Université Libre de Bruxelles - Faculté des Sciences - Département d'informatique
MEMO-F-400 : Travail de préparation au mémoire

Travail de préparation au mémoire
Problèmes de coloriage de graphes

Etudiant : Christophe DUMEUNIER
Promoteur : Mr. Jean CARDINAL

Introduction

Ce document consiste en une présentation du sujet de mémoire de Christophe Dumeunier pour l'année académique 2008-2009. Les sujets abordés ici traitent de différents problèmes et algorithmes de coloriage de graphes. La documentation choisie pour ce travail consiste principalement en deux ouvrages d'introduction [1],[2] sur la théorie des graphes, et en deux articles de J. Cardinal, S. Fiorini et G. Joret [3] et de T. Fukunaga, M. M. Halldórsson et H. Nagamochi [4].

Une présentation générale du problème de coloriage de graphes est faite en introduction. Le lecteur pourra alors se pencher de façon plus précise d'une part sur le problème de coloriage de graphes d'entropie minimum, et d'autre part sur les problèmes de coloriage de graphes pour différentes fonctions de coûts, telles que les fonctions concaves, sous-additives, et quelques résultats sur les fonctions monotones ; et ce dans des graphes pondérés, aux propriétés bien choisies.

Cette présentation a pour simple but de prendre contact avec le sujet, tout en le laissant suffisamment vaste pour différents projets.

Chapitre 1

Notions élémentaires sur les graphes et coloriages de graphes

Nous commencerons par présenter quelques notions élémentaires et indispensables à la poursuite du sujet traité, à savoir les coloriages de graphes. Par soucis de simplicité, les résultats et théorèmes présentés ne seront pas démontrés ni esquissés, la littérature traitant du sujet étant abondante. Nous nous calquerons sur les présentations faites par W. Kocay et D. L. Kreher dans *Graphs, Algorithms, and Optimization* [1] et par J. Gross et J. Yellen dans *Graph Theory and its applications* [2].

1.1 Présentation théorique

Tout d'abord, voici quelques définitions de graphes particuliers que nous serons amenés à rencontrer.

Nous appellerons *graphe d'intervalles* un graphe où chaque sommet est attribué à un intervalle ouvert sur la droite réelle, et pour lequel deux sommets sont joints par une arête si et seulement si les deux intervalles associés possèdent une intersection non vide. Un *graphe d'intervalles propres* est un graphe d'intervalles tel qu'aucun intervalle ne contienne entièrement un autre. De façon similaire, un *graphe d'arcs circulaires* est un graphe pour lequel chaque sommet est un arc ouvert sur un cercle c, deux sommets étant joints s'il y a intersection entre les arcs associés.

Un graphe est appelé *graphe cordal* si chacun de ses cycles de plus de trois noeuds possède une corde, c'est-à-dire que tout cycle de plus de trois noeuds possède un raccourci.

Soit L une collection de droites dans le plan telles que trois droites ne sont jamais concourantes. Soit V_L l'ensemble des intersections entre les droites de L, et soit E_L l'ensemble des segments des droites de L reliant les sommets de V_L. Le graphe $G_L(V_L, E_L)$ est appelé *graphe planaire engendré par* L.

Nous appellerons *coloriage des sommets d'un graphe* $G = (V, E)$ toute application $f : V \to C$ où C est un ensemble fini ou dénombrable de couleurs. Pour plus de facilités nous désignerons généralement les différentes couleurs à l'aide d'entiers positifs. Nous nommons *k-coloriage de sommets* (ou *k-coloriage* lorsque cela ne provoque pas d'ambiguïté) un coloriage de graphe utilisant exactement k couleurs distinctes.

La condition suivante, que nous poserons sur tous les problèmes de coloriage des sommets d'un graphe, rend la notion de coloriage très intéressante ; nous appelons *coloriage propre* d'un graphe un coloriage des sommets du graphe tel que deux sommets quelconques adjacents ne soient jamais coloriés de la même couleur. Sans condition supplémentaire sur les couleurs proprement dites (par exemple appliquer un prix par sommet colorié selon la couleur), cette notion revient à considérer des graphes dont les sommets sont partitionnés en ensembles stables, que nous appellerons *classes de couleur*. Nous dirons qu'un graphe est *k-coloriable* s'il est possible de trouver un k-coloriage propre. Nous décidons également que S est appelé un *k-sous-graphe* de G lorsque S est un sous-graphe de G et est k-coloriable.

Le *nombre chromatique* d'un graphe G, noté $\chi(G)$, est défini comme le nombre minimum de couleurs nécessaires pour obtenir un coloriage propre du graphe G. Nous parlerons alors de *graphe k-chromatique* lorsque $\chi(G) = k$.

