

Auteur : Greg Depoire

1 Intro informelle

1.1 Contexte

1.1.1 Intervenants

- Laurent Feuilloley (LIRIS)
- Théo Pierron (LIRIS)
- Stéphan Tomassé (LIP) qui remplace Nicolas Bousquet

Poser plutôt les questions à Laurent Feuilloley et Théo Pierron.

1.1.2 MCC

- Examen sur table
- *Scribing notes* : un cours \rightarrow un scribe + un comité éditorial

1.1.3 Le modèle étudié

- Le graphe en entrée est le réseau. Les nœuds sont les unités de calcul et les arêtes sont les liens de communication.
- Chaque nœud commence avec la seule connaissance de lui-même et de ses liens. Il doit terminer avec une partie de la solution.

Exemple 1 (Coloration). Chaque nœud termine avec une couleur différente de ses voisins.

Exemple 2 (Arbre couvrant). Chaque nœud termine avec un parent (voisin ou lui-même si racine) tel que le total forme un arbre.

- *Breaking symmetry* : dans certains graphes comme les cycles, les nœuds sont équivalents. On donne donc à chaque nœud x un identifiant unique $i(x) \in [n^2] = \{1, \dots, n^2\}$ où n est le nombre de nœuds du graphe.

1.1.4 Rounds de communication

Initialement, chaque nœud connaît son identifiant et ses liens. Lors d'un round, les nœuds envoient et reçoivent un message de leurs voisins puis effectuent un calcul. À la fin du dernier round, chaque nœud termine avec sa solution locale.

1.1.5 Complexité

La complexité est le nombre de rounds effectués pour calculer une solution. Le temps de calcul est négligé devant les communications.

1.1.6 Localité

Après k rounds, un nœud connaît son voisinage à distance k . Pour que l'algorithme soit efficace, il faut donc prendre la décision en utilisant une petite partie locale de l'instance (« fast = local »).

1.2 Un mauvais cas

Supposons que l'on veuille effectuer une tâche simple de colorer un chemin de n nœuds.

Il n'existe pas d'algorithme distribué qui 2-colore P_{n+3} en moins de $n/2$ rounds.

Preuve 1. Supposons qu'un tel algorithme existe.

L'algorithme colorie x et y sur la connaissance de leur voisinage à distance $< n/2$.

Ainsi, en l'appliquant sur le cycle de longueur $n + 3$ on obtient une 2-coloration d'un cycle impair ce qui est impossible. L'algorithme n'existe donc pas.

1.3 3-colorer un chemin

On suppose maintenant que le réseau est le chemin *orienté* de longueur n . Chaque nœud peut toujours communiquer avec ses deux voisins, mais il y a une distinction entre le voisin de gauche et le voisin de droite.

1.3.1 Algo naïf

Complexité 1. $\Theta(n^2)$

1.3.2 Algo bruteforce

Il y a $n - 1$ rounds de communication. Chaque nœud se colorie avec la parité de la distance à la racine. On obtient une 2-coloration.

Complexité 2. $\Theta(n)$

1.3.3 Réduction des couleurs

Si un nœud x a une couleur plus grande que ses voisins, on le recolorie en 1, 2 ou 3 en fonction de ses voisins.

Il existe une méthode en $\log^*(n)$.

1.3.4 Cole-Vishkin

L'algorithme de Cole-Vishkin permet de réduire une C -coloration en une 6-coloration en $\log^* C$ rounds.

Note 1. $\log^* n$ = nombre de $\lfloor \log \rfloor$ à appliquer pour atteindre 1

C'est la fonction réciproque de $\text{Tour}(n) = \underbrace{2^{2^{\cdot^2}}}_{n \text{ fois}}$.

On note x le nœud qui exécute l'algorithme, y son nœud voisin et $c(x)$ la couleur du nœud x .

On obtient une 6-coloration.

Correction 1. Premièrement, l'invariant « $c(x) \neq c(y)$ » est vrai au début et est préservé à chaque round donc la coloration retournée est bien propre.

Deuxièmement, si toutes les couleurs sont inférieures à C au début d'un round, alors elles sont inférieures à $2^{\lceil \log C \rceil}$ à la fin du round. On peut prouver par induction qu'au bout de $\log^* n^2$ rounds, on obtient une 6-coloration.

Note 2. 6 est un point fixe de $2^{\lceil \log \rceil}$.

n	$2^{\lceil \log n \rceil}$
9	7
8	6
7	6
6	6

Complexité 3. $O(\log^* n)$

Note 3. Pour les chemins non orientés, on calcule $c(x)$ à partir des couples (p, b) correspondants aux deux voisins.

Pour 3-colorier le chemin, on peut à présent appliquer la réduction des couleurs 3 fois.

