

République Algérienne Démocratique et Populaire
Ministère d'Enseignement Supérieur et de la Recherche Scientifique
Université Abderrahmane Mira de Béjaia
Faculté des Sciences Exactes
Département de Recherche Opérationnelle



Mémoire de fin de cycle
En Vue de l'obtention du diplôme de Master en mathématiques
appliquées
Spécialité : Modélisation Mathématique et Evaluation de Performance Des
Réseaux

Thème

Quelques techniques de coloration des
sommets dans un graphe

Réalisé par :
BOUNECHACHE Chahira
BENSMAIL Salma

Encadré par :
M^r K. Kabyle

Devant le jury :
Président *M^r* : TALEM Djamel
Examinatrice *M^{me}* YOUNSI leila

Année universitaire 2019/2020

Remerciements

Ce n'est pas par coutume que cette page figure au préambule de ce mémoire, mais c'est plutôt un devoir moral et une reconnaissance sincère qui nous poussent à la faire.

Au terme de ce travail, nous tenons à remercier le bon Dieu tout puissant d'avoir guidé notre pas vers les portes du savoir tout en illuminant notre chemin, et nous avoir donné suffisamment de courage et de persévérance pour mener notre travail à terme.

C'est avec un grand plaisir que nous exprimons nos gratitude et nos sincères remerciements à notre promoteur : *Mr KABYL.K* qui a accepté de diriger ce travail et pour tous les conseils et l'aide qu'il nous a prodigué et l'encadrement qu'il nous a offert dans l'élaboration de ce mémoire de fin d'étude.

Nous remercions vivement les membres du jury *Mr* TALEM et *M^{me}* YOUNSI d'avoir accepté d'examiner ce travail. Leurs remarques et leurs conseils nous permettront sûrement de corriger et d'améliorer ce document.

Toutes nos reconnaissances sont adressées à tous les enseignants qui nous ont suivi infatigablement durant tout notre cursus universitaire.

Nous tenons à exprimer tout au fond de nos coeurs les reconnaissances à

nos familles qui nous ont offert toujours un appui sûr par leurs soutiens et leurs encouragements.

Enfin, Nous remercions tous ceux qu'ont contribués de loin ou de près à l'aboutissement de ce travail.

Dédicaces

Je dédie ce modeste travail

AUX personnes les plus chères à mon Coeur mes parents, source d'amour, de courage et d'encouragement (que dieu les garde).

A Mes chères soeurs Kanza, Siham, Silya et Silina, pour leurs encouragements et leur soutien moral.

A Mes chers frères, Samir, Ilyes et Halim pour leurs appuis et leurs encouragements.

A ma chère soeur Nassima et son mari Farid et leur enfants Amine, Islam et Ayan.

A mes tantes et leurs familles.

A mes oncles et leurs familles.

A ma binôme chahira et sa famille.

A tout(e) mes ami(e).

A mes professeurs surtout mon promoteur M^r Kabyle.

A tous ceux qui ont contribué de près ou de loin à l'édition de ce rapport.

** Salma Bensmail **

Dédicaces

Je dédie ce modeste travail

AUX personnes les plus chères à mon Coeur mes parents source d'amour, de courage et d'encouragement (que dieu les garde).

A Mes chères soeurs Nacera, Nadia et wissam pour leurs encouragement et leur soutien moral.

A Mon cher frère Hicham pour son appuis et encouragements.

A ma chère grande mère que je la souhaite une longue vie.

A mes tantes et leurs familles.

A mes oncles et leurs familles.

A ma binôme salma et sa famille.

A tout (es) mes ami(es).

A mes professeurs surtout mon promoteur M^r Kabyle.

A tous ceux qui ont contribué de près ou de loin à l'édition de ce rapport.

** Chahira Bounechache **

TABLE DES MATIÈRES

| | |
|--|----------|
| Table des matières | 6 |
| Table des figures | 6 |
| Introduction générale | 7 |
| 1 Notions de bases | 9 |
| 1.1 Définitions et concepts de graphe | 9 |
| 1.1.1 Graphe | 9 |
| 1.1.2 Boucle | 10 |
| 1.1.3 Graphe simple | 10 |
| 1.1.4 Graphe orienté | 10 |
| 1.1.5 Graphe non orienté | 11 |
| 1.1.6 Degrés | 11 |
| 1.1.7 Successeurs, Prédécesseurs d'un sommet | 12 |
| 1.1.8 Graphe partiel | 13 |
| 1.1.9 Sous-graphe | 13 |
| 1.1.10 Sous-Graphe Partiel | 14 |
| 1.1.11 Une clique | 14 |
| 1.2 chaînes, cycles, chemins et circuits | 15 |
| 1.3 Connexité et forte connexité | 16 |
| 1.3.1 Connexité | 16 |
| 1.3.2 Forte connexité | 16 |

| | | |
|----------|--|-----------|
| 1.4 | Mode de représentation graphique | 16 |
| 1.4.1 | Matrice d'adjacence | 17 |
| 1.4.2 | Matrice d'incidence | 18 |
| 1.4.3 | Liste d'adjacence | 19 |
| 1.5 | Quelques classes de graphes | 20 |
| 1.5.1 | Graphe complet : | 20 |
| 1.5.2 | Graphe biparti et biparti complet : | 21 |
| 1.5.3 | Graphe planaire | 21 |
| 1.5.4 | Graphe complémentaire | 22 |
| 1.5.5 | Graphe triangulé et non triangulé | 23 |
| 1.5.6 | Graphe représentatif des arêtes | 23 |
| 1.5.7 | Graphe K-régulier | 24 |
| 1.5.8 | Arbre et Arborescence | 24 |
| 2 | Coloration des sommets dans un graphe | 26 |
| 2.1 | Introduction | 26 |
| 2.2 | Historique | 26 |
| 2.3 | Domaines d'applications | 27 |
| 2.4 | Coloration simple des sommets | 27 |
| 2.4.1 | Ensemble stable | 29 |
| 2.4.2 | Nombre Chromatique | 29 |
| 2.4.3 | Détermination du nombre chromatique | 30 |
| 2.4.4 | Bornes pour le nombre chromatique | 31 |
| 2.4.5 | Encadrement du nombre chromatique | 31 |
| 2.4.6 | Nombre chromatique de graphes classique | 34 |
| 2.4.7 | Complexité | 34 |
| 2.4.8 | Problème de coloration de sommets dans un graphe | 35 |
| 2.5 | Quelques types de coloration de sommets | 36 |
| 2.5.1 | Coloration par listes des sommets | 36 |
| 2.5.2 | Coloration forte | 38 |
| 2.5.3 | Coloration forte stricte | 39 |
| 2.5.4 | Coloration K-équitable | 40 |
| 2.5.5 | Coloration avec satisfaction de contraintes CSP | 40 |

| | | |
|----------|---|-----------|
| 2.5.6 | Coloration par sommes | 43 |
| 2.6 | Coloration des graphes parfaits | 44 |
| 2.7 | Coloration des graphes planaires | 45 |
| 2.8 | Conclusion | 46 |
| 3 | Quelques algorithmes de coloration | 47 |
| 3.1 | Introduction | 47 |
| 3.2 | Graphes quelconques : des algorithmes d'approximation | 47 |
| 3.2.1 | Coloration séquentielle : l'algorithme glouton | 47 |
| 3.2.2 | Une première heuristique, Welsh & Powell | 49 |
| 3.2.3 | Deuxième heuristique, DSATUR | 54 |
| 3.3 | Graphes triangulés : un algorithme linéaire de coloration optimale | 58 |
| 3.3.1 | Généralités sur les graphes triangulés | 58 |
| 3.3.2 | Parcours en largeur lexicographique - LexBFS | 58 |
| 3.4 | Les métaheuristiques | 60 |
| 3.4.1 | Les Algorithmes de recherche locale | 60 |
| 3.4.2 | Les algorithmes génétiques | 61 |
| 3.4.3 | Les algorithmes hybrides | 61 |
| 3.5 | Conclusion | 61 |
| 4 | Application | 62 |
| 4.1 | Introduction au Code Block | 62 |
| 4.2 | Allocation des fréquences (Gestion du spectre de fréquence et implémentation des réseaux de télécommunications) | 64 |
| 4.2.1 | Introduction | 64 |
| 4.2.2 | Quelques définitions | 65 |
| 4.2.3 | Les interférences | 65 |
| 4.2.4 | Problématique | 66 |
| 4.2.5 | Modélisation mathématique du problème par la théorie des graphes | 66 |
| 4.2.6 | Problèmes tests | 67 |
| 4.2.7 | Coloration de graphe d'interférences | 68 |
| 4.3 | Coloration des cartes géographiques | 71 |
| 4.3.1 | Introduction | 71 |
| 4.3.2 | Coloration de la carte de Béjaia | 72 |

TABLE DES FIGURES

| | | |
|------|--|----|
| 1.1 | Graphe simple et graphe non simple. | 10 |
| 1.2 | Graphe orienté. | 11 |
| 1.3 | Graphe non orienté. | 11 |
| 1.4 | Successesurs, prédécesseurs d'un sommet. | 13 |
| 1.5 | Graphe G et l'un de ses graphes partiels. | 13 |
| 1.6 | Sous graphe d'un graphe G | 14 |
| 1.7 | Cliques. | 14 |
| 1.8 | Graphe connexe. | 16 |
| 1.9 | Graphe G | 17 |
| 1.10 | Graphe orienté. | 18 |
| 1.11 | Graphe non orienté. | 19 |
| 1.12 | Liste d'adjacence. | 20 |
| 1.13 | Exemple de graphes complets. | 20 |
| 1.14 | Représentation planaire du K_4 | 21 |
| 1.15 | Graphe G et son complémentaire. | 22 |
| 1.16 | Graphe triangulé, Graphe non triangulé. | 23 |
| 1.17 | Graphe et son graphe représentatif des arêtes. | 23 |
| 1.18 | Graphe 4-régulier. | 24 |
| 1.19 | Arbre. | 25 |
| 1.20 | Arborescence. | 25 |
| 2.1 | Coloration des sommets d'un graphe G | 28 |
| 2.2 | Graphe 3-colorable : classes de couleurs | 29 |

| | | |
|------|---|----|
| 2.3 | Coloration de graphe G | 30 |
| 2.4 | Le graphe qui modélise le problème de planification d'examens | 36 |
| 2.5 | Un biparti qui n'est pas 2-liste coloration. | 37 |
| 2.6 | Un graphe biparti de choisissabilité trois | 38 |
| 2.7 | Dominance X_s coloration | 39 |
| 2.8 | Coloration 3-équitable | 40 |
| 2.9 | Graphe de contraintes | 41 |
| 2.10 | Problème de coloriage de carte et son graphe des contraintes associé. | 42 |
| 2.11 | Exemple de coloration par sommes.. | 43 |
| 2.12 | Un graphe planaire coloré avec 4-couleurs. | 46 |
| 3.1 | Déroulement de l'algorithme Lex-BFS sur un exemple | 60 |
| 4.1 | Un réseau d'antennes. | 65 |
| 4.2 | Graphe d'interférences correspondant au tableau ci-dessus. | 68 |
| 4.3 | Graphe d'interférences correspondant au tableau ci-dessus. | 68 |
| 4.4 | Coloration obtenue par l'algorithme de Welsh and Powell. | 70 |
| 4.5 | Les Dairas de la Wilaya de Béjaia. | 72 |

INTRODUCTION GÉNÉRALE

L'application des mathématiques dans divers domaines a connu un essor spectaculaire ces dernières décennies. Avec le développement prodigieux des ordinateurs, les mathématiques discrètes en particulier sont devenues incontournables en informatique, en théorie des jeux dans les problèmes de décision et dans la plupart des sciences vu qu'elles s'occupent de l'étude de structures algébriques ou combinatoires finies. La théorie des graphes, une branche des mathématiques discrètes est avérée être particulièrement utile pour un grand nombre de domaines vu sa richesse en résultats théoriques et même d'applications aux problèmes du monde réel.

L'un des domaines le plus étudié en théorie des graphes est celui de la coloration des graphes. En effet, de nombreux problèmes réels tels que les problèmes d'ordonnements ou encore d'allocations de fréquences se traduisent comme une partition d'ensemble d'objets (sommets, arêtes, faces,...) en plusieurs classes suivant certaines contraintes. La contrainte la plus courante est celle de la propreté, qui consiste à affecter à deux éléments voisins des couleurs différentes. Il existe une grande variété de problèmes de colorations où il est question de rechercher des structures optimales particulières telle que le nombre minimum de couleurs d'une coloration propre des sommets d'un graphe. Ce dernier problème figure à ce jour parmi les paramètres fondamentaux dont l'origine remonte au problème des quatre couleurs.

Dans ce présent mémoire, nous nous intéresserons à l'étude de la coloration des sommets de graphes, voici un résumé des différents chapitres qui composent ce travail :

Le premier chapitre est consacré essentiellement aux notions de base, nous donnons des définitions fondamentaux et rappels sur les notions de base en théorie des graphes que nous utiliserons

aux cours de ce mémoire.

Le deuxième chapitre porte sur la coloration des sommets de graphes, nous commençons par donner quelques définitions et propriétés, ensuite nous présentons la caractérisation et quelques résultats concernant la coloration de quelques graphes particulières à savoir les graphes planaires et les graphes parfaits.

Dans le troisième chapitre, nous présentons quelques algorithmes de reconnaissance de certains graphes particuliers ainsi les algorithmes permettant de colorier les graphes, parmi ces algorithmes, on a Welsh and Powell qui sera utilisé par la suite dans le chapitre d'application.

Enfin, dans le dernier chapitre nous avons illustré ce travail par des exemples pratiques sur la coloration des graphes dans le monde réel tels que le problème d'allocation de fréquence et la coloration des cartes géographiques.

L'implémentation en C de cet algorithme appliqué à ces applications réelles présentés dans le chapitre précédent.

Une conclusion finale nous a permis de résumer tout ce que nous avons fait dans ce mémoire.

CHAPITRE 1

NOTIONS DE BASES

Avant d'entamer notre travail il est nécessaire de commencer par énoncer certaines définitions et concepts que nous avons jugé être nécessaires à la compréhension de notre travail.

1.1 Définitions et concepts de graphe

1.1.1 Graphe

Un graphe est noté par $G = (X(G), E(G))$, si aucune confusion n'est possible on note X au lieu de $X(G)$ et E au lieu de $E(G)$. Le graphe G est défini par deux ensembles finis :

$X = \{x_1, x_2, \dots, x_n\}$ dont les éléments sont appelés sommets avec $|X| = n$.

$E = \{e_1, e_2, \dots, e_m\}$ dont les éléments sont appelés arêtes avec $|E| = m$.

Une arête e de l'ensemble E est définie par une paire de sommets, appelés extrémités. Si l'arête e relie deux sommets x et y , on dira que ces sommets sont adjacents ou incidents avec e ou bien que l'arête e est incidente avec les sommets x et y .

- Le nombre des arêtes incidentes à un sommet est le nombre des arêtes sortants du sommet.
- Lorsque plusieurs arêtes relient deux sommets, nous les appelons des arêtes multiples.
- La taille d'un graphe $G = (X, E)$ est définie par le nombre d'arêtes de ce graphe.
- Nous appelons ordre d'un graphe $G = (X, E)$ le nombre de sommets de ce graphe qu'on note $O(G)$.
- On définit le voisinage d'un sommet x dans le graphe G comme l'ensemble de ses voisins, on le note par $N_G(x)$ tel que : $N_G(x) = \{y \in X / xy \in E\}$.

1.1.2 Boucle

Une boucle est une arête dont les deux extrémités sont identiques.

1.1.3 Graphe simple

Un graphe est simple s'il ne contient ni boucle ni arêtes multiples.

Le graphe $G = (X, E)$ donné sur la figure 1.1 (a) est un graphe simple ayant pour ensemble de sommets $X = \{x, u, v, w, y\}$ et pour ensemble d'arêtes $E = \{xu, uv, vw, wy, yx, yu\}$.

Le graphe $G = (X, E)$ est d'ordre $|X| = 5$ et de taille $|E| = 6$.

Le graphe $H = (X, E)$ donné sur la figure 1.1 (b) n'est pas un graphe simple, car il possède une boucle en x et une arête multiple uv .

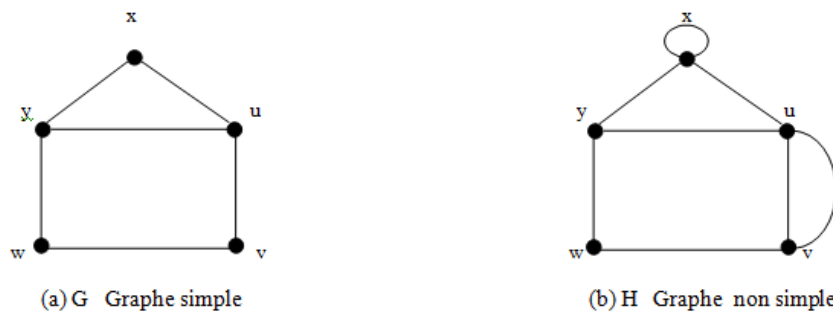


FIGURE 1.1 – Graphe simple et graphe non simple.