Remarque 1.1 *Etant donné un graphe* $G = (V, E)$, $\chi(G) = 1 \Leftrightarrow E = \emptyset$.

Proposition 1.2 *Considérons un graphe* $G = (V, E)$ *possédant un sous-graphe* K_k *de* k *sommets qui soit complet. Alors* $\chi(G) \geq k$.

Nous appellerons *clique d'un graphe* un ensemble maximal de sommets formant un sous-graphe complet ; et notons $w(G)$ le nombre de sommets de la clique.

Proposition 1.3 *Etant donnés un graphe* $G = (V, E)$ *et un sous-graphe* $H = (V', E')$ *de* G, *nous avons* $\chi(G) \geq \chi(H)$.

Corollaire 1.4 *Soit un graphe* $G = (V, E)$, *alors* $\chi(G) \geq w(G)$.

Nous appelons *graphe parfait* un graphe dont le nombre chromatique de tout sous-graphe induit est égal à la taille de la plus grande clique du sous-graphe induit considéré.

Proposition 1.5 *Un graphe est parfait si et seulement si son complémentaire est parfait.*

Le théorème suivant fut prouvé en 2002 par M. Chudnovsky, N. Robertson, P. D. Seymour et R. Thomas [5] :

Théorème 1.6 *Un graphe est parfait si et seulement si ni lui ni son complémentaire ne contiennent de cycle impair induit de longueur au moins 5.*

Nous désignons par $G + H = (V_G \cup V_H, E_G \cup E_H \cup (V_G \times V_H))$ la *jointure* des deux graphes $G = (V_G, E_G)$ et $H = (V_H, E_H)$, c'est-à-dire le graphe composé des sommets de G et H, de toutes leurs arêtes respectives ainsi que de toutes les arêtes formées par les paires de sommets dans G et H.

Proposition 1.7 *Soit $G + H$ la jointure de deux graphes $G = (V_G, E_G)$ et $H = (V_H, E_H)$, le nombre chromatique de $G+H$ est donné par $\chi(G+H) = \chi(G)+\chi(H)$.*

Remarque 1.8
Si le graphe K_n est un graphe complet à n sommets, alors $\chi(K_n) = n$.
Si le graphe B est un graphe biparti, alors $\chi(B) = 2$.
Si le graphe T est un arbre, alors $\chi(T) = 2$.

Un graphe $G = (V, E)$ connexe est dit k-critique si $\chi(G) = k$ et si pour toute arête $e \in E$ nous avons $\chi(G - e) \leq k - 1$.

Proposition 1.9 *Soit $G = (V, E)$ un graphe k-critique, et soit $v \in V$ un sommet quelconque du graphe ; alors le sous-graphe $G - v$ est $(k - 1)$-coloriable.*

Théorème 1.10 *Cosidérons un graphe $G = (V, E)$ qui soit k-critique. Aucun sommet $v \in V$ n'a un degré inférieur à $k - 1$.*

Etant donné un graphe $G = (V, E)$, nous appelons *k-obstruction* (ou *obstruction k-chromatique*) un sous-graphe G' de G qui forcerait tout graphe H le contenant à avoir un nombre chromatique plus grand que k, c'est-à-dire $\chi(H) > k$. Une collection $\{G_i\}$ de graphes $(k + 1)$-critiques est appelée *ensemble complet d'obstructions k-chromatiques* si tout graphe $(k + 1)$-coloriable contient au moins un des G_i comme sous-graphe.

Proposition 1.11 *Etant donné un graphe $G = (V, E)$, nous avons $\chi(G) = \lceil \frac{|V|}{\mathrm{ind}(G)} \rceil$ où $\mathrm{ind}(G)$ est la taille du plus grand ensemble stable dans G.*

Théorème 1.12 *Pour tout graphe $G = (V, E)$, si $\delta_{max}(G)$ désigne le degré maximum des sommets de V, alors $\chi(G) \leq \delta_{max}(G) + 1$.*

Lemme 1.13 *Considérons un graphe $G = (V, E)$ connexe.*
Si G est non-régulier, alors $\chi(G) \leq \delta_{max}(G)$.
Si G est régulier, alors $\chi(G) \leq \delta_{max}(G) + 1$.

Théorème 1.14 (Théorème de Brooks)
Soit $G = (V, E)$ un graphe qui ne soit ni un graphe complet ni un cycle de longueur impaire. Alors $\chi(G) \leq \delta_{max}(G)$.