1.4 Nombre chromatique et line graph

Définition 1. Le nombre chromatique $\chi(G)$ d'un graphe G est le nombre minimal de couleurs nécessaire à la coloration de G .

On peut construire des graphes de nombre chromatique arbitrairement grand.

Exemple 3. La clique K_n vérifie $\chi(K_n) = n$.

Et si on interdit les triangles (3 sommets reliés les uns aux autres) ?

Exemple 4 (Propositions des élèves).

Exemple 5 (Mycielski).

1.4.1 Pour plus tard

Définition 2. La *girth* d'un graphe est la longueur d'un plus court cycle du graphe.

Note 4. Si un graphe a pour girth $2k + 2$ alors pour chaque sommet x , le graphe des sommets à distance k de x est un arbre.

Théorème 1 (Erdős). Il existe des graphes de grand nombre chromatique et de girth arbitraire.

1.4.2 Autre modèle : le *shift graph*

Définition 3. Shift_n est le graphe orienté dont les nœuds sont les couples ij avec $1 \leq i < j \leq n$ et les arêtes sont les couples $ij \rightarrow jk$.

Exemple 6 (Shift₄).

Il n'y a pas de triangle dans les shift graphs.

Colorer les nœuds de Shift_n équivaut à colorer les arêtes de K_n .

Colorer les nœuds de *shift_n* équivaut à colorer les arêtes de K_n .

Définition 4. Ramsey_k(3) est le n minimum tel que toute k -coloration des arêtes de K_n contient un triangle monochromatique.

Lemme 1.

$$\text{Ramsey}_2(3) = 6$$

Preuve 2. Il existe une 2-coloration des arêtes K_5 sans triangle monochromatique. Cependant, une coloration des arêtes de K_6 en 2 couleurs contient forcément un triangle monochromatique.

Soit un sommet x . Il existe une couleur majoritaire c parmi les arêtes de x . On considère les voisins correspondants aux arêtes de cette couleur c .

Si toutes les arêtes entre eux sont de l'autre couleur alors il y a un triangle monochromatique.

Sinon, il existe une arête de couleur c entre eux et il y a aussi un triangle monochromatique.

La notion de *line graph* généralise les shift graphs.

Définition 5 (Line graph). Soit G un graphe orienté. Le *line graph* de G est le graphe orienté $L(G)$ défini par :

- Les sommets sont les arêtes de G .
- Les arêtes sont les couples $i \rightarrow j$ où i, j et k sont des sommets de G .

Exemple 7.

$$\text{Shift}_n = L(\overrightarrow{K_n})$$

où $\overrightarrow{K_n}$ est le graphe à n sommets dans lequel $i \rightarrow j$ ssi. $i < j$.

On veut comparer $\chi(L(G))$ à $\chi(G)$.

Lemme 2.

$$\log \chi(G) \leq \chi(L(G))$$

Preuve 3. Considérons une k -coloration de $L(G)$. On transforme cette coloration en une coloration de G . Si x est un sommet de G alors on pose

$$c'(x) = \{c(xy) | x \rightarrow y\}$$

c est propre car si $x \rightarrow y$ alors $c(xy) \in c'(x)$ mais $c(xy) \notin c'(y)$. En effet, si il existait un z tel que $c(xy) = c(yz) \in c'(y)$, alors la coloration de $L(G)$ ne serait pas propre car $xy \rightarrow yz$ dans $L(G)$.

La coloration de G obtenue est une 2^k coloration donc $\chi(G) \leq 2^{\chi(L(G))}$.

Auteur : Julien Cocquet

Lemme 3.

$$\chi(L(G)) \leq \log \chi(G) + \log \log \chi(G)$$

Preuve 4. L'idée de la preuve est de définir une fonction injective qui à chaque couleur associe un ensemble de n entiers. Pour cela, il faut et il suffit que le nombre de couleurs soit suffisamment petit devant le nombre d'éléments disponibles pour fabriquer nos ensembles.

Considérons

$$\phi \left| \begin{array}{l} \mathbb{N} \longrightarrow \mathbb{N} \\ x \longmapsto \min\{2n, \binom{2n}{n} \geq x\} \end{array} \right. \quad (1)$$

$\phi(x)$ est le nombre minimum (pair) d'éléments qui réalise la condition énoncée au début de la preuve.

Supposons donc que G est k -colorable et trouvons une coloration de $L(G)$. Soit c tel que $\phi(k) = c$ (c est donc pair). Par la remarque précédente, on peut remplacer la couleur de chaque sommet de D par un sous ensemble de taille $\frac{c}{2}$ de $\{1, \dots, c\}$, tel que 2 sous-ensembles adjacents ont une couleur (ensembliste) différente.