1.1.4 Graphe orienté

Un graphe orienté est un graphe dont chaque arc (e) tel que $e = (x_i; x_j) \in E$ possède une extrémité initiale $I(e) = x_i$ et une extrémité terminal $T(e) = x_j$.

Si $I(e) = T(e)$ on dit que e est une boucle.[1]

Le graphe de la figure 1.2 représente un graphe orienté possède six sommets et sept arêtes qui sont :

$X = \{1; 2; 3; 4; 5; 6\}$ et $E = \{e_1; e_2; e_3; e_4; e_5; e_6; e_7\}$, avec $I(e_1) = 2, T(e_1) = 1$.

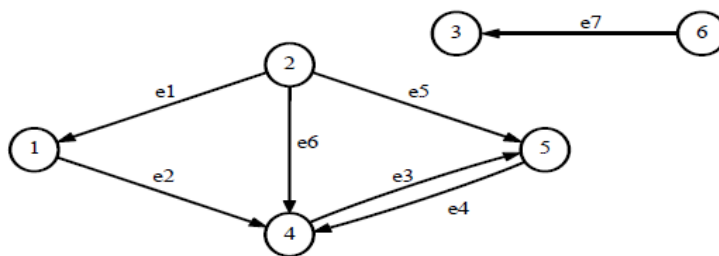


FIGURE 1.2 – Graphe orienté.

1.1.5 Graphe non orienté

Un graphe non orienté est un graphe dont les arêtes ne sont pas orientées et chaque arête (e) est définie comme suit : $e = (x_i; x_j) = x_i x_j$. Une arête de type $x_i x_i$ est appelée une boucle. Le graphe de la figure 1.3 représente un graphe non orienté.[1]

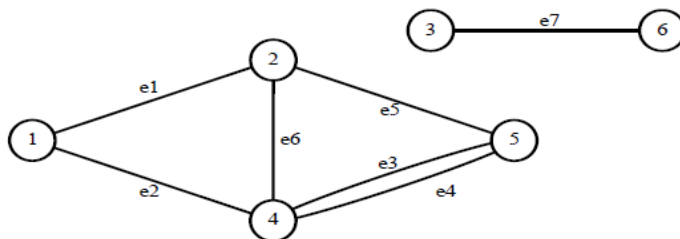


FIGURE 1.3 – Graphe non orienté.

1.1.6 Degrés

Soit $G = (X(G), E(G))$ un graphe.

Le degré d'un sommet x de G , noté $d_G(x)$ (ou $d(x)$), est égal au nombre de ses voisins, autrement dit $d_G(x) = |N_G(x)|$. Un sommet de degré k (respectivement au moins k , au plus k est noté k -sommet (respectivement k^+ -sommet, k^- -sommet).

Un sommet de degré 0 est dit isolé.

Proposition 1.1. (Lemme des poignées de mains) :

Pour tout graphe $G = (X(G), E(G))$ on a :

$$\sum_{x \in X(G)} d_G(x) = 2 | E(G) |$$

On peut déduire que dans un graphe le nombre de sommets de degré impair est pair.

- Le degré maximum d'un graphe G noté $\Delta(G)$ est le maximum des degrés des sommets de G .

En d'autres termes, $\Delta(G) = \max_{x \in X(G)} d(x)$.

- le degré minimum de G noté $\delta(G)$ est le minimum des degrés des sommets de G .

En d'autres termes, $\delta(G) = \min_{x \in X(G)} d(x)$.

1.1.7 Successeurs, Prédécesseurs d'un sommet

a) L'ensemble des successeurs d'un sommet x_i est l'ensemble de toutes les arcs sortants du sommet x_i noté par $\Gamma^+(x_i)$ tel que :

$$\Gamma^+(x_i) = \{x_j \in X / (x_i, x_j) \in E\}.$$

b) L'ensemble des prédécesseurs d'un sommet x_i est l'ensemble de toutes les arcs entrants au sommet x_i noté par $\Gamma^-(x_i)$ tel que :

$$\Gamma^-(x_i) = \{x_j \in X / (x_j, x_i) \in E\}.$$

c) x est un voisin de y si x est un prédécesseur ou un successeur de y . [1]

Dans le graphe orienté de la figure 1.4 on a : $\Gamma^-(4) = \{1, 2\}$ et $\Gamma^+(4) = \{3\}$.

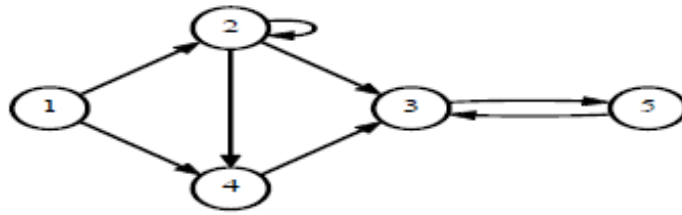


FIGURE 1.4 – Successeurs, prédécesseurs d'un sommet.

1.1.8 Graphe partiel

Soit $G = (X, E)$ un graphe. Le graphe $G' = (X, E')$ est un graphe partiel de G si E' est inclus dans E . Autrement dit, on obtient G' en enlevant une ou plusieurs arêtes au graphe G .

Le graphe de la figure 1.5 représente un graphe G et son graphe partiel.

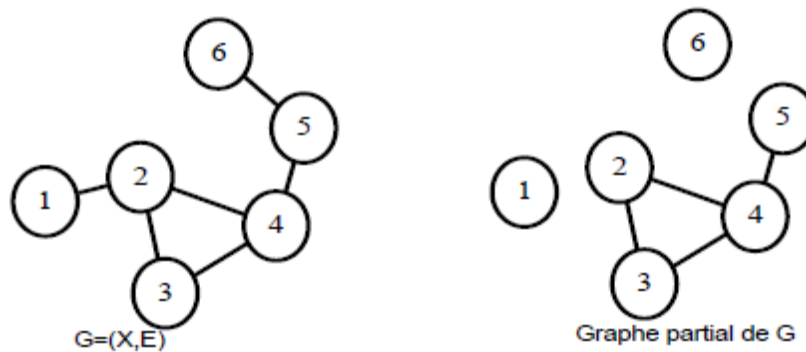


FIGURE 1.5 – Graphe G et l'un de ses graphes partiels.

1.1.9 Sous-graphe

Soit $G = (X, E)$ un graphe et A un sous ensemble de sommets inclus dans X .

Le sous-graphe $G_A = ((X_A), E(A))$ est un graphe qu'est formé de toutes les arêtes de G ayant les deux extrémités dans A .

Soit $G = (X, E)$ le graphe de la figure 1.6, alors $A = \{1, 2, 4\}$ et $E(A) = \{e_1, e_2, e_6\}$ est un sous graphe de G .

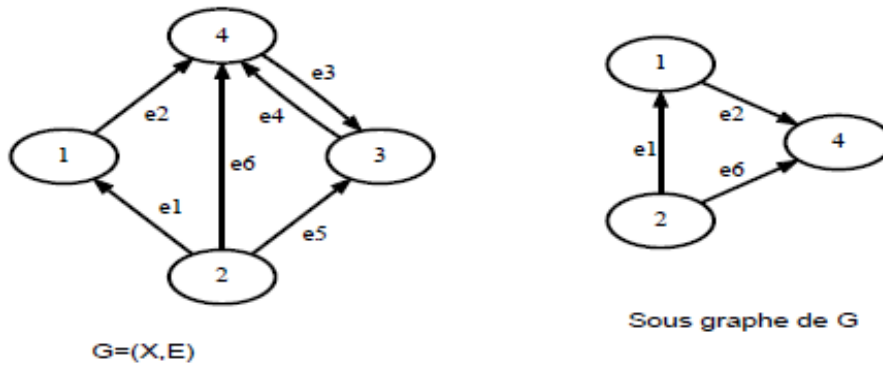


FIGURE 1.6 – Sous graphe d'un graphe G .

1.1.10 Sous-Graphe Partiel

En combinant ces deux dernières définitions, on peut définir un sous-graphe partiel. Il s'agit donc d'un graphe partiel d'un sous graphe. C'est concrètement un graphe au quel on a enlevé des sommets et leurs arêtes incidentes. Un sous-graphe partiel de $G = (X; E)$ est un graphe partiel d'un sous-graphe.

1.1.11 Une clique

Soit $G = (X, E)$ un graphe simple tel que X un ensemble de sommets de G deux à deux adjacents. Cet ensemble est dite une clique de n sommets est noté K_n . Donc un graphe G est une clique si $d_G(x) = n - 1$ pour tout ses sommets telle que $|X| = n$. [1]

La figure 1.7 représente des cliques de taille 3 et 5 respectivement.

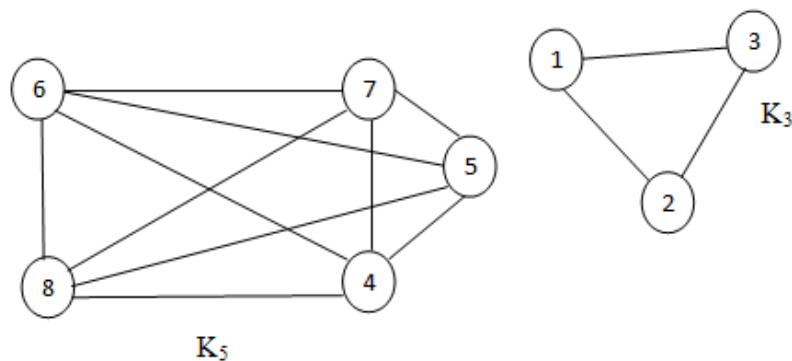


FIGURE 1.7 – Cliques.

1.2 chaînes, cycles, chemins et circuits

a) chaîne : Une chaîne joignant deux sommets x_0 et x_k dans un graphe G est une suite de sommets relié par des arêtes telle que deux sommets successifs ont une arête commune. On note la chaîne par (x_0, x_1, \dots, x_k) .

- Le premier et le dernier sommet sont appelés les extrémités de la chaîne.
- La longueur de la chaîne est égale au nombre d'arêtes qui la composent.
- Si aucune des sommets composant la séquence n'apparaît plus d'une fois, la chaîne est dite chaîne élémentaire.
- Si aucune des arêtes composant la séquence n'apparaît plus d'une fois, la chaîne est dite chaîne simple.
- Une chaîne eulérienne est une chaîne empruntant une fois seulement chaque arête de G .
- On appelle chaîne hamiltonienne une chaîne passant une fois seulement par chacun des sommets de G .

b) Cycle : Un cycle est une chaîne dont les extrémités coïncident.

- Un cycle élémentaire est un cycle minimal pour l'inclusion, c'est-à-dire ne contenant aucun autre cycle.
- Un cycle eulérien est une chaîne eulérienne dont les extrémités coïncident.
- Un cycle hamiltonien est un cycle qui passe une seule fois par chacun des sommets de G .

c) Un Chemin : Un chemin conduisant du sommet **a** au sommet **b** est une suite ayant pour éléments alternativement des sommets et des arêtes, commençant et se terminant par un sommet, telle que chaque arête est encadré à gauche par son sommet origine et à droite par son sommet destination.

- Si aucun des sommets composant la séquence n'apparaît plus d'une fois, le chemin est dit chemin élémentaire.
- Si aucun des arêtes composant la séquence n'apparaît plus d'une fois, le chemin est dit chemin simple.
- Un chemin eulérien est un chemin simple qui passe une et une seule fois par chaque arc du graphe.
- On appelle chemin hamiltonien un chemin passant une fois et une fois seulement par chacun des sommets de G .

d) Circuit : Un circuit est un chemin dont les extrémités coïncident (G orienté).

- Un circuit est dit élémentaire, si on ne rencontre pas deux fois le même sommet.

- Un circuit eulérien est un chemin eulérien dont les extrémités coïncident.
- Un circuit hamiltonien est un circuit qui passe une fois par chacun des sommets de G .

1.3 Connexité et forte connexité

1.3.1 Connexité

Un graphe $G = (X, E)$ est dit connexe si pour toute paire de sommet (x, y) , il existe une chaîne joignant x et y .

La figure 1.8 montre bien un graphe connexe.

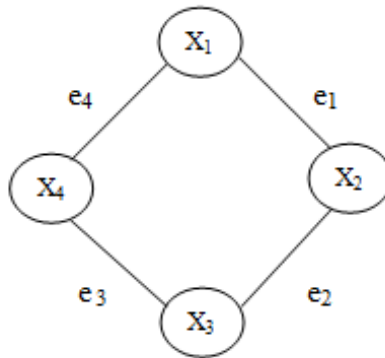


FIGURE 1.8 – Graphe connexe.

1.3.2 Forte connexité

Un graphe $G = (X, E)$ est dit fortement connexe si $\forall x, y \in X$, il existe un chemin de x vers y et de y à x . Une composante fortement connexe (cfc) est une sous-ensemble de sommets tel qu'il existe un chemin entre deux sommets quelconques. Une composante fortement connexe maximale (cfcmax) est un ensemble maximal de cfc. Les différentes cfcmax définissent une partition de X . Un graphe est fortement connexe s'il comporte une seule cfcmax.

1.4 Mode de représentation graphique

Un certain nombre de représentations existent pour d'écrire un graphe. En particulier, elles ne sont pas équivalentes du point de vue de l'efficacité des algorithmes. On distingue principalement

la représentation par matrice d'adjacence, par matrice d'incidence sommets-arcs (ou sommets-arêtes dans le cas non orienté) et par liste d'adjacence.

1.4.1 Matrice d'adjacence

Soit $G = (X, E)$ un graphe d'ordre n , avec $X = \{x_1, x_2, \dots, x_n\}$. On appelle matrice d'adjacence de G de taille $n \times n$ la matrice carrée $M = (m_{ij})_{1 \leq i, j \leq n}$ telle que :

$$m_{ij} = \begin{cases} 1 & \text{si } x_i x_j \in E \\ 0 & \text{sinon} \end{cases}$$

La matrice d'adjacence d'un graphe est symétrique, ayant pour éléments 0 et 1, avec des 0 sur la diagonale principale.

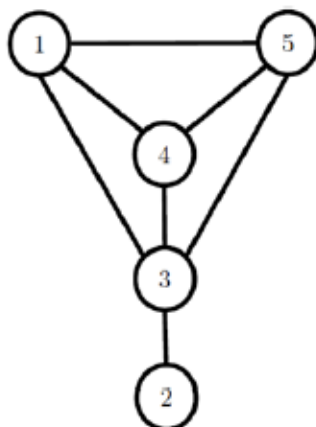


FIGURE 1.9 – Graphe G.

La matrice d'adjacence associée au graphe de la figure 1.9 est donné par :

$$M = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

1.4.2 Matrice d'incidence

a) Pour un graphe orienté

La matrice d'incidence (sommets-arcs) pour le graphe orienté est de taille $n \times m$, les sommets sont associés aux lignes, les arcs aux colonnes. Notons $M = (m_{ij})$ la matrice d'incidence associée au graphe $G = (X, E)$, pour pouvoir l'écrire, nous devons numéroter les arcs du graphe. On obtient alors :

$$m_{ij} = \begin{cases} 1 & \text{si le sommet } x \text{ est le sommet origine de l'arc } e. \\ -1 & \text{si le sommet } x \text{ est le sommet destination de l'arc } e. \\ 0 & \text{sinon} \end{cases}$$

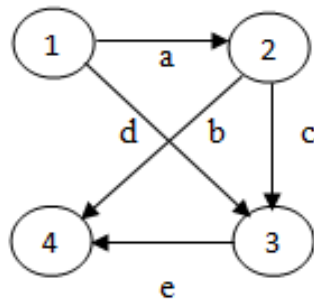


FIGURE 1.10 – Graphe orienté.

La matrice d'incidence associée au graphe de la figure 1.10 est donné par :

$$M = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 \\ 0 & -1 & -1 & 0 & 1 \\ 0 & 0 & 0 & -1 & -1 \end{pmatrix}$$

b) Pour un graphe non orienté

Un graphe non orienté sans boucle à n sommets numérotés et m arêtes numérotés peut être représenté par une matrice (n, m) d'entiers, l'élément m_{ij} vaut :

$$m_{ij} = \begin{cases} 1 & \text{si le sommet } x \text{ est une extrémité de l'arête } e. \\ 0 & \text{sinon} \end{cases}$$

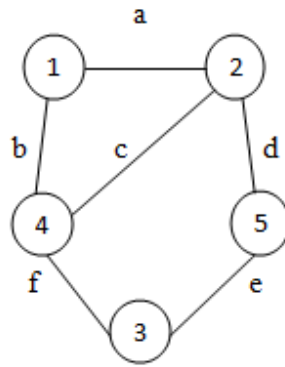


FIGURE 1.11 – Graphe non orienté.