Proposition 1.15 *Soit L une collection de droites du plan telles que trois quelconques ne sont jamais concourantes, et soit $G_L(V_L, E_L)$ le graphe planaire engendré par L. Alors $\chi(G_L) \leq 3$.*

Considérons un graphe $G = (V, E)$ sans triangle tel que $\chi(G) \geq 3$, par exemple un cycle de 5 sommets. Nous étendons le graphe G au graphe G_M de la façon suivante : Pour chaque sommet $v \in V$, nous ajoutons un sommet v' adjacent aux mêmes sommets que v (mais non adjacent à v). Ensuite ajoutons un sommet v_0 adjacent à tous les sommets v' construits. Cette construction porte le nom de *construction de Mycielski.*

Lemme 1.16 *Soit un graphe G ne possédant pas de triangle comme sous-graphe, et soit G_M le graphe construit sur G par la construction de Mycielski. Alors G_M ne possède pas de triangle.*

Lemme 1.17 *Soient G et G_M définis comme ci-dessus. Si $\chi(G) = k$ alors $\chi(G_M) = k + 1$.*

Théorème 1.18 *Pour tout entier $k \geq 3$, il existe un graphe G ne possédant pas de triangle et tel que $\chi(G) \geq k$.*

Nous définissons le *polynôme chromatique* d'un graphe $G = (V, E)$ par $\pi(G, \lambda)$, le nombre de façons de colorier G en maximum λ couleurs distinctes.

Lemme 1.19 *Soit T un arbre de n sommets. Alors $\pi(T, \lambda) = \lambda(\lambda - 1)^{n-1}$.*

Théorème 1.20 *Soient $G = (V, E)$ un graphe et λ un entier positif. Si $\chi(G) = m$ alors $\frac{\lambda!}{(\lambda-m)!}$ divise $\pi(G, \lambda)$.*

1.2 Algorithmes et heuristiques

1.2.1 L'algorithme séquentiel

Ce premier algorithme fournit un coloriage propre $f : V \to C$ d'un graphe $G = (V, E)$ où $V = \{v_1, \dots, v_p\}$. Nous supposons que l'ensemble C des couleurs est totalement ordonné et nous définissons $c(v_i)$ la plus petite couleur non utilisée par les sommets adjacents à v_i.

Algorithme *Séquentiel*
1. **pour** $i \leftarrow 1$ **à** p
2. **faire** $f(v_i) \leftarrow c(v_i)$;
3. **renvoyer** f

1.2.2 L'algorithme LDF

L'algorithme LDF (Largest Degree First) est basé sur le fait qu'il semble difficile de colorier tard un sommet dont le degré est élevé. Ainsi LDF va se concentrer en priorité sur les sommets non-encore coloriés de degré maximum. L'algorithme fournit un coloriage propre $f : V \to C$ d'un graphe $G = (V, E)$ où $V = \{v_1, \dots, v_p\}$.

Algorithme *LDF*
1. **tant que** il y a des sommets à colorier
2. **faire** parmi les sommets de degré maximum de V, trouver celui qui a déjà le plus de couleurs parmi ses voisins et le nommer v ;
3. assigner la plus petite couleur à v : $f(v) \leftarrow c(v_i)$;
4. **renvoyer** f

1.2.3 L'algorithme de saturation

Cet algorithme est très proche du précédant, à la différence qu'il sélectionne un sommet de degré maximum parmi les sommets qui ont le plus de couleurs différentes chez leurs voisins. On note $t(v)$ le nombre de couleurs distinctes déjà définies sur les sommets voisins de v. L'algorithme fournit un coloriage propre $f : V \to C$ d'un graphe $G = (V, E)$ où $V = \{v_1, \dots, v_p\}$.

Algorithme *de Saturation*
1. **tant que** il y a des sommets à colorier
2. **faire** parmi les sommets u tels que $t(u)$ est maximum, sélectionner un sommet de degré maximum et le nommer v ;
3. assigner la plus petite couleur à v : $f(v) \leftarrow c(v_i)$;
4. **renvoyer** f

Chapitre 2

Coloriage d'entropie minimum

Cette section présente un résumé de l'article de J. Cardinal, S. Fiorini et G. Joret [3] traitant des coloriages de graphes minimisant l'entropie du coloriage. Nous parlons alors d'*entropie chromatique*.

2.1 Notions préliminaires

Considérons un graphe $G = (V, E)$ auquel on associe une distribution de probabilité P sur les sommets de V. La paire (G, P) est appelée *graphe probabiliste*. Par extension, nous notons $P[S]$ la somme $\sum_{x \in S} P[x]$ lorsque S est une partie de V. Si nous considérons un coloriage propre $\phi : V \to C = \mathbb{N}^+$ des sommets du graphe G et une variable aléatoire X de loi de probabilité P, l'entropie $H(\phi)$ du coloriage est définie comme l'entropie de la variable $\phi(X)$, c'est-à-dire $H(\phi) = -\sum_i c_i \log(c_i)$ où $c_i = P[\phi^{-1}(i)]$. L'*entropie chromatique* $H_\chi(G, P)$ est définie comme la plus petite entropie sur tous les coloriages propres de G.