On colorie les arêtes de G : si x et y sont coloriés par X et Y (ensembles de taille $\frac{c}{2}$), alors $X \neq Y$ car l'image par une fonction injective d'une coloration propre est une coloration propre. Pour colorier l'arête xy , il suffit de choisir $i \in Y \setminus X$ et de lui assigner cette couleur. La coloration du Linegraph ainsi définie est propre : si une arête de G est incidente à xy , alors elle aura une couleur dans X , et donc nécessairement différente de i .

Conclusion : $\chi(L(G)) \leq \phi(\chi(G))$.

Il reste à étudier ϕ pour conclure. Montrons (qualitativement) que $\phi(x) \leq \log x + \log \log x$:

Par la formule de Stirling, $\binom{2n}{n} \underset{+\infty}{\sim} \frac{2^{2n}}{\sqrt{2n}}$.

$$\text{Donc } \binom{\frac{\log x + \log \log x}{2}}{\frac{\log x + \log \log x}{2}} \sim \frac{2^{\frac{\log x + \log \log x}{2}}}{\sqrt{2 \frac{\log x + \log \log x}{2}}} \sim \frac{x \log x}{\sqrt{2 \log x}} \sim x \sqrt{\log x} \geq x.$$

On en déduit l'encadrement (fin) suivant :

Théorème 2. $\log \chi(G) \leq \chi(L(G)) \leq \phi(\chi(G))$

Interrogation : Pour quelle raison $\chi(G)$ peut-il être grand ?

Une piste naturelle pour le comprendre est de répondre à la question suivante : Si G et H ont nombre chromatique t , est-ce que l'intersection de G et de H a nombre chromatique t ?

1.4.3 De la nécessité de définir l'intersection : détour en théorie des catégories

Plaçons nous dans la catégorie des graphes, dans laquelle les objets sont des graphes et les morphismes sont des homomorphismes de graphe.

Définition 6 (Homomorphisme de graphe). Un homomorphisme d'un graphe G vers un graphe H est une fonction $f : V(G) \rightarrow V(H)$ respectant la condition :

$$xy \in E(G) \implies f(x)f(y) \in E(H)$$

Le coloriage a également un sens catégorique ; $\chi(G) = \min\{t, \exists f : G \rightarrow K_t \text{ homomorphisme}\}$. Une t -coloration définit un homomorphisme, et il existe un homomorphisme seulement s'il y a une t -coloration.

Définition 7 (Union et "intersection" de graphes). Soient G et H deux graphes.

- L'union de G et de H est le plus petit graphe K qui contient G et H , c'est-à-dire que pour tout graphe L tel que $G \rightarrow L$ et $H \rightarrow L$, alors $K \rightarrow L$.
- Le produit catégorique (ou "l'intersection") de G et de H est un graphe $K = G \times H$ tel que $K \rightarrow G$, $K \rightarrow H$ et pour tout graphe L tel que $L \rightarrow G$ et $L \rightarrow H$, alors $L \rightarrow K$.

La définition catégorique du produit catégorique permet de définir élémentairement la bonne notion d'intersection de graphes.

Définition 8 (Produit catégorique de 2 graphes). Soient $G = (V(G), E(G))$ et $H = (V(H), E(H))$ deux graphes. On définit $G \times H$ par $V(G \times H) = V(G) \times V(H)$, et $E(G \times H) = \{ ((u, v), (w, z), (u, w) \in E(G) \text{ et } (v, z) \in E(H) \}$.

1.4.4 Une piste à explorer

On considère pour cette partie des graphes orientés.

Soit $g : \mathbb{N} \rightarrow \mathbb{N}$ qui vérifie $g(t) = \min_{G, \text{Orientés}} \{ \chi(G \times H), \chi(G) = t \text{ et } \chi(H) = t \}$.

Théorème 3. Soit $\lim_{+\infty} (g) = +\infty$, soit g est bornée par 4.

Preuve 5 (Sketch de preuve). Imaginons que g soit bornée par c . Alors il existe G et H orientés de nombre chromatique arbitrairement grand tels que $\chi(G \times H) \leq c$.

Observation 1 : Il existe un entier k fixé tel que $\phi^k(c) = 4$.

Observation 2 : $L(G \times H) = L(G) \times L(H)$.

On obtient $\chi(L(G) \times L(H)) = \chi(L(G \times H)) \leq \phi(\chi(G \times H))$. Donc $\chi(L^k(G) \times L^k(H)) = \chi(L^k(G \times H)) \leq \phi^k(\chi(G \times H)) \leq \phi^k(c) = 4$.

Nous avons donc montré que $g(\phi^k(t))$ est bornée par 4, ... k étant indépendant de t , on obtient bien le résultat : g est bornée par 4.

Note 5. Le meilleur résultat connu à ce jour est une borne améliorée, qui passe de 4 à 3. Il existe un résultat similaire pour les graphes non orientés, pour lequel la borne est 9.

1.5 Les arguments de N. Linial

On peut adapter une idée précédente pour colorier un chemin de longueur n .