La matrice d'incidence associée au graphe de la figure 1.11 est donné par :

$$M = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

1.4.3 Liste d'adjacence

Cette représentation consiste en un tableau de n (avec n le nombre de sommets du graphe) Listes chaînées. Pour chaque sommet x du graphe, on aura sa liste d'adjacence $liste(x)$ qui sera composée des sommets adjacents à x . Toujours avec notre graphe non orienté, on aura :

1.5.2 Graphe biparti et biparti complet :

Un graphe est dit biparti s'il existe une partition de son ensemble de sommets en deux sous-ensembles X_1 et X_2 telle que chaque arête ait une extrémité dans X_1 et l'autre dans X_2 .

- Un graphe G est dit biparti si et seulement s'il ne contient pas de cycle de longueur impaire.
- Un graphe biparti est dit complet si chaque sommet de X_1 est relié à chaque sommet de X_2 , on le note K_{pq} avec $p=|X_1|$ et $q=|X_2|$
- Un graphe biparti complet dont une partition contient un seul sommet est appelé étoile.

1.5.3 Graphe planaire

Un graphe G est dit planaire s'il peut être représenté sur le plan de manière que deux arêtes ne se rencontrent pas en dehors de leurs extrémités. (Voir la figure 1.14)



FIGURE 1.14 – Représentation planaire du K_4 .

Une face f de G est une région du plan bordée par un ensemble des arêtes de G . L'unique Face infinie est la face extérieure de G . Les sommets et les arêtes sur la frontière de f sont dit Incidentes à f . Deux faces différentes incidentes à la même arête sont dites adjacentes.

Le degré d'une face f , note $d_G(f)$ (ou $d(f)$), est le nombre des arêtes bordant f . L'ensemble Des faces d'un graphe planaire G est noté $F(G)$.

Le graphe de la figure 1. 14(b), a exactement 4 faces, elles sont toutes bordées par 3 arêtes, c'est-à-dire qu'elles sont toutes de degré 3 ($\forall f \in F(G); d(f) = 3$).

Si nous plaçons un point à l'intérieur de chaque face d'un graphe G dessine dans le plan et que pour chaque paire de faces adjacentes suivant une arête e , nous relient les points correspondants par une courbe traversant e , nous obtenons la représentation planaire d'un graphe appelé graphe dual G' de G .

Lemme 1.1. [3]

Pour tout graphe planaire $G = (X(G), E(G), F(G))$, on a :

$$\sum_{f \in F(G)} d_G(f) = 2 |E(G)|$$

De plus, il existe une formule simple établissant une relation entre le nombre des sommets, des arêtes et des faces d'un graphe planaire établie par Euler en 1752 et connue sous le nom de formule d'Euler

Théorème 1.2. Si G est un graphe planaire connexe, $|X(G)|$ son nombre des sommets, $|E(G)|$ son nombre des arêtes et $|F(G)|$ son nombre des faces, alors :

$$|X(G)| - |E(G)| + |F(G)| = 2$$

Etendue aux graphes non connexes, cette formule nous donne :

Corollaire 1.3. Pour tout graphe planaire G ayant k composantes connexes,

$$|X(G)| - |E(G)| + |F(G)| = k + 1$$

1.5.4 Graphe complémentaire

Le graphe complémentaire d'un graphe simple $G = (X(G), E(G))$ noté $\bar{G} = (X(G), E(\bar{G}))$ est un graphe ayant le même ensemble de sommets que G et $E(\bar{G})$ a exactement les paires de Sommets qui ne sont pas des arêtes de G .

La figure 1.15 représente un graphe G et son complémentaire \bar{G} . [4]

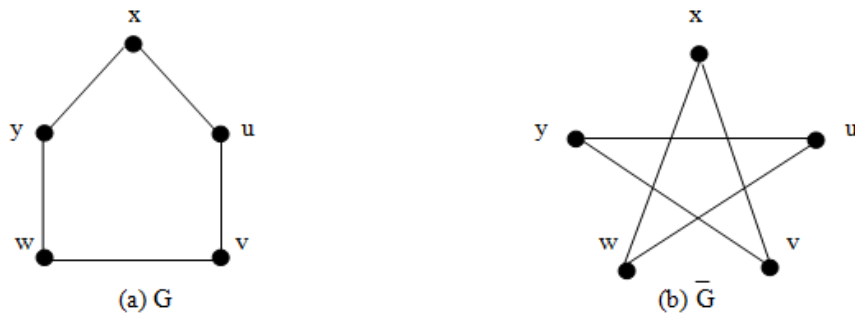


FIGURE 1.15 – Graphe G et son complémentaire.

1.5.5 Graphe triangulé et non triangulé

Une corde est une arête qui relie deux sommets non adjacents dans un cycle.

Un graphe G est triangulé si tout cycle dans G de longueur supérieure à 3 admet une corde.

Le graphe de la figure 1.16(a) n'est pas un graphe triangulé car il existe un cycle de longueur 4 qui n'admet pas de corde. Le graphe de la figure 1.16(b) est un graphe triangulé.



FIGURE 1.16 – Graphe triangulé, Graphe non triangulé.

1.5.6 Graphe représentatif des arêtes

Soit $G = (X, E)$ un graphe. Le graphe représentatif des arêtes de G (graphe adjoint) noté $L(G)$, est le graphe défini de la façon suivante :

Chaque sommet de $L(G)$ représente une arête de G et deux sommets de $L(G)$ sont adjacents si et seulement si les arêtes correspondantes ont une extrémité commune dans G (on dit alors qu'elles sont adjacentes).

La figure 1.17 représente un graphe G et son graphe représentatif des arêtes $L(G)$.



FIGURE 1.17 – Graphe et son graphe représentatif des arêtes.

1.5.7 Graphe K -régulier

Un graphe G est dit régulier de degré K , si tout ses sommets sont de degré K ,
 $(\forall x \in X, d_G(x) = K)$.

Le graphe ci-dessous est d'ordre 6 et il est 4-régulier.

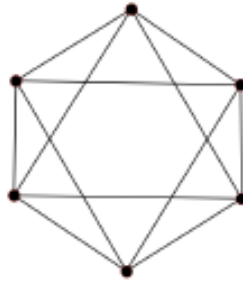


FIGURE 1.18 – Graphe 4-régulier.

1.5.8 Arbre et Arborescence

a) Arbres :

Définition

Un arbre est un graphe connexe sans cycle.

Soit G un arbre à n sommets $n > 2$. Les propriétés suivantes sont équivalentes :

- 1) G connexe et sans cycle
- 2) G sans cycle et possède $n - 1$ arêtes
- 3) G connexe et possède $n - 1$ arêtes
- 4) G sans cycle et si ajoutant une arête, on crée un et un seul cycle
- 5) G connexe et si on supprime une arête, il n'est plus connexe
- 6) Il existe une chaîne et une seule entre toutes paires de sommets.

Un graphe G vérifiant les une des propriétés ci-dessus est un arbre.

La figure 1.19 montre un arbre à 7 sommets

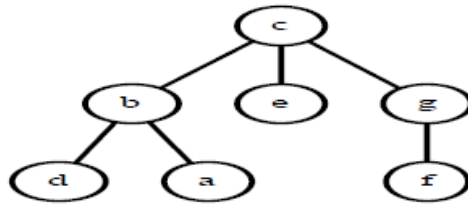


FIGURE 1.19 – Arbre.

b) Arborescence :

Une arborescence est un graphe orienté sans circuit admettant une racine $x_0 \in X$ telle que pour tout autre sommet $x_i \in X$, il existe un chemin unique allant de x_0 vers x_i .

Si l'arborescence comporte n sommets alors elle comporte exactement $(n - 1)$ arêtes.

Le graphe de la figure 1.20 est une arborescence.

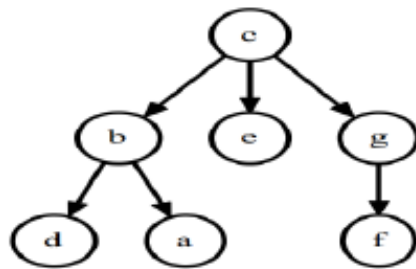


FIGURE 1.20 – Arborescence.

CHAPITRE 2

COLORATION DES SOMMETS DANS UN GRAPHE

2.1 Introduction

On va traiter dans ce chapitre le problème de la coloration des sommets. Dans la première partie de ce chapitre, on va commencer par donner quelques définitions en suite traiter un exemple sur la colration des sommets.La deuxième partie est consacré pour donner bien évidemment les différents théorèmes et propriétiés relatifs au problème et nous reprenons l'étude de la compléxité du problème de coloration des sommets d'un graphe. Mais avant ces deux parties, on va parler un peu de l'histoire et les domaines d'applications de la coloration des sommets des graphes.

2.2 Historique

L'origine du problème de coloration remonte au *XIX^{ème}* siècle par l'un des problèmes les plus célèbres et les plus productifs qui est le problème des quatre couleurs. Ce problème est resté pendant plus d'un siècle sans solution. Le problème des quatre couleurs consistait à trancher la question suivante : peut-on colorer n'importe quelle carte géographique de sorte que deux régions voisines n'aient pas la même couleur en n'utilisant que quatre couleurs au maximum? Les premiers résultats de coloration de graphes concernent presque exclusivement les graphes planaires, il s'agissait alors de colorer des cartes.

En 1852, le jeune mathématicien F.Guthrie s'est demandé s'il est toujours possible de colorer une carte de géographie à l'aide de quatre couleurs de sorte à ce que des pays voisins aient des couleurs différentes. La première trace écrite de la conjecture des quatre couleurs apparait dans une lettre

que A. D. Morgan envoya à W.R. Hamilton en 1852. P. J. Heawood montra quant à lui le théorème des cinq couleurs en reprenant des idées de A. Kempe. De nombreux travaux ont été publiés lors du siècle suivant pour réduire le nombre de couleurs à quatre, jusqu'à la démonstration finale de deux chercheurs américains, K. Appel et W. Haken en 1976. Ils ont pu répondre affirmativement à cette conjecture des quatre couleurs.

2.3 Domaines d'applications

Nous avons vu que la coloration d'un graphe est utile dans la coloration des cartes géographiques mais ce n'est pas là son unique domaine d'application. En effet la coloration des sommets et des arêtes d'un graphe ont de nombreuses utilités :

- Allocation de fréquences, par exemple dans un réseau de téléphone mobile GSM
- Confection d'horaires
- Problèmes d'incompatibilité : Le stockage de produits chimiques qui peuvent exploser s'ils entrent en contact, désignation d'un endroit pour des personnes ou des animaux en tenant compte des relations
- Problème du Sudoku : ce jeu est ni plus ni moins qu'un problème de coloration avec $k = 9$ couleurs
- Ordonnancement
- Emplois du temps
- Calcul de dérivées, de matrices jacobiniennes et hessiennes
- Gestion du trafic aérien
- Gestion de chaînes logistiques.

2.4 Coloration simple des sommets

Actuellement, les problèmes de coloration qui ont été étudiés le plus souvent sont ceux qui s'intéressent à la coloration de sommets.

Définition 2.1. Colorer un graphe, c'est associer une couleur à chaque sommet de façon que deux sommets adjacents soient colorés avec des couleurs différentes.

Plus formellement, une coloration d'un graphe $G = (X, E)$ est une application $c : X \rightarrow S$, où les éléments de S sont appelés les couleurs. Une coloration c est propre (dite simple, basique ou classique) si pour toute arête $xy \in E(G)$, $c(x) \neq c(y)$. Comme la taille de S est le paramètre im-

portant, on considère souvent que $S = \{1, 2, \dots, k\}$.

Une k -coloration d'un graphe G est une coloration propre de G à valeurs dans $S = \{1, 2, \dots, k\}$.

Un graphe est k -colorable s'il admet une k -coloration. Le plus petit entier k tel que G soit k -colorable est le nombre chromatique de G , noté $\chi(G)$. [19]

La figure 2.1 montre un graphe 4-coloriable.

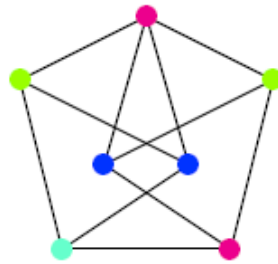


FIGURE 2.1 – Coloration des sommets d'un graphe G .

Remarques

- La coloration d'un graphe n'est bien évidemment pas unique, d'une part les couleurs peuvent être changées (remplacer le noir par du rouge par exemple), d'autre part la répartition des couleurs peut varier, ainsi que le nombre de couleurs utilisées.
- Un graphe d'ordre n pourra toujours être coloré de n couleurs différentes.

Théorème 2.1. [4][5] Si H est un sous-graphe de G alors $\chi(H) \leq \chi(G)$.

Proposition 2.2. $\chi(G) = \max\{\chi(C) \text{ tq } C \text{ composante connexe de } G\}$

Preuve

D'après la proposition précédente, $\chi(G) \geq \max\{\chi(C), C \text{ composante connexe de } G\}$.

Soit C_1, C_2, \dots, C_k les composantes connexes de G . Pour $1 \leq i \leq k$, soit c_i une coloration propre de C_i avec les couleurs $1, 2, \dots, \chi(C_i)$. Considérons c définie par $c(x) = c_i(x)$ si $x \in C_i$. Comme il n'y a pas d'arêtes entre deux sommets de composantes connexes différentes, c est une coloration propre de G . Ainsi $\chi(G) \leq \max\{\chi(C), C \text{ composante connexe de } G\}$.

2.4.1 Ensemble stable

Soit $G = (X, E)$ un graphe non orienté. Un sous ensemble $S \subset X$ est un ensemble stable s'il ne comprend que des sommets non adjacents deux à deux, en d'autre terme :

$$\forall i, j \in S \implies (i, j) \notin E$$

Le cardinal du plus grand ensemble stable est appelé le nombre de stabilité de G , on le note $\alpha(G)$, tel que :

$$\alpha(G) = \max\{\text{card}(S), \text{ tel que } S \text{ est stable}\}$$

On peut définir aussi une k -coloration comme une partition de l'ensemble des sommets en k parties stables.

La figure 2.2 représente un graphe avec 3 parties stables colorées.

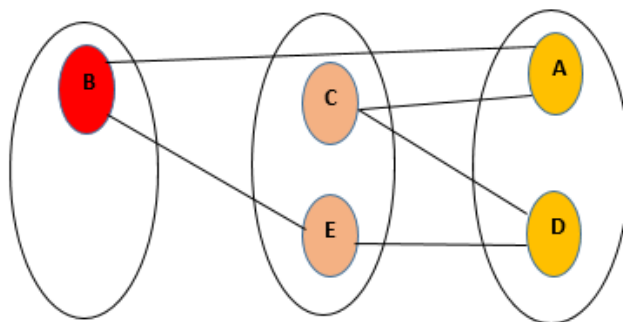


FIGURE 2.2 – Graphe 3-colorable : classes de couleurs .

2.4.2 Nombre Chromatique

Un graphe G est k -colorable s'il existe une coloration de G à partir d'un ensemble de k couleurs. L'entier minimum positif k pour lequel G est k -colorable représente le nombre chromatique de G noté $\chi(G)$.

En effet, un graphe G est k -colorable si et seulement si $\chi(G) \leq k$. Certainement, tout graphe d'ordre n est n -colorable. Nécessairement, si une k -coloration d'un k -chromatique graphe G est donnée, alors toutes les couleurs k doivent être utilisées.

Deux colorations différentes d'un graphe G sont présentées dans la Figure 2.3, la coloration de la Figure 2.3 (a) est une 5-coloration et celle de la Figure 2.3(b) est une 4-coloration. Comme

l'ordre de G est 10, le graphe G est k -colorable pour tout entier k avec $4 \leq k \leq 10$. Comme G est 4-colorable, alors $\chi(G) \leq 4$. Il n'existe pas une 3-coloration de G comme nous allons expliquer. Pour tout état de coloration c de G , les sommets a , b et c doivent être colorés différemment car ils induisent un triangle. Par conséquent, nous pouvons supposer que a est coloré par 1, b est coloré par 2 et c est coloré par 3 comme indiqué dans la figure 2.3(b). Si nous essayons de colorer G avec seulement trois couleurs 1, 2 et 3 alors nous sommes obligé de colorer les sommets de G comme suit :

$$\begin{aligned} c(a) = 1, c(b) = 2, c(c) = 3, c(d) = 1, c(e) = 3, \\ c(f) = 2, c(g) = 1, c(h) = 2, c(i) = 3, c(j) = 4. \end{aligned}$$

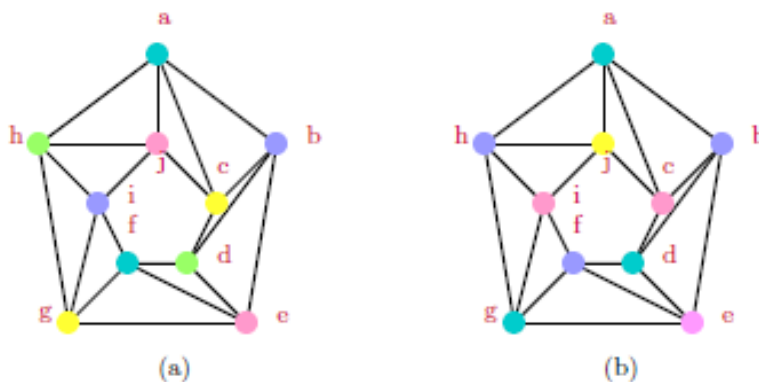


FIGURE 2.3 – Coloration de graphe G .