Tout coloriage d'un graphe probabiliste (G, P) fournit une *séquence chromatique* $c = c_1, c_2, c_3, \ldots$. Une telle séquence est dite faisable s'il existe un graphe (G, P) et un coloriage ϕ la fournissant. Nous restreindrons généralement aux séquences c décroissantes, ce qui s'obtient aisément en renommant les couleurs. Notons alors que $H(\phi) = H(c)$. Nous dirons d'une séquence c d'un graphe probabiliste (G, P) qu'elle *domine* une autre séquence d du même graphe (et nous notons $c \succeq d$) si $\sum_{i=1}^{j} c_i \geq \sum_{i=1}^{j} d_i$ pour tout $j > 0$. Une séquence m décroissante d'un graphe (G, P) est dite *maximale* lorsqu'elle n'est dominée par aucune autre séquence décroissante de (G, P).

Les trois lemmes suivants sont la clef des résultats sur l'entropie présentés dans l'article [3] :

Lemme 2.1 (présenté par Alon et Orlitsky en 1996)
Soit c une séquence chromatique décroissante. Soient i,j et α tels que $i < j$ et $0 < \alpha \leq c_j$. Alors $H(c) > H(c_1, \ldots, c_{i-1}, c_i + \alpha, c_{i+1}, \ldots, c_{j-1}, c_j - \alpha, c_{j+1}, \ldots)$.

Lemme 2.2 *Soient c et d deux séquences rationnelles décroissantes telles que $c \succ d$. Alors $H(c) < H(d)$.*

La démonstration fournie s'appuie itérativement sur le lemme 2.1.

Lemme 2.3 *Etant donné un graphe probabiliste $(G = (V, E), P)$ tel que $P(v) > 0$ pour tout sommet $v \in V$. Soit ϕ un coloriage de (G, P) d'entropie minimum. Considérons c une séquence chromatique associée qui soit décroissante. Alors la i-ème classe de couleur de ϕ est un ensemble stable maximum dans le sous-graphe de G induit par les couleurs supérieures ou égale à la couleur i.*

La démonstration, par l'absurde, se sert du lemme 2.1 pour réduire l'entropie et aboutir à la contradiction.

2.2 Complexité et approximabilité

Le premier résultat présenté concerne la complexité de la recherche d'une méthode pour trouver un coloriage d'entropie minimum dans un graphe d'intervalles probabiliste.

Théorème 2.4 *Trouver un coloriage d'entropie minimum d'un graphe d'intervalles probabiliste (G, P) est fortement $\mathcal{NP}$-difficile.*

La démonstration se fait à l'aide d'une réduction au problème qui doit décider si un graphe d'arcs circulaires C est k-coloriable, ce problème étant connu pour être $\mathcal{NP}$-complet (Garey, Johnson, Miller et Papadimitriou, 1980). L'idée de la preuve est de considérer une instance du problème de coloriage de graphes d'arcs circulaires, et de couper le cercle à un endroit p bien choisi, et le dérouler pour obtenir une collection de segments, c'est-à-dire un graphe d'intervalles. Il faudra bien-sûr que dans cette instance les segments aux extrémités (qui sont issus d'un arc coupé) soient de même couleur dans le coloriage. Choisissant un point p du cercle qui ne soit pas à l'extrémité d'un intervalle, on ajoute suffisamment de *petits* arcs autour de p afin que p soit contenu dans exactement k arcs, ce qui forme une clique de taille k dans le graphe d'arcs sans augmenter l'entropie chromatique. Bien entendu, si p appartient à plus de k segments, l'instance peut immédiatement être rejetée. Un algorithme de Golumbic (2004) permet de lister en temps polynomial les cliques maximales du graphe d'intervalles obtenu. A nouveau, si la taille est supérieure à k, on rejette l'instance. Dans le cas contraire, comme précédemment, on peut ajouter à chacune de ces cliques maximales des *petits* segments pour augmenter leur taille à k sans augmenter l'entropie. Un graphe biparti contenant d'une part les intervalles du nouveau graphe d'intervalles et d'autre part les cliques maximum de ce même graphe, avec adjacence selon l'appartenance aux cliques, est construit et muni d'une distribution de probabilité bien choisie, générant une séquence chromatique faisable si le problème initial est k-coloriable et dominant toute séquence faisable admettant la condition des segments extrêmes (issus du point de découpe) de même couleur. Cela permet de conclure à l'aide du lemme 2.2.

