier qu'il n'y a jamais deux voisins de G qui se re-colorient (avec FIRSTFREE) à la même ronde.

The goal now is to give a full description of a distributed algorithm for this method. We will assume that a vertex does not know its degree in G when it starts its computation. But note that it can learn it within a single round of communication with its neighbors.

Question 11. *Describe a distributed algorithm for Step 1 of the method that is in charge of computing the level of each vertex u of G , hereafter denoted by $\ell(u)$. Give an analysis of the number of rounds.*

SOLUTION. On considère le code suivant qui s'exécute en t rondes.

- $\ell(u) := 0$
- Pour $i := 1$ à t :
 - NEWROUND
 - si $\ell(u) > 0$, STOP
 - SENDALL(0, v)
 - #RECEIVE(v) $\leq k$, alors $\ell(u) := i$

Tant que le niveau d'un sommet n'est pas fixé, il participe aux échanges. Le numéro de ronde où il reçoit $\leq k$ messages détermine son niveau.

Question 12. *Describe a distributed algorithm for Step 2 of the method. Using fonction f defined above in Question 2, give an analysis of the number of rounds.*

SOLUTION. Il faut calculer une $(k + 1)$ -coloration dans G_i , qui a au plus n sommets. Il faut donc $f(k, n)$ rondes par définition de f .

Question 13. *Describe a distributed algorithm for Step 3 of the method. Give an analysis of the number of rounds.*

SOLUTION. Le FIRSTFREE dans G_i^+ prend une ronde. On l'applique pour les $k + 1$ couleurs de G_i , et pour $t - 1$ graphes $G_{t-1}, \dots, G_1$, soit $(t - 1) \cdot (k + 1)$ rondes pour l'étape 3.

Question 14. *Give the total number of rounds of your distributed algorithm for computing a $(k + 1)$ -coloring for G .*

SOLUTION. D'après les questions précédentes, il faut un total de $t + f(k, n) + (t - 1)(k + 1) < t \cdot (k + 2) + f(k, n)$ rondes.

Question 15. *Explain and justify how to compute in $O(\log n)$ rounds a 3-coloring of a tree with n vertices.*

SOLUTION. Il faut prendre une $(2, \lceil \log_{3/2}(n) \rceil - 1)$ -décomposition, soit $k = 2$ et $t = \lceil \log_{3/2}(n) \rceil - 1$. On obtient une $(k + 1)$ -coloration, soit une 3-coloration en temps $(t - 1) \cdot (k + 1) + f(k, n) = 4t + f(4, n) = O(\log n)$, car $f(4, n) = O(\log^* n + 2^{O(1)}) \ll O(\log n)$.

NB: Il s'agit d'un résultat connu de [MR89]: G.L. Miller and J.H. Reif. *Parallel Tree Contraction Part 1: Fundamentals*. Advances in Computing Research, 5:47–72, 1989.

Question 16. *Discuss whether this coloring is a progress or not with respect to the 3-coloring distributed algorithm for trees and cycles we have seen in the lecture and that runs in $O(\log^* n)$ rounds.*

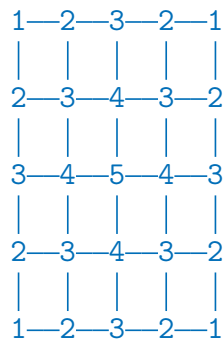
SOLUTION. L'algorithme en $O(\log^* n)$ ne s'applique que si G possède une 1-orientation ($u \mapsto \text{PARENT}(u)$), alors qu'ici l'algorithme fonctionne sans aucune orientation. Il est donc plus général,

bien que moins rapide.

Question 17. *Consider a square grid G with a total of n vertices. Give a $(3, t)$ -decomposition for G with the minimum possible t . What is this minimum t . Justify. Same question for a $(2, t)$ -decomposition.*

SOLUTION. Les sommets de degrés $k \leq 3$ sont situés sur le périmètre de la grille. Lorsqu'on enlève un périmètre, on enlève 2 lignes et 2 colonnes. Et donc après $t = \sqrt{n}/2 + O(1)$ étapes la grille est vide.

Les sommets de degré $k = 2$ sont situés sur les coins. On enlève donc successivement, en rognant chaque coin, 4×1 , 4×2 , $4 \times 3 \dots 4i$ sommets jusqu'à atteindre un diamant central à la grille et de côté $i = \sqrt{n}/2$, ce qui prend $i = \sqrt{n}/2$ étapes. Puis cela rediminue de $4i$ jusqu'à 4×3 , 4×2 , 4×1 et éventuellement 1 sommet, ce qui prend encore aussi $\sqrt{n}/2$ étapes. Au final c'est donc $t = \sqrt{n} + O(1)$ étapes environ. Voir l'exemple ci-dessous pour une grille 5×5 avec ces 5 niveaux.



Question 18. *Show that every planar graph with n vertices has a distributed algorithm for computing a 7-coloring in $O(\log n)$ rounds.*

SOLUTION. Prendre $k = 6$ et $b = 7$. On a alors $m \leq (6+1) \times (1-1/7) \times n/2 = 7 \times 6/7 \times n/2 = 3n$ ce qui est vrai pour les graphes planaires qui ont au plus $3n - 6$ arêtes. On a alors, avec $B = b/(b-1) = 7/6$, $t = \log_B n + O(1)$, ce qui fait $O(\log n)$ rondes.

NB: Il a été montré par en 2019 qu'il est possible de produire une 6-coloration [CM19], mais aussi une 5-coloration en $O(\log n)$ rondes [Postle19], alors qu'une 4-coloration nécessite $\Omega(n)$ rondes [CM19] (et $O(n)$ sont clairement suffisantes).

FIN.