Cependant, les sommets j et h sont adjacents à la fois de la couleur 2, ce qui est impossible. Ainsi, G n'admet pas une 3-coloration. Par conséquent, $\chi(G) \geq 4$, et par suite $\chi(G) = 4$.

En général, pour montrer qu'un graphe G possèdent un nombre chromatique k .

Par exemple, nous avons besoin de montrer qu'il existe une k -coloration de G (et donc $\chi(G) \leq k$) et de montrer que chaque coloration de G nécessite au moins k couleurs (et donc $\chi(G) \geq k$).

2.4.3 Détermination du nombre chromatique

La détermination du nombre chromatique $\chi(G)$ qui désigne le nombre minimum de couleurs d'un graphe G , ainsi que l'obtention d'une coloration minimale des sommets de G reste un problème assez complexe. En effet on procède de la manière suivante :

- Enumérer tous les ensembles stables maximaux de G ;
- Chercher un recouvrement des sommets formé d'un nombre minimum d'ensemble stables maximaux ;

– Déduire de recouvrement une partition minimum de X en sous ensemble stables maximaux, et donc une coloration minimale de G . Cependant le nombre d'ensemble stables maximaux sera souvent trop important par ce que cette procédure soit acceptable, il est utile de recevoir à des algorithmes de coloration heuristiques simples et qui menant à une coloration des sommets de G non nécessairement minimal.

2.4.4 Bornes pour le nombre chromatique

Le nombre chromatique est généralement inconnu, c'est pour cette raison que la borne inférieure $\chi^-(G)$ et la borne supérieure $\chi^+(G)$ peuvent être établis pour $\chi(G)$:

$$\chi^-(G) \leq \chi(G) \leq \chi^+(G)$$

Proposition 2.3. Si $G = (X, E)$ est un graphe simple non orienté tel que : $n = |X|$, $m = |E|$

– $\alpha(G)$: la taille d'un stable maximum dans G .

– k : la k -coloration de G .

– $\Delta(G)$: le degré maximum dans G .

– $\delta(G)$: le degré minimum dans G .

– $\omega(G)$: la taille d'une clique maximum dans G .

Alors, le nombre chromatique du G est compris entre $\chi^-(G)$ et $\chi^+(G)$ avec :

$$- \chi^+(G) = \min\{n + 1 - \alpha(G), k, \Delta(G) + 1\}.$$

$$- \chi^-(G) = \max\left\{\frac{n}{n - \delta(G)}, \omega(G), \frac{n^2}{n^2 - 2m}, \frac{n}{\alpha(G)}\right\}$$

Corollaire 2.4. un graphe G d'ordre n a un nombre chromatique n si et seulement si $G = K_n$.

Puisque les couleurs distinctes sont nécessaires pour colorer deux sommets adjacents d'un graphe.

Corollaire 2.5. Un graphe G d'ordre n a un nombre chromatique 1 si et seulement si $G = \overline{K_n}$.

Théorème 2.6. (Gaddum, Nordhaus 1960) [6] Soit G le graphe complémentaire du graphe simple G , alors

$$\chi(G) + \chi(\overline{G}) \leq n + 1$$

2.4.5 Encadrement du nombre chromatique

Majoration

- $\chi(G) \leq \Delta(G) + 1$, ou $\Delta(G)$ est le plus grand degré des sommets de graphe G .

Minoration

Le nombre chromatique du graphe est supérieur ou égal à l'ordre de sa plus grande clique, on a alors :

- $\chi(G) \geq \omega(G)$ ou $\omega(G)$ est la taille maximum des cliques du graphe G .

Les notions suivantes donnent des informations sur le nombre chromatique :

- les sommets d'une même clique doivent être coloriés d'une couleur différente, ainsi trouver une clique à k sommets permet d'affirmer que $\chi(G) \geq k$;
- les sommets d'un même stable peuvent être coloriés de la même couleur.

Théorème 2.7. [5]/[6] Soit $G = (X, E)$ un graphe simple d'ordre n , On a l'encadrement suivant :

$$\left(\frac{n}{\alpha(G)}\right) \leq \chi(G) \leq \Delta(G) + 1$$

où $\Delta(G)$ est le degré maximal des sommets du graphe G et $\alpha(G)$ est le cardinal de la plus grande partie stable (le nombre de stabilité de G).

Démonstration

Notons $\{S_1, S_2, \dots, S_k\}$ une partition de X en k parties stables avec $k = \chi(G)$, Alors :

$$n = \sum_{i=1}^k |S_i| \leq \sum_{i=1}^k \alpha(G) = k \cdot \alpha(G) = \chi(G) \cdot \alpha(G)$$

$n, \alpha(G)$ et $\chi(G)$ étant des entiers naturels, l'inégalité de gauche est établie.

Construisons une partition de X en sous-ensembles stables de la manière suivante :

- On considère un sommet x_1 arbitraire de X et S_1 est une plus grande partie stable de X contenant x_1 .
- S'il existe un sommet x_2 de X qui n'est pas dans S_1 , on construit S_2 , une plus grande partie stable de X contenant x_2 disjointe de S_1 .
- S'il existe un sommet x_3 de X qui n'est pas dans $S_1 \cup S_2$, on construit S_3 plus grande partie stable de X contenant x_3 tel que S_1, S_2 et S_3 soient deux à deux disjointes.
- X étant un ensemble fini, ce procédé se terminera et nous obtenons une partition $\{S_1, S_2, \dots, S_k\}$ de X ; En choisissant une couleur par élément de la partition, nous aurons nécessairement : $\chi(G) \leq k$.

• Considérons à présent un sommet x de la partie S_k , Le caractère maximal des parties construites assure que ce sommet x est adjacent à au moins un sommet de chaque partie $S_i, i \in \{1, 2, \dots, k-1\}$, On en déduit alors que $d(v) \geq k-1$ d'où $\Delta(G) \geq d(v) \geq k-1 \geq \chi(G) - 1$ ce qui établit la deuxième inégalité.

Proposition 2.8. Soit $G = (X, E)$ un graphe simple d'ordre n , alors :

$$\chi(G) + \alpha(G) \leq n + 1$$

Démonstration

Considérons S , une partie stable de X de cardinal $\alpha(G)$. Une coloration possible des sommets consiste à colorier les sommets de S d'une même couleur et les $n - \alpha(G)$ autres sommets de couleurs toutes différentes. On en déduit que : $\chi(G) \leq 1 + (n - \alpha(G))$

Corollaire 2.9. Soit un graphe $G = (X, E)$.

$$\chi(G) = 1 \Leftrightarrow G \text{ est stable (n'a pas d'arêtes)}$$

$$\chi(G) = 2 \Leftrightarrow G \text{ est biparti}$$

Proposition 2.10. Soit G un graphe, si G est un cycle impair ou un graphe complet alors :

$$\chi(G) = \Delta(G) + 1$$

.

Démonstration

Si G est un cycle élémentaire de longueur impaire, alors on a $\chi(G) > 2$.

Dans un cycle élémentaire, tous les sommets sont de degré 2, on sait que $\chi(G) \leq \Delta(G) + 1 = 3$.

On en conclut que $\chi(G) = 3 = 2 + 1 = \Delta(G) + 1$.

Si G est un graphe complet alors tous ses sommets doivent être coloriés différemment et

$\Delta(G) = |X| - 1$, donc $\chi(G) = |V| = \Delta(G) + 1$.

Proposition 2.11. Soit G un graphe simple, on a :

$$\omega(G) \leq \chi(G) \leq \Delta(G) + 1.$$

Proposition 2.12. Soit G un graphe simple, on a :

$$\chi(G) \leq 1 + \max(\delta(G')) \leq 1 + \Delta(G)$$

où G' graphe induit de G et $\delta(G')$ est le degré minimum des sommets de G' .

Corollaire 2.13. Soit G un graphe simple connexe qui n'est pas régulier alors :

$$\chi(G) \leq \Delta(G)$$

Démonstration :

Supposons que $\chi(G) > \Delta(G)$, c'est à dire $\chi(G) = \Delta(G) + 1$. D'après la proposition précédente il existe un sous graphe induit G' tel que $\delta(G') = \Delta(G)$. Ainsi G' est $\Delta(G)$ -régulier. Ce graphe ne peut pas être connecté à un autre sommet de G car tout les sommet de G' sont de degré $\Delta(G)$. Comme G est connexe, on a $G = G'$ donc G est régulier.

Une question intéressante est de savoir quel graphe ne vérifie pas l'inégalité du corollaire précédent. Le théorème de Brooks montre que les exceptions sont des graphes particuliers.

Théorème 2.14. (Brooks,1941) [5][6][7] Soit G un graphe connexe tel que $\Delta(G) \geq 3$ et G ne contient pas de clique à $\Delta(G) + 1$ sommets. Alors $\chi(G) \leq \Delta(G)$.

2.4.6 Nombre chromatique de graphes classique

Il est facile de déterminer le nombre chromatique de certains graphes classiques :

- Graphe isolé d'ordre n : $\chi(I_n) = 1$.
- Graphe cyclique d'ordre n : $\chi(C_n) = 2$ si n pair et 3 si n impair.
- Graphe complet d'ordre n : $\chi(K_n) = n$.
- Un graphe G biparti avec au moins une arête : $\chi(G) = 2$ (en fait un graphe est 2- coloriable si et seulement s'il est biparti).
- G arbre avec au moins une arête : $\chi(G) = 2$.

2.4.7 Complexité

Décider si un graphe possède une coloration en k couleurs est un problème NP-complet, et trouver le nombre chromatique $\chi(G)$ d'un graphe est un problème NP-difficile. Le problème du nombre chromatique devient polynomial si les sommets du graphe ont un degré d'au plus 2 ou si le graphe est biparti.

2.4.8 Problème de coloration de sommets dans un graphe

Depuis plusieurs décennies, la coloration des graphes est un domaine très attractif de la théorie des graphes par ses nombreuses applications. La coloration de graphe permet de modéliser de nombreux problèmes réels, depuis le placement de personnes autour d'une table ou de pièces sur un échiquier jusqu'aux différents problèmes d'ordonnancement et de planning de la vie de tous les jours et dans le domaine des réseaux.

Problème de planification d'examens

Dans une université l'administration doit organiser les horaires des examens. On suppose qu'il y a 7 épreuves à planifier, correspondant aux cours numérotés de 1 à 7 et il y a des étudiants qui rentrent dans deux épreuves différentes, le tableau suivant résume les paires de cours qui ont des étudiants en communs.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 | | × | × | × | | | |
| 2 | × | | × | × | | × | |
| 3 | × | × | | × | | × | × |
| 4 | × | × | × | | × | × | |
| 5 | | × | | × | | × | × |
| 6 | | | × | × | × | | × |
| 7 | × | × | × | | × | × | |

Pour répondre à la question suivante :

Comment organiser ces épreuves de façon qu'aucun étudiant n'ait à passer deux épreuves en même temps et cela sur une durée minimale ?

Nous considérons le graphe G de la figure 2.4 dont les sommets sont les épreuves qui sont numérotées de 1 à 7, une arête relie deux de ses sommets lorsque les deux cours correspondant possèdent des étudiants en communs.

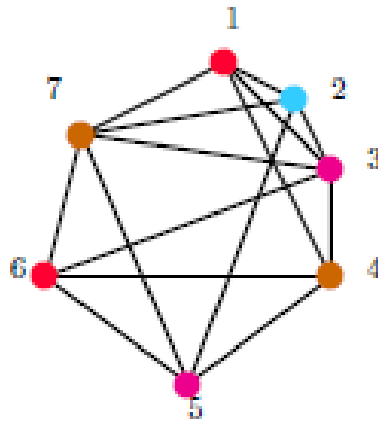


FIGURE 2.4 – Le graphe qui modélise le problème de planification d’examens .

Planifier les examens en un temps minimal consiste à déterminer une k -coloration G , avec $k = \chi(G)$.

Le graphe G possède un sous-graphe complet d’ordre 4 (de sommets 1, 2, 3, 7), donc $\chi(G) > 4$.

Déterminons une partition des sommets de G en sous-ensembles stables :

$$S_1 = \{1, 6\}.$$

$$S_2 = \{2\}.$$

$$S_3 = \{3, 5\}.$$

$$S_4 = \{4, 7\}.$$

D’où $\chi(G) \leq 4$ et finalement $\chi(G) = 4$. Les examens peuvent être répartis en 4 périodes de la manière suivante :

- * 1^{re} période, épreuves des cours 1 et 6.
- * 2^{ème} période, épreuve du cours 2.
- * 3^{ème} période, épreuves des cours 3 et 5.
- * 4^{ème} période, épreuves des cours 4 et 7.

2.5 Quelques types de coloration de sommets

2.5.1 Coloration par listes des sommets

Dans la plupart des problèmes pratiques de coloration, il y a des restrictions sur les couleurs qui peuvent être attribuées à certains sommets. Ceci amène à la notion de coloration par listes.

La coloration par listes des sommets d'un graphe est une version plus générale de la coloration propre des sommets, on affecte toujours une couleur aux sommets du graphe, mais celle-ci est choisie dans une liste de couleurs associée au sommet. La notion de la coloration par liste a été introduite par Vizing en 1976 et Erdős, Rubin et Taylor en 1979.

Soit G un graphe et L une fonction qui attribue à chaque sommet x de G un ensemble $L(x)$ d'entiers strictement positifs appelée la liste de x . Une coloration $c : X \rightarrow N$ tel que $c(x) \in L(x)$ pour tout $x \in X$ est appelée une coloration par listes de G selon L , ou L -coloration et nous disons que G est L -colorable. Observons que si $L(x) = \{1, 2, \dots, k\}$ pour tout $x \in X$, une L -coloration est simplement une k -coloration. Par exemple, si G est un graphe biparti et $L(x) = \{1, 2\}$ pour tous les sommets x , alors G admet la L -coloration qui attribue la couleur 1 à tous les sommets d'une partie et la couleur 2 à tous les sommets de l'autre partie. Observons également que donner une liste de longueur un à un sommet revient à précolorer ce sommet avec la couleur de cette liste. Comme pour la coloration, la notion de coloration par listes s'étend directement aux digraphes.

Au premier abord, on pourrait croire que si on prend un graphe k -chromatique et qu'on attribue à chacun de ses sommets une liste $L(x)$ d'au moins k couleurs, alors il admet une L -coloration, mais ce n'est pas toujours le cas.

On peut vérifier que le graphe biparti représenté sur la figure 2.5 n'admet pas de coloration par listes pour les listes qui sont indiquées. En revanche, quelles que soient des listes de longueur 3 que l'on attribue aux sommets de ce graphe, il y aura une coloration par listes compatible.

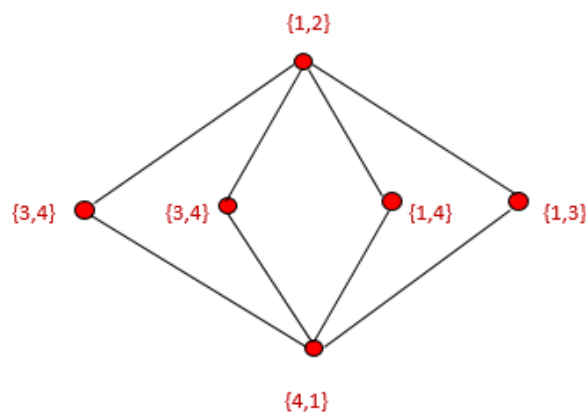


FIGURE 2.5 – Un biparti qui n'est pas 2-liste coloration. .

2.5.1.1 Le nombre chromatique par liste

Le nombre chromatique par liste d'un graphe G est noté $\chi_l(G)$, est définie comme étant le plus petit entier k tel que pour toute k -affectation L sur le graphe G , il existe une L -coloration de G . Une borne supérieure du nombre de liste chromatique est donnée par $\chi_l(G) \geq \chi(G)$.

2.5.1.2 La choisissabilité

Un graphe G ou digraphe D est dit k -choisissable ou (k -liste-colorable) s'il a une coloration par listes à chaque fois que toutes les listes sont de longueur k . Tout graphe G est clairement n -choisissable. Le plus petit entier k pour lequel G est k -choisissable est appelé choisissabilité de G est noté $ch(G)$. Par exemple, la choisissabilité du graphe de la Figure 2.6 vaut trois, alors que son nombre chromatique vaut deux.

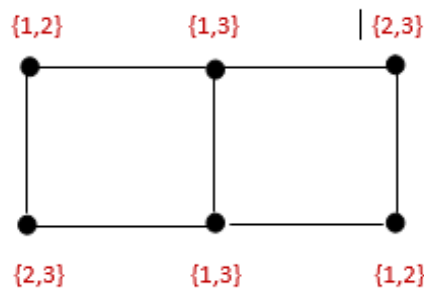


FIGURE 2.6 – Un graphe biparti de choisissabilité trois .