Le second résultat que propose l'article présente le lien entre le problème d'entropie minimum et le problème classique de minimisation du nombre de couleurs, dans le cadre des algorithmes polynomiaux. La fonction objectif prenant ses valeurs entre 0 et $\log(n)$, il est logique d'appeler *δ-approximation* la solution d'un algorithme fournissant une solution dont l'entropie n'excède pas $H_\chi(G, P) + \delta$. A noter que l'algorithme coloriant chaque sommet de couleur différente est un algorithme linéaire fournissant une $\log(n)$-approximation.

Théorème 2.5 *Soit c un nombre réel tel que $0 < c \leq 1$ et supposons que pour un certain réel $\epsilon > 0$ il existe un algorithme A approximant le problème de coloriage d'entropie minimum avec un facteur $(c - \epsilon)\log(n)$. Alors il existe un algorithme en temps polynomial coloriant un graphe G avec au plus $n^{c-\epsilon/2}\chi(G)$ couleurs.*

La preuve commence par démontrer qu'il existe une classe de couleur dans le coloriage ϕ fourni par l'algorithme A sur le graphe (G, U) (où U représente la distribution uniforme sur les sommets de G) contenant au moins $n^{1-c+\epsilon}/\chi$ sommets. Nous pouvons alors considérer un algorithme polynomial A' utilisant la sous-routine A pour trouver dans un graphe G de n sommets et de nombre chromatique χ un ensemble stable de taille au moins $n^{1-c+\epsilon}/\chi$. Un algorithme A'' peut être construit en utilisant itérativement A' pour colorier le graphe en coloriant de la même couleur les sommets d'un ensemble stable, puis en retirant ces sommets du graphe. La suite de la preuve montre alors que le nombre de couleurs est une grandeur $\ell + 1 \leq n^{c-\epsilon/2}\chi$ où ℓ est le nombre d'itérations appelant A', ce qui termine de prouver le théorème.

La classe de complexité $\mathcal{ZPP}$ est définie comme l'ensemble des problèmes qui peuvent être résolus par un algorithme randomisé qui retourne toujours la bonne réponse et dont le temps d'exécution est polynomial pour toute instance. Feige et Kilian avaient montré qu'il n'existe pas d'algorithme polynomial coloriant un graphe G avec au plus $n^{1-\epsilon}\chi(G)$ couleurs, à moins que $\mathcal{ZPP}=\mathcal{NP}$. D'autre part Bellare, Goldreich et Sudan avaient montré que si le nombre de couleurs utilisées par un tel algorithme est borné par $n^{1/7-\epsilon}\chi(G)$ alors cela implique que $\mathcal{P}$=$\mathcal{NP}$. Combinant cela au théorème ci-dessus, nous obtenons les corollaires suivants :

Corollaire 2.6 *Soit un réel $\epsilon > 0$. Il n'existe pas d'algorithme de $(1-\epsilon)\log(n)$-approximation pour le problème de coloriage d'entropie minimum, à moins que $\mathcal{ZPP}$=$\mathcal{NP}$.*

Corollaire 2.7 *Soit un réel $\epsilon > 0$. Il n'existe pas d'algorithme de $(1/7-\epsilon)\log(n)$-approximation pour le problème de coloriage d'entropie minimum, à moins que $\mathcal{P}$=$\mathcal{NP}$.*

Il existe cependant certains graphes permettant un algorithme polynomial. En rappelant que $\alpha(G)$ désigne la taille du plus grand ensemble stable dans un graphe G, nous avons le théorème suivant :

Théorème 2.8 *Il existe un algorithme en temps polynomial pour résoudre le problème de coloriage d'entropie minimum d'un graphe lorsque nous nous restreignons à la classe des graphes G tels que $\alpha(G) \leq 2$.*

La démonstration s'appuie sur le fait que chaque classe de couleur est un sommet ou une paire de sommets, en particulier les sommets et arêtes du graphe complémentaire G^c. Les combinaisons possibles ne se font que sur des paires de sommets, ce qui permet d'obtenir un algorithme en temps polynomial.
Remarque : Une section de l'article est consacrée à l'entropie de Körner, mais cette notion ne nous intéresse pas ici.