On se donne un ensemble de couleurs de départ $[1, n^2]$ (voir Breaking Symetry), et on suppose que les noeuds se sont mis d'accord en avance sur une fonction injective $[1, n^2] \rightarrow \{\text{Parties à } \frac{k}{2} \text{ éléments de } [1, k]\}$. Après un round de communication, on peut réduire l'ensemble de couleurs à $[1, \phi(n^2)]$. Pour se faire, chaque noeud interprète sa couleur avec la fonction injective, et choisit une couleur parmi les $\frac{k}{2}$ disponibles dans son ensemble, qui est différente des $\frac{k}{2}$ de son antécédent.

Correction 2. La preuve est la même que celle de la borne inférieure de $\chi(L(D))$. Le nombre de rounds est le nombre de fois où l'on doit appliquer ϕ avant de boucler, donc heuristiquement en \log^* . Il suffit d'appliquer la réduction de couleur pour passer de 4 à 3 couleurs.

Peut-on améliorer $\log^*(n)$?

Soit un protocole de 3-coloration du chemin orienté de longueur n en k étapes, cherchons une borne inférieure sur k . Au bout de k étapes, x doit décider d'une couleur en connaissant sur k successeurs/ k prédécesseurs. Autrement dit, x connaît un mot sur $\{1, \dots, n^2\}$ de longueur $2k + 1$.

Le protocole peut être vu comme $P : \{1, \dots, n^2\}^{2k+1} \rightarrow \{1, 2, 3\}$. Pour que le protocole soit correct, il doit vérifier la condition suivante : si 2 mots M et M' sont deux mots de taille $2k + 1$ tels que M' soit un shift de M (c'est-à-dire que la i ème lettre de M' est $i+1$ ème lettre de M), alors ils doivent être colorés différemment.

En particulier, on peut restreindre cette condition aux mots de P croissants (en termes d'identifiants).

Note 6. K_{n^2} orienté (voir page 5) est le graphe dont les sommets sont les mots de longueur 1, et dont les arêtes correspondent aux mots de longueur 2.
 Shift_{n^2} est donc le graphe dont les sommets sont les mots croissants de longueur 2, et dont les arêtes correspondent aux mots de longueur 3.
 $L^{(2k)}(K_{n^2})$ est donc le graphe dont les sommets correspondent aux mots de longueur $2k+1$.

Donc P fournit une 3-coloration de $L^{(2k)}(K_{n^2})$. Or, $\chi(L^{(2k)}(K_{n^2})) \geq \log^{(2k)}(K_{n^2}) = \log^{(2k)}(K_{n^2})$.

En particulier, si $2k < \frac{\log^*(n^2)-3}{2}$, on a $\chi(L^{(2k)}(K_{n^2})) \geq 3$, ce qui est contradictoire car le protocole donne une 3-coloration de $L^{(2k)}(K_{n^2})$.

Théorème 4. Le nombre de rounds est minoré par $\frac{\log^*(n^2)-3}{2}$.

1.6 Colorer des graphes

Comme pour les chemins (où l'on ne peut 2-colorier), on va tenter de colorer en Δ (i.e. le degré maximum des sommets) + 1 couleurs.

En réalité, on peut (presque) toujours colorier un graphe G en Δ couleurs, les seuls contre-exemples à cette règle étant les cycles de longueur impaire et les cliques.

Définition 9. On dit que $F \subset 2^{[m]}$ famille d'ensembles est k -cover-free si $\forall X \in F$, $\forall X_1, \dots, X_k$ distincts de X , X n'est pas inclus dans $X_1 \cup \dots \cup X_k$.

1.6.1 Construire des familles k -cover-free sur un ensemble de base petit

On se donne un nombre premier p , on considère des polynômes de degré d sur $\mathbb{F}_p[X]$. On observe que 2 polynômes distincts de degré d ont au plus $d+1$ valeurs sur lesquels ils coïncident.

Principe de construction : L'ensemble de base qu'on considère est $\mathbb{F}_p \times \mathbb{F}_p$, à un polynôme P de degré d on associe $X_p \subset \mathbb{F}_p^2$, où $X_p = \{(a, P(a)), a \in \mathbb{F}_p\}$. Cette famille est k -cover-free pour $k < \lfloor \frac{p}{d} \rfloor$ par le principe des tiroirs !

Théorème 5.

- Fait 1 : Pour chaque entiers x, Δ tels que $x > \delta \geq 2$, il existe une famille Δ -cover-free de x ensembles d'un ensemble de taille $m \leq 4(\Delta + 1)^2 \log^2(x)$.
- Fait 2 : Pour chaque $x \leq (11(\Delta + 1))^3$ et Δ , il existe une famille Δ -cover-free de x ensembles d'un ensemble de base de taille $m \leq (22(\Delta + 1))^2$.