Théorème 2.15. [5]/[8] Pour tout graphe G nous avons $\chi(G) \leq \chi_l(G) \leq \Delta(G) + 1$.

Théorème 2.16. [5] Chaque cycle pair est 2-choisissabilité.

Corollaire 2.17. [3] Pour chaque entier n paire, $n \geq 4$ nous avons $\chi_l(C_n) = 2$.

Théorème 2.18. [8] Chaque graphe planaire biparti est 3-choisissabilité.

Théorème 2.19. [8] Chaque graphe planaire de maille 5 est 3-choisissabilité.

2.5.2 Coloration forte

Coloration X_s dominance

La coloration de graphes et les problèmes de dominance sont souvent en relation. Chellali et Volkmann montrent certaines relations entre le nombre chromatique et autres paramètres de

dominance dans les graphes. Récemment, une nouvelle coloration d'un graphe induisant une relation de dominance entre les sommets du graphe et les classes de couleur a été proposée. Cette coloration est appelée coloration forte.

On dit qu'un sommet x domine une classe de couleur si et seulement si x est adjacent à tous les sommets de cette classe (voir la Figure 2.7).

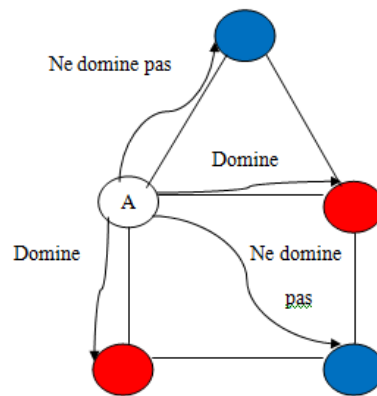


FIGURE 2.7 – Dominance X_s coloration .

Définition 2.2. Une k -coloration $\{C_1, C_2, \dots, C_k\}$ d'un graphe G est dite forte si pour tout sommet $x \in X$, il existe au moins un indice $i \in \{1, 2, \dots, k\}$ tel que x est adjacent à tous les sommets de C_i . Autrement dit, pour tout $x \in X$, x domine au moins une classe de couleur C_i tel que $i \in \{1, 2, \dots, k\}$.

Un graphe est fortement k -colorable s'il a une k -coloration forte. Le nombre chromatique d'une coloration forte est noté $\chi_s(G)$ du graphe G est le plus petit entier k pour lequel G est fortement k -colorable.

2.5.3 Coloration forte stricte

La coloration forte stricte est un nouveau paramètre de coloration de graphes qui s'inspire de la coloration forte.

Telle que déjà définie, la coloration forte admet la présence d'une classe de couleur vide qui permet à tous les sommets du graphe de dominer une couleur. Pratiquement, cette classe de couleur vide fait perdre la dominance aux sommets. Pour cette raison, une nouvelle variante de coloration a été définie récemment appelée : Coloration Forte Stricte (en anglais : Strict Strong Coloring). Cette dernière est une coloration forte qui n'admet pas de classe de couleur vide. Elle est très utile pour de nombreuses applications comme la diffusion, la dissémination de

données,... etc.

2.5.4 Coloration K-équitable

Soit $G = (X, E)$ un graphe donné. Une coloration K-équitable du graphe G est une coloration des sommets du G ou le nombre de sommets coloriés par la couleur c_i est égale au nombre de sommets coloriés par la couleur $c_j \forall (i, j) \in 1, \dots, k$.

La Figure 2.8 représente une coloration K-équitable dun graphe à 6 sommets.

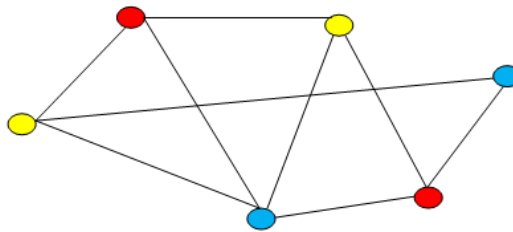


FIGURE 2.8 – Coloration 3-équitable .

2.5.5 Coloration avec satisfaction de contraintes CSP

Définition 2.3. Un CSP (Constraint Satisfaction Problems) est défini comme étant un ensemble de contraintes impliquant un ensemble de variables, dont chacune est définie sur son domaine propre. L'objectif consiste à trouver un ensemble de valeurs choisies dans les domaines susmentionnés à affecter à ces variables de sorte que toutes les contraintes soient satisfaites.

2.5.5.1 Concepts de base d'un CSP :

Plus formellement, un problème de satisfaction de contraintes est défini par le quadruplet (X, D, C, R) tel que :

- $X = (x_1, x_2, \dots, x_n)$ est l'ensemble des n variables du problème ;
- $D = (d_1, d_2, \dots, d_n)$ est un ensemble de n domaines finis dont chacun est associé à une variable de X . C'est à dire le domaine d_i est associé à la variable x_i (leurs valeurs possibles) ;
- $C = \{c_1, c_2, \dots, c_m\}$ un ensemble fini de contraintes. Chaque contrainte c_i est définie par un uplet $(x_{i1}, x_{i2}, \dots, x_{ik})$ de k variables de X liées par c_i . On nomme $\text{scope}(c_i)$ l'ensemble $(x_{i1}, x_{i2}, \dots, x_{ik})$;
- $R = \{r_1, r_2, \dots, r_n\}$ est un ensemble fini de relations associées aux m contraintes de C . Chaque relation r_i représente un ensemble de n -uplets de valeurs autorisées par la contrainte c_i .

Définitions :

Pertinence : on dit qu'une variable x_i est pertinente pour la contrainte c_k , si c_k porte sur la variable x_i .

Arité de la contrainte : l'arité de la contrainte c_k est le nombre de variables pertinentes pour c_k .

CSP binaire : un CSP binaire est un CSP dont toutes les contraintes c_k ont une arité égale à 2. En d'autres termes, chaque contrainte a exactement 2 variables pertinentes. Notons que tout CSP n-aire peut être ramené à un CSP binaire équivalent.

Matrice de contraintes : une matrice de contraintes est une matrice Mat à m lignes (m étant le nombre de contraintes) et n colonnes (n étant le nombre de variables du CSP) telle que :

$$Mat[i,j] = \begin{cases} 1 & \text{si } x_i \text{ est pertinent pour } c_i \\ 0 & \text{sinon} \end{cases}$$

$$\begin{pmatrix} 1 & 1 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 1 \\ 0 & 0 & 1 & \dots & 1 \\ 1 & 0 & 0 & \dots & 1 \end{pmatrix}$$

Matrice de contraintes $Mat[i,j]$

Sur cet exemple, on peut lire : x_1 et x_2 sont pertinentes pour c_1 , x_2 et x_n sont pertinentes pour c_2 .

Graphe de contraintes : une contrainte relie les deux variables pour lesquelles elles sont pertinentes par une arête, les sommets du graphe étant les variables.

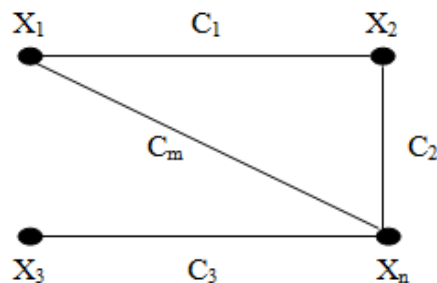


FIGURE 2.9 – Graphe de contraintes

Exemple d'un CSP (coloriage d'une carte)

L'objectif consiste à colorier une carte de sorte que deux régions ayant des frontières en commun soient coloriées avec des différentes couleurs.

Ce problème peut être modélisé par le CSP suivant :

- $X = \{x_1, x_2, \dots, x_6\}$ est l'ensemble des régions à colorier ;
- $D = \{d_1, d_2, \dots, d_6\}$ est l'ensemble des domaines de chaque variable telle que pour tout i , on a : d_i est l'ensemble des couleurs pouvant être attribuées à la variable x_i avec $d_i = \{rose, jaune, bleu\}$;
- $C = \{c_1, c_2, \dots, c_{10}\}$ est l'ensemble fini des contraintes où chaque c_i est définie par un couple (x_{i1}, x_{i2}) de variables de X représentant des régions voisines sur la carte avec $c_1 = (x_1, x_2), c_2 = (x_1, x_3), \dots, c_{10} = (x_5, x_6)$;
- $R = \{r_1, r_2, \dots, r_{10}\}$ est l'ensemble des valeurs autorisées respectivement par $\{c_1, c_2, \dots, c_{10}\}$.

Chaque relation r_i représente l'ensemble de couples de couleurs différentes. on a donc $\forall i, r_i = \{(rose, jaune), (rose, bleu), (jaune, rose), (jaune, bleu), (bleu, rose), (bleu, jaune)\}$.

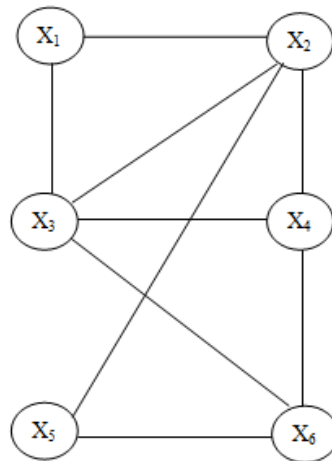


FIGURE 2.10 – Problème de coloriage de carte et son graphe des contraintes associé.

La figure 2.10 illustre cet exemple et sa représentation graphique. Les domaines qui sont les couleurs possibles de chaque variable n'apparaissant pas dans cette représentation graphique (il existe une représentation en microstructure qui permet de voir les valeurs des domaines des variables en explicitant les n-uplets autorisés par chacune des contraintes). Les contraintes sont représentées par les aretes du graphe, elles expriment la différence des valeurs affectées aux

variables.

Cette représentation ne fait pas apparaître le détail des domaines et des valeurs respectant chaque contrainte. Mais elle donne une idée générale de la structure du problème.

2.5.6 Coloration par sommes

La coloration par sommes consiste à attribuer un poids aux arêtes du graphe de façon à ce que pour chaque sommet, la somme des poids des arêtes incidentes à ce sommet soit différente de celle de ses voisins. Autrement dit, la somme des poids des arêtes incidentes à un sommet lui donne une couleur et le but est d'obtenir une coloration propre des sommets.

Ce type de coloration a été défini en 2004 par Karonski, Łuczak et Thomason qui ont émis la conjecture suivante :

Conjecture 2.1. [18] Pour tout graphe $G = (X, E)$ qui ne possédant pas d'arête isolée, il existe $K \in \{1, 2, 3\}$ pour lequel toutes les arêtes de G sont colorier par K et que les sommets adjacentes ne peuvent pas avoir le même poids.

La Figure 2.11 donne un exemple de coloration par sommes sur arêtes pour le cycle C_6 de six sommets. Sur le dessin, la somme des poids des arêtes incidentes à un sommet est inscrit dans le sommet.

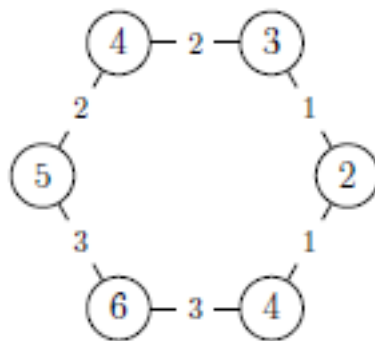


FIGURE 2.11 – Exemple de coloration par sommes..

Le graphe donné en exemple nécessite trois couleurs. Cela montre en particulier que si la Conjecture 2.1 est vraie, alors elle est optimale.

2.6 Coloration des graphes parfaits

Soit G un graphe, la taille d'une clique de taille maximum de G est notée $\omega(G)$. Il est clair que si un graphe G contient une clique de taille k , il faut au moins k couleurs pour la colorier. Donc, tout graphe G vérifie : $\omega(G) \leq \chi(G)$, il est naturel de se demander s'il y a égalité entre le nombre chromatique et la taille d'une plus grande clique. Ce n'est pas le cas et le plus petit contre-exemple est C_5 (le graphe C_k désigne un trou sur k sommets), $\chi(C_5) = 3$ et $\omega(C_5) = 2$. Plus généralement, n'importe quel trou impair vérifie $\chi(C_{2k+1}) = 3$ et $\omega(C_{2k+1}) = 2$. Ghouila-Houri [9] a également remarqué que n'importe quel antitrou impair vérifie $\chi(C_{2k+1}) = k + 1$ et $\omega(C_{2k+1}) = k$. Donc, l'égalité $\chi(G) = \omega(G)$ n'est pas vérifiée en général, mais il est intéressant de se demander pour quels graphes on a égalité. Dans un graphe G , une coloration de G et une clique de G sont dites de même taille si la taille de la clique est exactement égale au nombre de couleur de la coloration. Dans ce cas, la clique est de taille $\omega(G)$ et la coloration utilise $\chi(G)$ couleurs.

En 1960, Berge [10] a défini la classe des graphes parfaits comme étant la classe des graphes G tel que tout sous-graphe induit H de G vérifie $\chi(H) = \omega(H)$. Cette classe de graphes est apparue comme étant une classe de graphes assez générale sur laquelle le problème de la coloration pouvait être résolue efficacement. En 1984, Gröstchel, Lovász et Schrijver [11] ont donné un algorithme polynomial qui trouve pour tout graphe parfait, une coloration et une clique de même taille.

Au début des années 1960, Berge a énoncé deux conjectures célèbres. La première, dite conjecture faible des graphes parfaits a été étudiée par Fulkerson, puis finalement démontrée par Lovász en 1971. C'est ce que nous appelons maintenant le théorème des graphes parfaits :

Théorème 2.20. [12][13] *Un graphe est parfait si et seulement si son complémentaire est parfait.*

La deuxième conjecture de Berge dite conjecture forte des graphes parfaits a fait l'objet de nombreuses recherches avant d'être finalement démontrée par Chudnovsky, Robertson, Seymour et Thomasen 2002. C'est ce que nous appelons maintenant le théorème fort des graphes parfaits :

Théorème 2.21. [14] *Un graphe est parfait si et seulement s'il est de Berge.*

Le problème de la reconnaissance des graphes parfaits a lui aussi été résolu. En 2002, Chudnovsky, Cornuéols, Liu, Seymour et Vuskovic ont donné un algorithme de complexité $O(n^9)$ permettant de reconnaître les graphes parfaits.[15]

2.7 Coloration des graphes planaires

Pour plus d'un siècle, le problème le plus célèbre non résolu jusqu'au 1976 en théorie des graphes était le problème de coloration des cartes appelé "le problème des quatre couleurs". Ce problème qui traite de déterminer si une conjecture appelée quatre couleurs était vraie ou fausse, a eu un impact majeur sur le développement de la théorie des graphes. En fait, ils se sont posé la question suivante :

Les pays de chaque carte peuvent-ils être colorés avec quatre couleurs au moins, de telle sorte que chaque deux pays ayant une frontière commune soient colorés différemment ?

Les graphes planaires ont d'abord été étudiés en tant que dual d'une carte, parce que chaque graphe planaire simple à n -sommets a au moins $3n-6$ arêtes, un graphe planaire possède un sommet de degré au plus 5. Cela donne une preuve inductive que les graphes planaires sont 6-colorable.

En 1852, Guthrie [16] a conjecturé que n'importe quelle carte peut être coloriée avec au plus quatre couleurs. Ce résultat est prouvé en 1876 par Kempe. En 1878 Cayley a annoncé le problème au société mathématique de London, et Kempe a publié une "solution". Onze ans plus tard Heawood [17] a découvert une erreur dans la solution proposée par Kempe du problème des quatre couleurs et ni lui ni Kempe ont été en mesure de corriger l'erreur, Heawood a pu utiliser la technique de Kempe pour prouver que chaque carte peut être coloriée avec cinq couleurs ou moins. Cela a conduit au théorème suivant :

Théorème 2.22. Théorème des quatre couleurs (Appel, Haken, Koch,1976)[16] Tout graphe planaire $G = (X, E)$ est 4-coloriable. Ainsi, On peut colorer les sommets d'un graphe planaire (sans boucles) en utilisant au plus quatre couleurs de telle sorte que toutes les arêtes aient des extrémités de couleurs différentes.

Cette conjecture a été formulée pour la première fois par l'Écossais Francis Guthrie en 1852. Il était alors question de coloration de carte de géographie. La preuve de ce théorème n'arriva qu'en... 1976, grâce à Kenneth Appel et Wolfgang Haken. La démonstration fit grand bruit car ce fut le premier théorème de l'histoire des mathématiques qui a nécessité l'usage systématique de l'ordinateur.

Le graphe de la figure 2.14 présente un graphe planaire coloré avec 4-couleurs.



FIGURE 2.12 – Un graphe planaire coloré avec 4-couleurs.

Définition 2.4. Un graphe planaire est triangulé si toutes ses faces sont des triangles. On appelle triangulation d'un graphe planaire G tout graphe planaire triangulé T tel que $X(T) = X(G)$ et $E(G) \subset E(T)$.

Proposition 2.23. Tout graphe planaire admet un sommet de degré 5.