2.3 Nombre de couleurs

Une troisième section de l'article se penche sur le nombre minimum de couleurs dans un coloriage d'entropie minimum sur un graphe probabiliste (G, P), ce nombre est noté $\chi_H(G, P)$. Il faut avant cela rappeler une notion déjà présente dans la littérature. Nous appellerons *coloriage de Grundy* d'un graphe, un coloriage tel que pour toute couleur c, si un sommet est de couleur c, alors il est adjacent à au moins un sommet de couleur c' pour tout $c' < c$. Le *nombre de Grundy* $\Gamma(G)$ d'un graphe G est le nombre maximum de couleurs dans un coloriage de Grundy du graphe G.

Proposition 2.9 *Tout coloriage d'entropie minimum d'un graphe probabiliste (G, P) est un coloriage de Grundy. De plus, pour tout coloriage de Grundy ϕ du graphe G, il existe une distribution de probabilité P_ϕ sur les sommets de G telle que ϕ est un coloriage d'entropie minimum unique de G.*

La première partie de la preuve est donnée par le lemme 2.3. La seconde partie est prouvée par induction sur le nombre k de couleurs utilisées dans ϕ.
A cela s'ajoute un résultat démontré auparavant par les auteurs de l'article :

Proposition 2.10 *$\chi_H(G, P)$ n'est borné par aucune fonction de $\chi(G)$, même si P est la distribution uniforme et G un arbre.*

Par contre, il est possible d'observer des bornes sur $\chi_H(G, P)$ comme fonction du degré maximum $\delta_{max}(G)$ de G.

Lemme 2.11 *Pour tout graphe probabiliste (G, P), nous avons $\chi_H(G, P) \leq \delta_{max}(G) + 1$.*

Le théorème de Brooks ne peut être étendu en substituant $\chi_H(G, P)$ à $\chi(G)$ sans faire de supposition sur P. C'est ce que nous dit la proposition 2.9. Cependant pour une distribution uniforme, nous pouvons observer le résultat suivant :

Théorème 2.12 *Si G est un graphe connexe qui ne soit ni complet si un cycle impair, alors $\chi_H(G, U) \leq \delta_{max}(G)$, où U est la distribution uniforme sur les sommets de G.*

Pour $\delta_{max}(G) \leq 2$ la preuve est triviale. L'idée de la preuve pour $\delta_{max}(G) \geq 3$ est de considérer un coloriage de (G, U) d'entropie minimum ϕ dont les couleurs sont comprises dans $\{1, 2, \ldots, \delta_{max}(G) + 1\}$ et de montrer que si les $\delta_{max}(G) + 1$ couleurs sont utilisées, alors ϕ ne peut être un coloriage de (G, U) que si G est complet.

Chapitre 3

Coloriages à coûts robustes

Nous nous penchons, dans cette section, sur l'article *Robust Cost Coloring* de T. Fukunaga, M. M. Halldórsson et H. Nagamochi [4].

3.1 Notions et notations

Soit $G = (V, E)$ un graphe. Pour chaque sommet $v \in V$ on associe un poids w_v. Notons que c'est équivalent à regarder une distribution de probabilité en posant $w'_v = \frac{w_v}{\sum_{u \in V} w_u}$. Nous notons également $w(S) = \sum_{u \in S} w_u$ et $w(G) = w(V)$.

La notation *k-MCS* fait référence au problème de trouver un k-sous-graphe de poids total maximal laissant l'appellation *coloriage de graphes* au problème classique utilisant le nombre chromatique de couleurs.

Nous admettrons également les notations suivantes :
Pour un ensemble S et un sommet u, $S{+}u$ désigne $S \cup \{u\}$. Une fonction $f : 2^V \to \mathbb{R}$ est dite *monotone* si $f(S + u) \geq f(S)$ pour tout ensemble S et sommet u ; f est dite *séparable* si $f(S) = f(T)$ lorsque $w(S) = w(T)$; f est dite *polymatroïdale* si $f(S + u) - f(S) \geq f(T + u) - f(T)$ lorsque $f(T) \geq f(S)$. Nous dirons que la fonction f est *représentée* par une fonction $f' : \mathbb{R} \to \mathbb{R}$ si $f(S) = f'(w(S))$ pour tout S. Une fonction séparable f est *sous-additive* si $f(x) = \lceil \frac{x}{x'} \rceil \min_{x' \leq x} f(x')$ et *concave* si $f((x + y)/2) = (f(x) + f(y))/2$. Notons que les fonctions séparables concaves sont aussi polymatroïdales, l'inverse n'est pas vrai en général, excepté lorsque les poids sont uniformes.

3.2 L'algorithme ACS

T. Fukunaga, M. M. Halldórsson et H. Nagamochi présentent tout d'abord un algorithme pouvant s'adapter à de nombreuses fonctions de coût pour résoudre le problème de coloriage de graphe.