Preuve

Soit G un graphe planaire. On a $\sum_{x \in G} d(x) = 2 |E(G)| \leq 6 |X(G)| - 12$. Le degré minimum de G est au plus égale au degré moyen qui vaut $\frac{6|X(G)|-12}{|X(G)|} < 6$. Il y a donc un sommet de degré inférieur à 6.

Proposition 2.24. Tout graphe planaire est 6-colorable.

Preuve

Puisque tout sous-graphe d'un graphe planaire est planaire, tout sous-graphe possède un sommet de degré au plus 5. La dégénérescence d'un graphe planaire est donc au plus 5. La proposition 2.16 donne le résultat.

2.8 Conclusion

Dans ce chapitre, nous avons rappelé les notions fondamentales liées au problème de coloration des sommets de graphes, un des problèmes classiques de la théorie de graphes, ainsi que ses variantes et ses applications.

CHAPITRE 3

QUELQUES ALGORITHMES DE COLORATION

3.1 Introduction

La coloration des graphes quand l'ordre est petit est assez aisée, mais le problème se complique dès que le nombre des sommets augmente donc pour colorier un graphe, des algorithmes ont été développés.

Les algorithmes que nous allons présenter (il en existe d'autres) permet d'obtenir une assez bonne coloration d'un graphe, c'est-à-dire une coloration n'utilisant pas un trop grand nombre de couleurs.

3.2 Graphes quelconques : des algorithmes d'approximation

3.2.1 Coloration séquentielle : l'algorithme glouton

Une première approche pour colorer le graphe est de prendre ses sommets les uns après les autres afin de leur affecter une couleur, tout en veillant à ce que deux sommets adjacents n'aient jamais la même couleur : c'est l'algorithme de coloration séquentielle.

On parle d'algorithme glouton dans le sens où il consiste à faire le choix d'une couleur "optimum" localement dans l'espoir d'obtenir une coloration finale "satisfaisante".

On peut par exemple décider de prendre pour chaque sommet la plus petite couleur non utilisée dans les voisins du sommet à colorier.

L'algorithme fourni ainsi une borne supérieure pour $\chi(G)$, c'est-à-dire une coloration propre, mais pas forcément optimale. La coloration obtenue dépendra fortement de l'ordre dans lequel les sommets seront colorés.

Notons ainsi que pour un graphe donné il existe toujours un ordre des sommets pour lequel l'algorithme fournisse une coloration optimale : il suffit de prendre une coloration optimale et de numéroter les sommets par ordre croissant de couleur. Malheureusement c'est un "cercle vicieux", l'on ne peut pas connaître à l'avance une telle numérotation.

Dans ce qui suit, on considère un graphe $G = (V, E)$ à colorer et on notera n (resp. m) l'ordre (resp. la taille) de G . Les sommets du graphe seront alors des entiers numérotés de 0 à $n - 1$: $V = [0; n - 1]$.

Les numérotations ou les ordres considérés sont alors des permutations de $[0; n - 1]$ dans lui-même. De plus, si on ne le précise pas lorsqu'il sera question de la complexité d'un algorithme il s'agira généralement d'une complexité temporelle.

ALGORITHME 1 : COLORATION SÉQUENTIELLE

Entrée(s) - Un graphe quelconque $G = (V, E)$

- Une permutation des sommets $\sigma : [0; n - 1] \rightarrow V$

Sortie(s) Une coloration propre $c : V \rightarrow \mathbb{N}$

```
1:  $n := |V|$ ;  
2: Couleur := Tableau (Taille :  $n$ ) (Défaut : -1);  
3: pour  $i$  de 0 à  $n - 1$  faire  
4:    $x := \sigma(i)$ ;  
5:   # Recherche de la plus petite couleur non utilisée dans  $\mathcal{N}(x)$  #;  
6:   Libre := Tableau (Taille : ???) (Défaut : vrai);  
7:   pour  $y$  dans  $\mathcal{N}(x)$  faire  
8:     si Couleur.( $y$ )  $\neq$  -1 alors  
9:       Libre.(Couleur.( $y$ ))  $\leftarrow$  faux;  
10:   fin si  
11: fin pour  
12: index := 0;  
13: tant que Libre.(index) = faux faire  
14:   index := index + 1;  
15: fin tant que
```

```

16:  # Affecter la couleur donnée par index à x #
17:  Couleur.(x) ← index ;
18: fin pour
19: Renvoyer Couleur
    
```

Proposition 3.1. La complexité temporelle d'un algorithme de coloration séquentielle est : $O(m + n)$.

Démonstration

On effectue une première boucle de n itérations. Pour chaque itération, la recherche de `index` (lignes 6 à 15) a une complexité temporelle $O(d_G(x))$, où $d_G(x) = |\mathcal{N}_{G(x)}|$ (nombre d'itérations de la boucle lignes 7 à 11). La complexité totale de ces opérations de recherche est donc dominée par $\sum_{x \in V} d_G(x) = m$. Finalement, on a bien une complexité temporelle totale $O(m + n)$.

3.2.2 Une première heuristique, Welsh & Powell

3.2.2.1 Sur le graphe

Il existe entre autres une méthode heuristique : l'algorithme de Welsh & Powell, il s'agit d'utiliser l'algorithme de coloration séquentiel mais cette fois-ci avec un ordre judicieux, afin d'obtenir une coloration propre la plus "acceptable" possible. L'algorithme de Welsh & Powell consiste ainsi à colorer séquentiellement le graphe en visitant les sommets par ordre de degré décroissant. L'idée est que les sommets ayant beaucoup de voisins seront plus difficiles à colorer, et donc il faut les colorer en premier. Cet algorithme couramment utilisé permet d'obtenir une assez bonne coloration d'un graphe, c'est-à-dire une coloration n'utilisant pas un trop grand nombre de couleurs, mais il n'assure pas que le nombre de couleurs utilisé soit minimum (et donc égal au nombre chromatique). On parle d'heuristique dans le sens où l'algorithme fournit une réponse approchée au problème de coloration.

Nous allons décrire ci-après l'algorithme de coloration de Welsh & Powell :

1. On classe d'abord les sommets du graphe dans l'ordre décroissant de leur degré. On obtient ainsi une liste $x_1 ; x_2, \dots, x_n$ de sommets telle que $\deg(x_1) \geq \deg(x_2) \geq \dots \geq \deg(x_n)$.
2. On choisit une couleur c_1 pour le sommet x_1 , et :

- en parcourant la liste dans l'ordre, on attribue la couleur c_1 au premier sommet non colorié et non adjacent à x_1 .
 - en continuant à parcourir la liste dans l'ordre, on attribue la couleur c_1 aux autres sommets non coloriés et non adjacents aux sommets déjà coloriés avec c_1 et jusqu'à la fin de la liste.
3. S'il reste des sommets non coloriés, on attribue une nouvelle couleur au premier sommet non colorié et on reprend la démarche.
 4. On s'arrête dès que tous les sommets ont été coloriés.

ALGORITHME 2 : WELSH & POWELL

Entrée(s) Un graphe quelconque $G = (V, E)$

Une permutation des sommets $\sigma : [0; n - 1] \rightarrow V$

Sortie(s) Une coloration propre $c : V \rightarrow \mathbb{N}$

- 1: $n := |V|$;
 - 2: Deg := Tableau (Taille : n) (Défaut : 0) ;
 - 3: Sommet := Tableau (Taille : n) (Défaut : 0) ;
 - 4: **pour** i de 0 à $n - 1$ **faire**
 - 5: Deg. $(i) \leftarrow d(i)$;
 - 6: Sommet. $(i) \leftarrow i$;
 - 7: **fin pour**
 - 8: Tri (Tableau : Sommet) (Relation d'ordre : $a \leq b$ ssi Deg. $(a) \geq$ Deg. (b)) ;
 - 9: Coloration séquentielle (Graphe : G) (Numérotation : Sommet)
-

Si on utilise par exemple un tri par dénombrement (en $O(n)$, car les degrés de deux sommets ne peuvent différer de plus de n), on obtient un algorithme de complexité $O(m + n)$. Si on utilise plutôt un tri par comparaison, l'algorithme est alors de complexité $O(n \ln(n) + n)$.

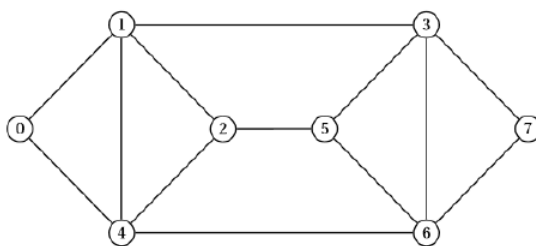
Considérons le graphe G dessiné ci-dessous.

Appliquons l'algorithme décrit ci-dessus : 1,3,6,4,2,5,0,7 est une liste des sommets classés dans l'ordre décroissant de leurs degrés.

D'après l'algorithme :

- à la première étape, on attribue une couleur c_1 aux sommets 1 et 6.

- à la deuxième étape, on attribue une couleur c_2 aux sommets 3 et 4 .
- à la troisième étape, on attribue une couleur c_3 aux sommets 2, 0 et 7.



- enfin à la dernière étape on attribue une couleur c_4 au sommet 5.

Un tel algorithme peut se présenter sous forme de tableau :

| Sommets(degré) | 1(4) | 3(4) | 6(4) | 4(4) | 2(3) | 5(3) | 0(2) | 7(2) |
|----------------|-------|-------|-------|-------|-------|-------|-------|-------|
| Etape1 | c_1 | | c_1 | | | | | |
| Etape2 | | c_2 | | c_2 | | | | |
| Etape3 | | | | | c_3 | | c_3 | c_3 |
| Etape4 | | | | | | c_4 | | |

On obtient un coloriage de G avec 4 couleurs, mais en fait le nombre chromatique de G est 3.

on peut parfois aboutir aux pires colorations possibles($\frac{n}{2}$ au lieu de 2).

L'heuristique DSATUR propose une amélioration du principe de l'algorithme de Welsh & Powell afin d'éviter ce problème.

3.2.2.2 Sur la matrice d'adjacence d'un graphe

Pour colorer les sommets d'un graphe on peut utiliser sa matrice d'adjacence au lieu d'utiliser le graphe complètement et aussi pour colorer le graphe à partir des logiciels informatiques. On peut suivre les étapes suivantes pour colorier les sommets d'un graphe :

- Etape 1 : calculer la somme des éléments de chaque ligne de la matrice et choisir le sommet qui correspond à la quantité maximale puis empiler ce sommet dans un ensemble S.
- Etape 2 : chercher le zéro (toujours dans la ligne correspond au sommet choisi à l'étape 1) qui existe dans une colonne qui ne contient pas des uns barrés et empiler le sommet correspond à ce zéro dans le même ensemble S.
- Etape 3 : barrer toutes les lignes et toutes les colonnes qui correspondent aux sommets de l'ensemble S.

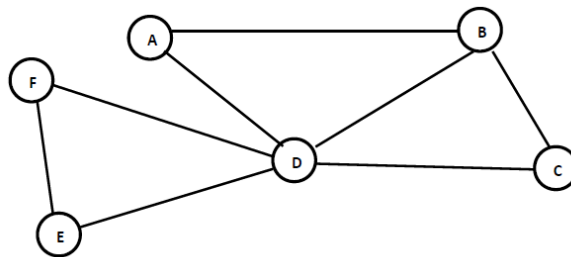
— Etape 4 : Si le test de tous les zéros est fini : Donner une couleur n'est pas encore utilisée à l'ensemble S et supprimer toutes les lignes et toutes les colonnes barrées de la matrice puis refaire les mêmes étapes (à partir de l'étape1) sur la nouvelle matrice à condition que la matrice ne soit pas nulle.

Sinon

Passer à l'étape 2.

Si tous les sommets sont colorés c.à.d. on a la matrice nulle. On s'arrête.

On va colorer le graphe suivant à partir de sa matrice d'adjacence :



La matrice d'adjacence de ce graphe est :

$$M = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Première étape : calculer la somme des éléments de chaque ligne

$$M1 = \begin{matrix} & \begin{matrix} A & B & C & D & E & F \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \\ F \end{matrix} & \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix} \end{matrix} \begin{matrix} 2 \\ 3 \\ 2 \\ 5 \\ 2 \\ 2 \end{matrix}$$

La quantité maximale égale à 5 qui correspond au sommet D dans ce cas la ligne D ne contient aucun deuxième zéro (c'est-à-dire que D est adjacent avec tous les sommets du graphe) donc $S = \{D\}$, puis on affecte la couleur 1 (blanc) à S et on barre la ligne et la colonne D et on supprime les lignes et les colonnes barrées. La matrice M1 devient :

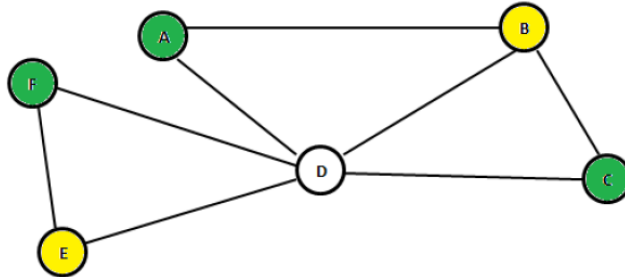
$$M2 = \begin{matrix} & \begin{matrix} A & B & C & F & F \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ E \\ F \end{matrix} & \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{matrix} \begin{matrix} 1 \\ 2 \\ 1 \\ 1 \\ 1 \end{matrix}$$

La quantité maximale égale à 2 qui correspond au sommet B. Le premier zéro correspond au sommet E donc $S = \{B, E\}$, puis on barre les deux lignes et colonnes B et E. La ligne B a un autre zéro correspondant au sommet F mais la colonne F a un (1) barré donc S reste $S = \{B, E\}$. On affecte la couleur 2 (jaune) à S et on supprime les lignes et les colonnes barrées la matrice devient :

$$M3 = \begin{matrix} & \begin{matrix} A & C & F \end{matrix} \\ \begin{matrix} A \\ C \\ F \end{matrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \end{matrix}$$

La matrice M est nulle donc les sommets A , C et F ne sont pas adjacents, on va affecter la couleur 3 (vert) à $S=\{A,C,F\}$.

Le graphe coloré est :



Remarque

L'algorithme de Welsh-Powell ne donne pas nécessairement le nombre minimal de couleurs. Ici, on a pu colorier ce graphe avec trois couleurs, on peut donc en déduire uniquement que : $\delta(G) \leq 3$.

3.2.3 Deuxième heuristique, DSATUR

DSATUR est un algorithme de coloration de graphes créé par Daniel Brélazen 1979 à l'EPFL (École polytechnique fédérale de Lausanne). Cette fois, on considère toujours qu'un sommet est plus difficile à colorer s'il a beaucoup de voisins, mais on prend également en compte le fait qu'un sommet qui possède déjà beaucoup de voisins de couleurs différentes sera aussi problématique. On modifie alors l'algorithme de coloration séquentielle pour colorer en priorité de tels sommets. Pour chaque sommet v de V , on définit le degré de saturation $DSAT(v)$ calculé à un certain moment de l'algorithme de la manière suivante :

Si

aucun voisin de v n'est colorié alors $DSAT(v) = \text{degré}(v)$.

Sinon

$DSAT(a) = \text{nombre de couleurs différentes utilisées pour colorer les voisins de } a$.

L'algorithme consiste alors à colorer séquentiellement le graphe en prenant à chaque étape un sommet non colorié de degré de saturation maximale, où on impose de prendre un sommet de degré maximal en cas de conflit. Il ne faudra pas donc oublier de mettre à jour à chaque étape le degré de saturation d'un sommet. On remarque également que lorsqu'on colore un sommet a , il n'est nécessaire de mettre à jour $DSAT(a)$ que pour les sommets voisins de a .

ALGORITHME 3 : DSATUR

Entrée(s) Un graphe quelconque $G = (V, E)$

Une permutation des sommets $\sigma : [0; n - 1] \rightarrow V$

Sortie(s) Une coloration propre $c : V \rightarrow \mathbb{N}$

```

1:  $n := |V|$ ;
2: Deg := Tableau (Taille :  $n$ ) (Défaut : -1);
3: Deg := Tableau (Taille :  $n$ ) (Défaut : 0);
4: Dsat := Tableau (Taille :  $n$ ) (Défaut : 0);
5: # Initialisation #
6: pour  $i$  de 0 à  $n - 1$  faire
7:   Deg. $(i) \leftarrow d(i)$ ;
8:   Dsat. $(i) \leftarrow d(i)$ ;
9: fin pour
10: # Boucle principale #
11: pour  $i$  de 0 à  $n - 1$  faire
12:    $x := 0$ ;
13:   # Recherche du sommet de Dsatmax #
14:   pour  $y$  de 0 à  $n - 1$  faire
15:     si Couleur. $(x) = -1$  & Dsat. $(y) > Dsat.(x)$  alors
16:        $x := y$ ;
17:     fin si
18:   fin pour
19:   # En cas de conflit, prendre celui de Degmax #
20:   pour  $y$  de 0 à  $n - 1$  faire
21:     si Couleur. $(x) = -1$  & Dsat. $(y) = Dsat.(x)$  & Deg. $(y) > Deg.(x)$  alors
22:        $x := y$ ;
23:     fin si
24:   fin pour
25:   # Recherche de la plus petite couleur non utilisée dans  $\mathcal{N}(x)$  #
26:   Libre := Tableau (Taille : ???) (Défaut : vrai);
27:   pour  $y$  dans  $\mathcal{N}(x)$  faire
28:     si Couleur. $(y) \neq -1$  alors

```

```

29:     Libre.(Couleur.(y)) ← faux;
30:   fin si
31: fin pour
32: index := 0;
33: tant que Libre.(index) = faux faire
34:   index := index + 1
35: fin tant que
36: # Mettre à jour Dsat pour les voisins de x #
37: pour y dans  $\mathcal{N}(x)$  faire
38:   si Dsat.(y) = Deg.(y) alors
39:     Dsat.(y) ← 1;
40:     si index  $\notin$  couleur( $\mathcal{N}(y)$ ) alors
41:       Dsat.(y) ← Dsat.(y) + 1;
42:     fin si
43:   fin pour
44: # Affecter la couleur donnée par index à x #
45: Couleur.(x) ← index;
46: fin pour
47: Renvoyer Couleur;

```

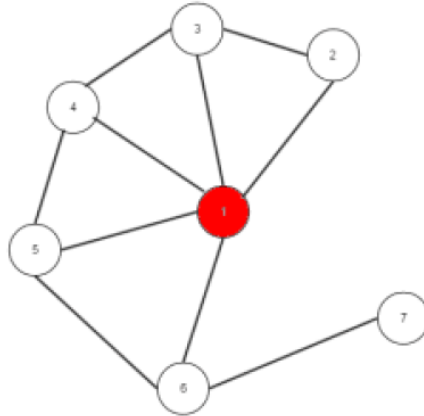
Donnons maintenant quelques explications sur le code proposé : les deux boucles (lignes 14 à 24) pour rechercher le sommet à colorer sont là pour des raisons de clarté, on pourrait très bien se ramener à une seule boucle. Quant à la mise à jour de $DSAT(y)$ pour un voisin y de x et ce avant de colorer x , on distingue trois cas :

- y n'a aucun voisin coloré, alors $DSAT(y) = d(y)$ par définition. Réciproquement, l'égalité $DSAT(y) = d(y)$ n'est vraie que dans deux cas : y n'a aucun voisin coloré, ou y a tous ses voisins colorés de couleurs différentes (auquel cas $|\sigma(\mathcal{N}(y))| = |\mathcal{N}(y)| = d(y)$), mais ce n'est pas le cas ici puisque $x \in \mathcal{N}(y)$ et n'est pas encore coloré. En regardant si $DSAT(y) = d(y)$, on est donc sûr qu'il faut régler $DSAT(y)$ à 1.
- y a des voisins colorés, mais aucun de la couleur index : il faut alors incrémenter $DSAT(y)$.

— y a des voisins colorés, dont un au moins de la couleur index : il n'y a alors rien à faire.

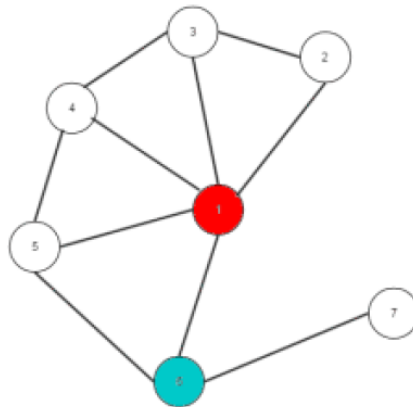
La complexité de l'algorithme DSATUR est $O(n^2)$.

On prend le sommet de plus haut DSAT soit le sommet 1. On lui attribue la première couleur minimale soit le rouge :

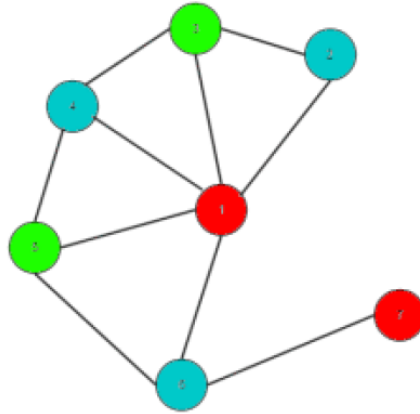


Puis, on passe à la mise à jour des DSAT des voisins du sommet déjà coloré.

On prend le sommet de plus haut DSAT, Nous allons prendre le sommet 6. On lui attribue la deuxième couleur minimale soit le bleu :



On fait la même chose pour les autres sommets, on obtient le graphe coloré suivant :



3.3 Graphes triangulés : un algorithme linéaire de coloration optimale

3.3.1 Généralités sur les graphes triangulés

Nous allons maintenant donner un algorithme optimal qui permet de colorer une classe de graphe bien particulière : les graphes triangulés. Nous présentons ici les définitions et propriétés utiles à cet algorithme.

- Un graphe $G = (V, E)$ est dit triangulé s'il ne contient pas de trou de longueur ≥ 4 .
- Un sommet a de G est simplicial si tous ses voisins sont adjacents 2 à 2.
- Un schéma (ou ordre) d'élimination simplicial est une permutation $\sigma : V \rightarrow [0; n - 1]$ telle que pour tout i dans $[0, n - 1]$, $\sigma(i)$ soit simplicial dans $G[\sigma < [0, i] >]$. On peut alors démontrer la caractérisation suivante, sur laquelle repose l'algorithme LexBFS :

Proposition 3.2. G est triangulé si et seulement si il admet un schéma d'élimination simplicial. L'utilité d'une telle caractérisation est manifeste, car on peut montrer qu'une coloration séquentielle selon un tel ordre offre une coloration optimale. Le problème consiste donc à trouver efficacement un tel schéma dans le cas des graphes triangulés, c'est le but de l'algorithme LexBFS.

3.3.2 Parcours en largeur lexicographique - LexBFS

Il se présente comme une variante du parcours en largeur classique de graphe (dit BFS pour "Breadth First Search"). Considérons l'alphabet $A = [1, n]$ et munissons l'ensemble des mots sur A de l'ordre lexicographique \leq , défini intuitivement comme étant

l'ordre du dictionnaire. On associe alors à chaque sommet x une étiquette $L(x) \in A^*$, initialement égale à ϵ où ϵ désigne le mot vide de A^* . On effectue ensuite les opérations suivantes :

Pour i variant de 0 à $n - 1$:

1. Prendre un sommet x d'étiquette maximale pour \leq parmi les sommets non visités.
2. Affecter le numéro i à x .
3. Effectuer la mise à jour $L(y) \leftarrow L(y) \cdot (n-i)$ pour $y \in \mathcal{N}(x)$.

ALGORITHME 4 : PARCOURS EN LARGEUR LEXICOGRAPHIQUE

Entrée(s) Un graphe quelconque $G = (V, E)$, un sommet source $s \in V$

Sortie(s) Une bijection $\sigma : V \rightarrow [0; n - 1]$

- 1: $n := |V|$;
 - 2: Numéro := Tableau (Taille : n) (Défaut : -1);
 - 3: Label := Tableau (Taille : n) (Défaut : "");
 - 4: **pour** i de 0 à $n - 1$ **faire**
 - 5: Soit x tel que $\text{Label}(x) = \max \text{Label}(y)$, $\text{Numéro}(y) = -1$;
 - 6: $\text{Numéro}(x) \leftarrow i$;
 - 7: **pour** $y \in \mathcal{N}(x)$ **faire**
 - 8: **si** $\text{Numéro}(y) = -1$ **alors**
 - 9: $\text{Label}(y) \leftarrow \text{Label}(y) \text{StringofInt}(n - 1)$;
 - 10: **fin si**
 - 11: **fin pour**
 - 12: **fin pour**
 - 13: Renvoyer Numéro;
-

Nous obtenons alors un ordre $\sigma : V \rightarrow [0; n - 1]$ sur les sommets du graphe, appelé "ordre lexicographique" qui vérifie la propriété suivante :

Proposition 3.3. Il en découle, en enlevant successivement le dernier sommet visité que σ^{-1} est un schéma d'élimination simplicial si G est triangulé.

Exemple d'application

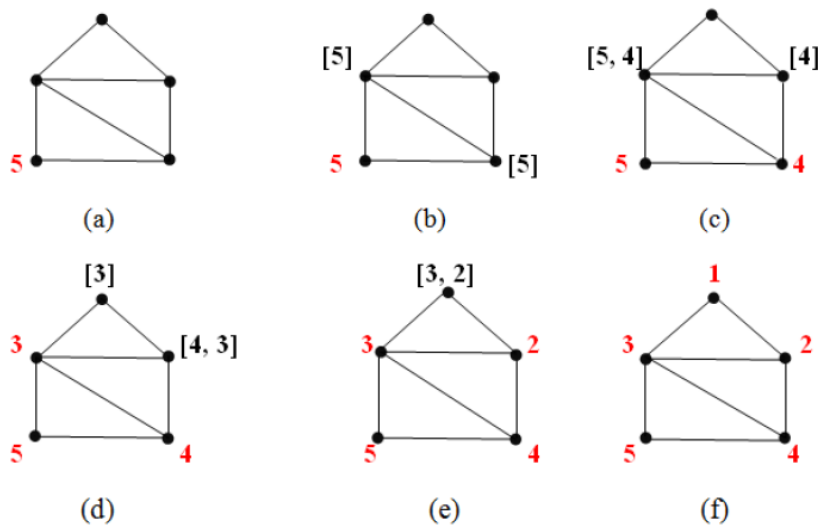


FIGURE 3.1 – Déroulement de l’algorithme Lex-BFS sur un exemple

3.4 Les métaheuristiques

Par opposition aux méthodes constructives qui partent d’un graphe non coloré et attribuent graduellement une couleur à un sommet ou à un ensemble de sommets pour aboutir à une coloration totale du graphe, les métaheuristiques débutent par une coloration du graphe choisie arbitrairement et tentent d’obtenir une meilleure solution en manipulant la coloration courante.

les algorithmes les plus nombreux dans la littérature, on peut les subdiviser en trois types : les algorithmes basés sur la recherche locale, les algorithmes génétiques et les algorithmes hybrides.

3.4.1 Les Algorithmes de recherche locale

Les algorithmes de recherche locale sont devenus très prisés depuis la publication du très populaire TABUCOL par Hertz et de Werra en 1987 en s’inspirant de la recherche taboue de Glover. À partir d’une coloration (solution initiale) du graphe avec conflits – arêtes ayant la même couleur à leurs extrémités, TABUCOL tente de minimiser le nombre de conflits (fonction objectif) en modifiant la couleur d’un sommet intervenant dans un conflit (voisinage).

La couleur changée ne peut être assignée au même sommet qu'après un certain nombre d'itérations : cette couleur est dite taboue pour ce sommet. La simplicité de TABUCOL et le nouveau type de voisinage suggéré par Morgenstern–Impasse Class neighborhood en 1996 ont contribué au design de meilleurs algorithmes proposés par la suite : Variable Neighborhood Search (VNS) de Avanthay et al. DYN PARTIACOL et FOO-PARTIACOL de Blöchliger et Zuffrey, et Variable Search Space (VSS) de Hertz et al. Le recuit simulé est une méthode de recherche locale également très connue. Il se différencie principalement de la recherche taboue par le critère d'acceptation d'une solution dans le voisinage. Ce critère est déterministe pour la recherche taboue mais repose sur un processus probabiliste pour le recuit simulé.

3.4.2 Les algorithmes génétiques

Une autre approche de résolution du problème a été suggérée par Davis en 1991. Il s'agit d'un algorithme purement génétique qui représente une solution par la permutation des sommets et leur attribue de façon séquentielle la première couleur disponible. Les résultats obtenus sur les graphes ne sont pas fameux comparés aux autres métaheuristiques.

3.4.3 Les algorithmes hybrides

La combinaison des algorithmes de recherche locale et des algorithmes génétiques a donné naissance aux algorithmes hybrides ou mémétiques. Ces derniers fournissent de bons résultats sur les graphes au prix d'une complexité sans cesse accrue. Le principe consiste à explorer et diversifier une population de solutions grâce à deux types principaux d'opérateurs : l'opérateur de mutation qui crée de nouvelles solutions en modifiant l'une des solutions initiales à l'aide d'un algorithme de recherche locale et l'opérateur de croisement qui crée de nouvelles solutions en combinant des parties de deux ou de plusieurs solutions existantes.

3.5 Conclusion

Les quatre algorithmes que nous avons présenté dans le troisième chapitre fournissent des colorations qui diffèrent selon que la coloration obtenue est optimale ou non.

CHAPITRE 4

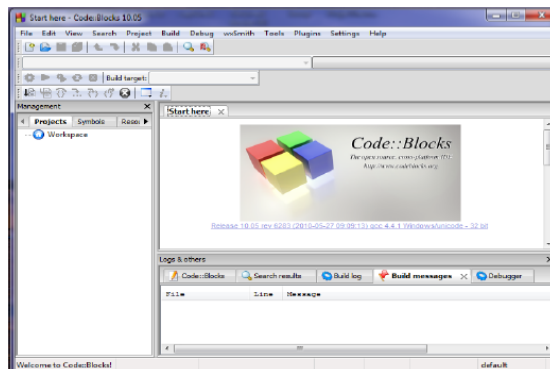
APPLICATION

Maintenant que nous avons terminé avec l'aspect théorique de la coloration des graphes et mis en évidence les différents algorithmes permettant de les colorer, il reste à aborder la partie pratique : l'implémentation des algorithmes en C et quelques applications de la coloration des graphes dans le monde réel.

4.1 Introduction au Code Block

Code Blocks est un environnement de développement intégré libre et multiplateforme. Il est écrit en C++ grâce à la bibliothèque wxWidgets. Code : :Blocks est orienté C et C++, mais il supporte d'autres langages comme FORTRAN ou le D.

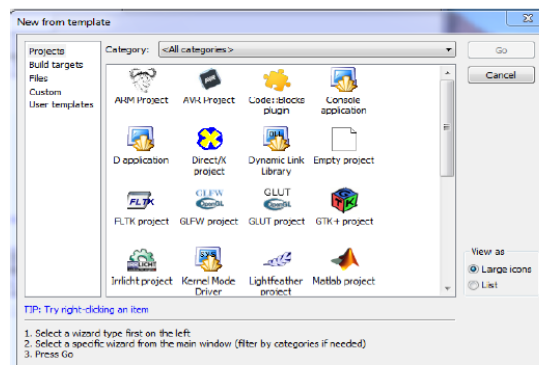
Code Blocks est développé pour Linux, Windows et Mac OS X. Des utilisateurs indiquent avoir réussi à compiler le code source sous FreeBSD. L'illustration cidessous montre l'apparence de la fenêtre de l'interface utilisateur de Code-Blocks.



Dans CodeBlocks, les sources et les paramètres d'un processus de génération sont stockés dans un fichier projet < name > :cbp. Les sources en C/C++ et les fichiers d'entêtes correspondants (ou headers) sont les composants typiques d'un projet. La façon la plus simple de créer un projet est de passer par la commande "Fichier" → "Projet" et de choisir un assistant. Vous pouvez alors ajouter des fichiers au projet via le menu de contexte 'Ajouter des fichiers' de la fenêtre de gestion. CodeBlocks gère les fichiers de projets en catégories qui dépendent de l'extension des fichiers. Les catégories suivantes sont prédéfinies :

- Sources contient les fichiers sources dont l'extension est *.c ; *.cpp.
- ASM Sources contient les fichiers sources dont l'extension est *.s ; *.S ; *.ss ; *.asm.
- Headers contient, entre autres, les fichiers dont l'extension est *.h ;. Ressources contient les fichiers pour paramétrer l'aspect des fenêtres des wxWidgets avec les extensions *.res ; *.xrc ;. Ces types de fichiers sont affichés dans l'onglet 'Ressources' de la fenêtre de Gestion.

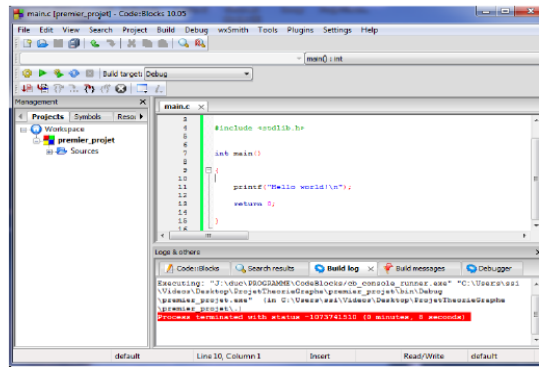
Pour créer un nouveau projet on click sur le menu File → New → Project... Une boîte de dialogue s'ouvre alors pour vous permettre de choisir le type de projet que vous souhaitez créer.



Code Blocks place automatiquement tous les fichiers du projet dans un répertoire qui porte le nom du projet.