Algorithme *ACS*
1. définir $\ell_0 \leftarrow \lceil \log_2(\chi(G)) \rceil$;
2. définir $G_0 \leftarrow G$;
3. **pour** $i \leftarrow 0$ **à** $\ell_0 - 1$
4. **faire** définir B_i un 2^i-sous-graphe de G_i de poids maximum ;
5. colorier B_i avec 2^i couleurs ;
6. $G_{i+1} \leftarrow G_i \setminus B_i$;
7. colorier le graphe restant avec au plus $\chi(G)$ couleurs ;

Nous pouvons observer que la dernière itération de la boucle utilise au plus $2^{\ell_0 - 1} \leq \chi(G) - 1$ couleurs, et le nombre total de couleurs utilisées avant ce point étaient $\sum_{i=0}^{\ell_0 - 2} 2^i = 2^{\ell_0 - 1} - 1$. En ajoutant les $\chi(G)$ couleurs finales, on obtient le résultat suivant :

Proposition 3.1 *L'algorithme ACS utilise au plus* $3(\chi(G) - 1)$ *couleurs.*

Il se trouve que l'algorithme ACS peut s'appliquer au coloriage de tout graphe d'intervalle pondéré, quelle que soit la fonction de coût utilisée, à condition qu'elle soit concave. C'est ce que nous dit le théorème énoncé ci-dessous. Sa démonstration nécessite un lemme similaire au lemme 2.1 :

Lemme 3.2 *Soit* f *une fonction concave et soient* $x_0, x_1, \ldots, x_t, y_0, y_1, \ldots, y_t$ *des réels positifs tels que*

$$\sum_{i=j}^{t} y_i \leq \sum_{i=j}^{t} x_i \text{ pour tout } j = 0, 1, \ldots, t$$

Supposons de plus que $x_0 \geq x_1 \geq \cdots \geq x_t$, *alors*

$$\sum_{i=0}^{t} f(y_i) \leq \sum_{i=0}^{t} f(x_i).$$

Ce lemme est démontré facilement par l'absurde.

Théorème 3.3 *L'algorithme ACS sur un graphe d'intervalles pondéré trouve un coloriage qui approxime toute fonction de coût concave avec un facteur* 4.

La preuve se base sur la propriété suivante : la concavité de f implique que $f(x) \leq rf(x/r)$ pour tout $r \geq 1$ et pour tout x. Nous pouvons supposer qu'un coloriage optimal pour f utilise les couleurs $0, 1, \ldots, \chi_{opt}$ classées par ordre décroissant de poids. En posant B_i l'ensemble des couleurs utilisées à l'itération i par ACS, et b_i le poids total de B_i ; en posant A_i l'ensemble des couleurs $\{2^{i-1}, \ldots, 2^i - 1\}$ dans la solution optimale, et a_i le poids total de A_i ; et en définissant $ACS(B_i)$ le coût de B_i dans la solution de l'algorithme ACS et OPT le coût total de la solution optimale, la preuve présente les résultats suivants :

$$\begin{aligned}\sum_{i=0}^{\ell} ACS(B_i) &\leq \textstyle\sum_{i=0}^{\ell} 2^i \cdot f(b_i/2^i) \leq & \sum_{i=0}^{\ell} 2^i \cdot f(a_i/2^i) \\ OPT &= \textstyle\sum_i \sum_{I \in A_i} f(w(I)) \geq & f(a_0) + \sum_{i=2}^{\ell} 2^{i-2} \cdot f(a_i/2^{i-1})\end{aligned}$$

où $\ell = \lceil \log_2(\chi_{opt}) \rceil$.
Ces résultats permettent facilement de conclure à un rapport de 4.

Théorème 3.4 *Considérons $G = (V, E)$, un graphe parfait. Un coloriage par ACS du graphe G approxime la solution optimale du problème de coloriage pour une quelconque fonction de coût concave avec un facteur* 6.

La preuve donnée est similaire à la preuve du théorème 3.3 précédant.

Un troisième théorème donne une borne d'approximabilité pour ACS, sur les fonctions sous-additives cette fois. Pour cela, remarquons que si f est séparable, la sous-additivité implique que $f(x) = \min_{x' \leq x} f(x') \lceil \frac{x}{x'}$. Nous dirons qu'une fonction f est *ρ-approximée* par une autre fonction $\hat{f}$ si $f(x) \leq \hat{f}(x) \leq \rho \cdot f(x)$ pour tout x sur le domaine de f.

Lemme 3.5 *Soit f' une fonction de coût sous-additive et séparable définie sur les réels positifs. Alors il existe une fonction concave séparable $\hat{f}$ qui* 2*-approxime f'.*

La démonstration de ce lemme n'a pas d'intérêt, par contre il signifie que l'algorithme ACS pourra approximer une large classe de fonctions qui ne sont pas concaves. En effet :

Théorème 3.6 *L'algorithme ACS appliqué à un graphe d'intervalles pondéré approxime le problème de coloriage pour une fonction de coût f sous-additive avec un facteur* 8.

3.3 Fonctions de coût arbitraires

Nous dirons qu'un coloriage ϕ' *raffine* un coloriage ϕ si chaque classe de couleur dans ϕ est une union de classes de ϕ'.

Théorème 3.7 *Considérons un graphe $G = (V, E)$ dont les poids sont uniformes, et soit f une fonction de coût monotone. Considérons un coloriage ϕ fourni par l'algorithme ACS sur le graphe G qui forme une ρ-approximation pour chaque fonction de coût concave. Alors il existe un coloriage ϕ' qui affine ϕ et qui est une 2ρ-approximation de f.*

La démonstration, très courte, s'appuie sur le lemme 3.5 proposé précédemment.

Corollaire 3.8 *Soit f une fonction de coût monotone. Dans le cas particulier des poids uniformes, il existe un algorithme qui trouve un coloriage selon f qui forme une 8-approximation sur les graphes d'intervalles et une 12-approximation sur les graphes parfaits.*

3.4 Difficultés et Algorithmes exacts

Cette section fournit quelques résultats difficiles à traiter sur différents types de graphes, et ce sans être sous la dépendance de suppositions théoriques telles que $\mathcal{P}\neq\mathcal{NP}$.

Théorème 3.9 *Il n'existe pas de coloriage ϕ d'un graphe G fournissant une c-approximation pour toute fonction concave séparable pour certains $c > 1,14$.*

La preuve consiste en un exemple sur le graphe d'intervalles possédant q copies de chacun des intervalles suivants : $[1, 7], [2, 3], [4, 5], [6, 12], [8, 9], [10, 11]$.

Proposition 3.10 *Soit f une fonction de coût séparable et fortement concave. Alors il est fortement $\mathcal{NP}$-difficile de trouver un coloriage selon f qui soit optimal sur un graphe d'intervalles.*

Théorème 3.11 *Soit f une fonction concave non-additive. Il est $\mathcal{NP}$-difficile de trouver un coloriage selon f sur les graphes planaires de degré 3.*

Il existe cependant des classes de graphes pour lesquels des algorithmes polynomiaux fournissent une solution optimale au problème de coloriage.

Théorème 3.12 *Soit f une fonction de coût et G un graphe tel que $\alpha(G) = 2$. Alors un coloriage de coût minimum selon f peut être calculé en temps polynomial.*

Théorème 3.13 *Soit $G = (V, E)$ un graphe d'intervalles propre avec une répartition de poids uniforme. Il existe un algorithme en temps polynomial fournissant un coloriage de G qui optimise simultanément toute fonction de coût concave.*

Bibliographie

[1] W. Kockay et D. L. Kreher. *Graphs, Algorithms, and Optimization*. Chapman & Hall/CRC Press. 2005.

[2] J. Gross et J. Yellen. *Graph Theory and its applications*. CRC Press. 1999.

[3] J. Cardinal, S. Fiorini et G. Joret. *Minimum Entropy Coloring*. Journal of Combinatorial Optimization. 2008.

[4] T. Fukunaga, M. M. Halldórsson et H. Nagamochi. *Robust Cost Coloring*. ACM-SIAM Symposium on Discrete Algorithms, January 20-22, 2008 Holiday Inn Golden Gateway San Francisco, California, pp. 1204-1212.

[5] M. Chudnovsky, N. Robertson, P. D. Seymour et R. Thomas. *Progress on perfect graphs*. Math Programming, Ser. B, 97 (2003), 405-422.

Table des matières

Introduction **2**

1 Notions élémentaires sur les graphes et coloriages de graphes **3**
- 1.1 Présentation théorique 3
- 1.2 Algorithmes et heuristiques 7
  - 1.2.1 L'algorithme séquentiel 7
  - 1.2.2 L'algorithme LDF 7
  - 1.2.3 L'algorithme de saturation 7

2 Coloriage d'entropie minimum **8**
- 2.1 Notions préliminaires 8
- 2.2 Complexité et approximabilité 9
- 2.3 Nombre de couleurs 11

3 Coloriages à coûts robustes **13**
- 3.1 Notions et notations 13
- 3.2 L'algorithme ACS 13
- 3.3 Fonctions de coût arbitraires 16
- 3.4 Difficultés et Algorithmes exacts 16

Bibliographie **17**

Table des matières **18**