Construire et executer un projet

Vous pouvez éditer le fichier main. En allant à gauche dans la fenêtre Management dans Sources → main.c. Ce fichier comporte la fonction main, fonction principale du programme.



4.2 Allocation des fréquences (Gestion du spectre de fréquence et implémentation des réseaux de télécommunications)

4.2.1 Introduction

Depuis les années 80, le problème d'affectation de fréquences a fait l'objet d'études menées par différents chercheurs dans le but d'une meilleure gestion. Le spectre de fréquences qui est attribué aux opérateurs de téléphonie est divisé en canaux fréquentiels. L'allocation de fréquences regroupe les mécanismes et procédures mis en œuvre afin de gérer l'attribution des canaux de fréquences aux demandes de communication. Cette gestion permet de déterminer la qualité du réseau.

Le spectre radio électrique est devenu une ressource extrêmement précieuse, en raison de son exploitation massive par des systèmes de communication de toutes sortes.

Il faut donc des méthodes novatrices pour gérer le spectre de manière dynamique afin que cette ressource puisse être disponible pour les nouveaux services à même d'éviter les brouillages de signaux.

L'objet de cette application sera de poser la problématique de l'affectation de fréquences et la traduire en théorie des graphes ; Ceci nous permettra d'une part de proposer des solutions à même de résoudre cette problématique et d'autre part, de présenter quelques définitions de base qui nous permettront de mieux comprendre la problématique posée. Le but de cette étude sera aussi de satisfaire la demande de chaque station de base en terme de fréquences tout en minimisant l'ensemble des interférences ainsi que la partie du spectre radio.

Pour cela, et à l'issue de ce travail nous proposons l'implémentation de quelques algorithmes

de coloration des graphes permettant la gestion du spectre de fréquence sur les stations de bases en utilisant un minimum de fréquences pour une couverture maximale.

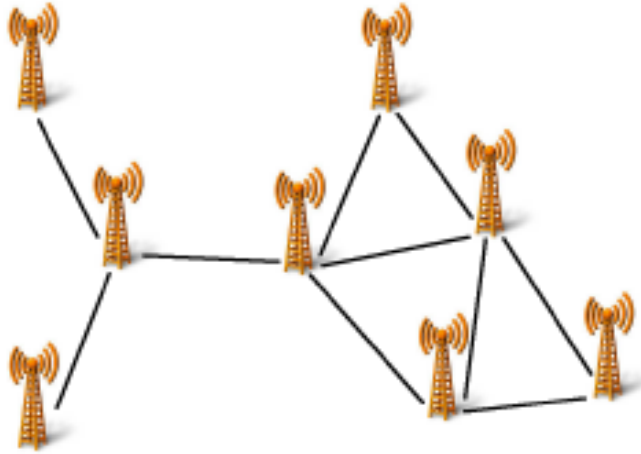


FIGURE 4.1 – Un réseau d'antennes.

4.2.2 Quelques définitions

L'attribution des fréquences : est l'ensemble des mécanismes qui permettent de définir comment sont réparties les fréquences entre les différents acteurs. L'enjeu principal est d'éviter les interférences entre les émetteurs.

Fréquence : En physique, la fréquence désigne en général la mesure du nombre de fois qu'un phénomène périodique se reproduit par unité de temps, C'est-à-dire le nombre de fois qu'un phénomène temporel régulier se reproduit identique à lui-même par intervalle de temps donné. L'unité de fréquence étant l'hertz(hz).

Spectre de fréquence : Le spectre de fréquence est la partie du spectre électromagnétique qui achemine les ondes radio, Il est subdivisé en neuf bandes de fréquences désignées par des nombres entiers consécutifs.

4.2.3 Les interférences

On parle d'interférence lorsqu'un point donné de l'espace de couverture reçoit en plus du signal utile (assurant le service) un signal dit interfèrent de puissance relativement élevée et porté sur une fréquence identique ou adjacente.

Il existe deux grands types d'interférences, mais dans notre cas nous intéressons à celles qui

sont dues à la réutilisation d'une même fréquence (interférence Co-canal).

4.2.4 Problématique

Aujourd'hui les opérateurs radio doivent faire face à un double déficit : répondre à une augmentation croissante du trafic tout en maintenant une bonne qualité radio. La difficulté du problème résulte du fait qu'une solution acceptable doit satisfaire des contraintes multiples, la plupart des fois contradictoires.

L'une des contraintes les plus sévères c'est le nombre restreint de fréquences (canaux) disponibles pour l'allocation (le spectre radio étant une ressource très limitée). Cette contrainte impose un degré élevé de réutilisation des fréquences, fait qui augmente la probabilité d'interférence.

Les gestionnaires disposent d'un nombre limité de fréquences pour couvrir la totalité de la zone d'étude (un territoire). Pour cela, la réutilisation des fréquences est par conséquent inévitable pour augmenter la capacité du réseau et répondre à la demande de plus en plus importante. Une distribution optimale du spectre sur les stations doit garantir un écoulement maximal du trafic tout en minimisant les interférences. L'écoulement de trafic porté par ces zones est alors conditionné par la qualité de l'affectation des fréquences.

Afin de satisfaire cette demande, il est indispensable d'exploiter cette ressource d'une manière rationnelle et optimale. Il est donc légitime de s'intéresser à une modélisation plus fine de la problématique de planification de fréquences. Un tel projet nécessite l'utilisation de méthodes scientifiques basées sur des techniques de recherche opérationnelle.

4.2.5 Modélisation mathématique du problème par la théorie des graphes

Comme mentionné précédemment, les gestionnaires sont souvent confrontés à de multiples problèmes et leur résolution s'avère souvent une tâche difficile. Ces problèmes se présentent sous forme de données de contraintes dont on doit tenir compte.

Pour arriver à résoudre un problème donné, nous devons commencer par interpréter tous ses paramètres et les transformer sous des formes qu'on peut gérer. La première étape dans la résolution d'un problème consiste en sa projection dans un espace permettant ainsi diverses manipulations sur le problème projeté. Ce dernier s'appelle le modèle associé au problème. La modélisation est donc une traduction des paramètres du problème dans un langage

accessible par la méthode de résolution utilisée, ou bien c'est une façon de décrire le problème sous une forme qui introduit sa réalisation.

Enfin, la modélisation d'un problème doit pouvoir donner une interprétation aux solutions concrètes répondant aux besoins du problème réellement posé.

Les données du problème :

N : nombre de stations.

N_F : nombre de fréquences disponibles.

$d(n_i; n_j)$: la distance entre deux stations n_i et n_j .

Contrainte :

Respecter la contrainte d'interférence, veut dire : éviter d'assigner la même fréquence à deux stations voisines.

Objectifs :

Trouver un bon plan de fréquence qui doit minimiser l'ensemble des interférences.

Soit un graphe $G = (X, E)$ défini par :

X : L'ensemble des sommets du graphe représentent les stations.

E : L'ensemble des arêtes du graphe représentent les risques d'interférences.

Il existe une arête $[x_i; x_j]$ de E ssi x_i est voisin avec x_j .

Pour résoudre le problème d'affectation de fréquences, on utilise la coloration des sommets qui consiste à affecter à tous les sommets du graphe une couleur (fréquence) de façon que chaque paire de sommets adjacents soit de couleurs différentes ; En d'autre termes, s'il existe une arête $[x_i; x_j]$ de E alors on a $c(x_i) \neq c(x_j)$.

Le nombre minimum de couleurs nécessaires pour colorier ce graphe en respectant cette contrainte, est appelé le nombre chromatique $X(G)$.

L'application de la théorie des graphes va nous permettre de trouver le nombre minimal de fréquences allouées aux stations de bases et qui minimise l'intégralité des interférences.

4.2.6 Problèmes tests

Afin de valider notre approche, nous exploitons les données générées aléatoirement présentées ci dessous à savoir.

Nous avons défini 16 stations et les distances entre elles, on désire attribuer des canaux de fréquences à ces stations. Deux stations distantes de moins de 45 km ne peuvent pas avoir le même canal afin d'éviter les interférences. Le but est de chercher le nombre minimum de canaux distincts nécessaires pour éviter toute interférence.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1 | - | 40 | 45 | 75 | 36 | 70 | 58 | 45 | 88 | 25 | 83 | 37 | 85 | 73 | 50 | 82 |
| 2 | 40 | - | 34 | 83 | 92 | 51 | 82 | 86 | 93 | 89 | 45 | 75 | 39 | 65 | 39 | 33 |
| 3 | 45 | 34 | - | 37 | 28 | 37 | 89 | 89 | 64 | 78 | 77 | 90 | 45 | 38 | 66 | 91 |
| 4 | 75 | 83 | 37 | - | 45 | 36 | 31 | 93 | 93 | 45 | 67 | 75 | 90 | 40 | 63 | 63 |
| 5 | 36 | 92 | 28 | 45 | - | 70 | 79 | 30 | 36 | 67 | 50 | 90 | 79 | 94 | 81 | 87 |
| 6 | 70 | 51 | 37 | 36 | 70 | - | 45 | 80 | 89 | 70 | 68 | 94 | 89 | 42 | 77 | 70 |
| 7 | 58 | 82 | 89 | 31 | 79 | 45 | - | 81 | 37 | 89 | 72 | 77 | 54 | 64 | 66 | 69 |
| 8 | 45 | 86 | 89 | 93 | 30 | 80 | 81 | - | 26 | 41 | 94 | 79 | 70 | 63 | 52 | 80 |
| 9 | 88 | 93 | 64 | 93 | 36 | 89 | 37 | 26 | - | 94 | 78 | 70 | 64 | 58 | 83 | 69 |
| 10 | 25 | 89 | 78 | 45 | 67 | 70 | 89 | 41 | 94 | - | 75 | 81 | 69 | 58 | 79 | 67 |
| 11 | 83 | 45 | 77 | 67 | 50 | 68 | 72 | 94 | 78 | 75 | - | 33 | 67 | 86 | 94 | 24 |
| 12 | 37 | 75 | 90 | 75 | 90 | 94 | 77 | 79 | 70 | 81 | 33 | - | 80 | 85 | 55 | 73 |
| 13 | 85 | 39 | 45 | 90 | 79 | 89 | 54 | 70 | 64 | 69 | 67 | 80 | - | 75 | 45 | 71 |
| 14 | 73 | 65 | 38 | 40 | 94 | 42 | 64 | 63 | 58 | 58 | 86 | 85 | 75 | - | 79 | 94 |
| 15 | 50 | 39 | 66 | 63 | 81 | 77 | 66 | 52 | 83 | 79 | 94 | 55 | 45 | 79 | - | 53 |
| 16 | 82 | 33 | 91 | 63 | 87 | 70 | 69 | 80 | 69 | 67 | 24 | 73 | 71 | 94 | 53 | - |

FIGURE 4.2 – Graphe d’interférences correspondant au tableau ci-dessus.

Afin de convertir le problème d’allocation des fréquences en un problème de coloration, On va traduire le tableau ci-dessus en terme de graphe dont les sommets sont les stations, et les arêtes relient deux stations qui interfèrent entre elles.

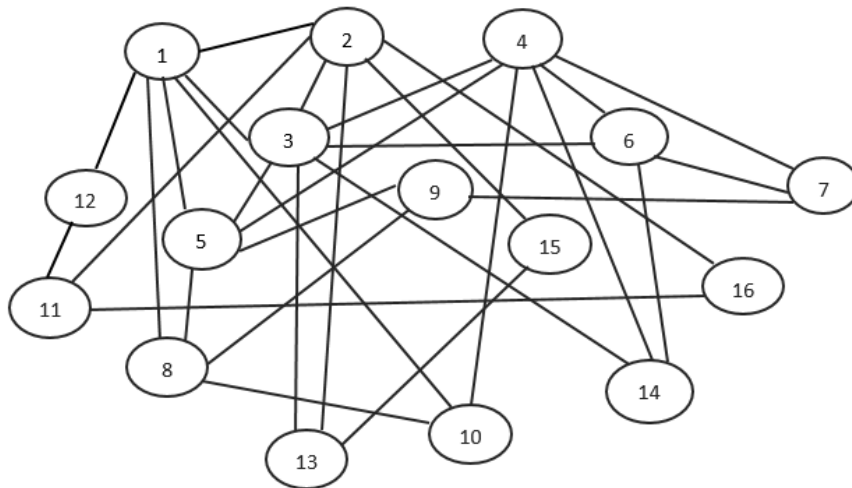


FIGURE 4.3 – Graphe d’interférences correspondant au tableau ci-dessus.

4.2.7 Coloration de graphe d’interférences

Après la modélisation du réseau des stations par le graphe appelé graphe d’interférences. On va maintenant s’intéresser à sa coloration par l’algorithme de Welsh-Powell détaillé

précédemment, les fréquences à allouer correspondent à des couleurs avec lesquelles nous colorons les sommets du graphe. La coloration doit être une coloration propre : deux sommets adjacents ne peuvent recevoir la même couleur.

La matrice d'adjacente noté S associée à ce graphe est donné par :

$$S = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

En suivant les mêmes étapes de l'exemple on trouve :

$$S_1 = \{3, 7, 8, 11, 15\} \implies \text{couleur 1}$$

$$S_2 = \{1, 4, 9, 13, 16\} \implies \text{couleur 2}$$

$$S_3 = \{6\} \implies \text{couleur 3}$$

$$S_4 = \{2, 5, 10, 12, 14\} \implies \text{couleur 4}$$

La résolution manuellement de l'algorithme Welsh-Powell nous donne 4 couleurs, telles que chaque couleur c_i est affectée à l'ensemble de station $S_i, \forall i \in \{1, \dots, 4\}$. Alors chaque ensemble de ces dernières peut recevoir le même canal de fréquence .

Programmation de l'algorithme Welsh-Powell en langage C :

L'algorithme de Welsh-Powell a été implémenté en C et la compilation de programme nous donne les informations suivantes :

```

"C:\Users\Master\Desktop\CHAPITRE 4\powel exemple 1\main.exe"
Couleur 1 pour les sommets : 3 7 8 11 15
Couleur 2 pour les sommets : 1 4 9 13 16
Couleur 3 pour les sommets : 6
couleur 4 : Pour le reste des sommets
Appuyez sur une touche pour continuer...
    
```

Et par suite nous colorons notre graphe d'interférences à partir de ces données :

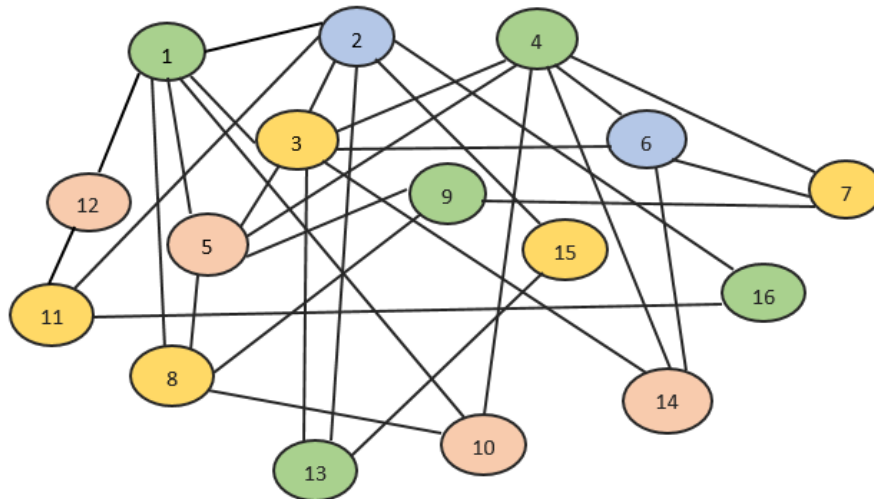


FIGURE 4.4 – Coloration obtenue par l'algorithme de Welsh and Powell.

Comme nous voyons 4 couleurs sont suffisantes pour colorier notre graphe. cela veut dire que le nombre minimum de canaux de fréquences distincts nécessaires pour éviter toute interférence entre les 16 stations de bases est quatre canaux.

4.3 Coloration des cartes géographiques

4.3.1 Introduction

Durant les études, essentiellement pendant un cours de géographie, nous avons sûrement tous coloré une carte géographique, comme par exemple celle des provinces de la Belgique (voir figure 10). Nous sommes confrontés au problème suivant : nous devons colorer cette carte de telle manière à ce que deux provinces limitrophes aient des couleurs différentes. Etant étudiant, nous n'avons que peu de crayons de couleurs dans notre trousse. Dès lors, nous nous sommes demandés : avons-nous suffisamment de couleurs différentes pour colorer la carte ? Est-ce que nous devons emprunter un crayon à notre voisin ? Nous pouvons à la main, montrer qu'il faut au minimum 3 couleurs pour cette carte (voir figure 11), deux couleurs ne suffisent clairement pas.



FIGURE 10 – Provinces de Belgique.



FIGURE 11 – Provinces de Belgique colorées avec 3 couleurs.

La coloration des cartes géographiques consiste à donner une couleur à chaque pays (où région) à condition que deux pays frontaliers ne soient pas porteurs de la même couleur.

4.3.2 Coloration de la carte de Béjaia

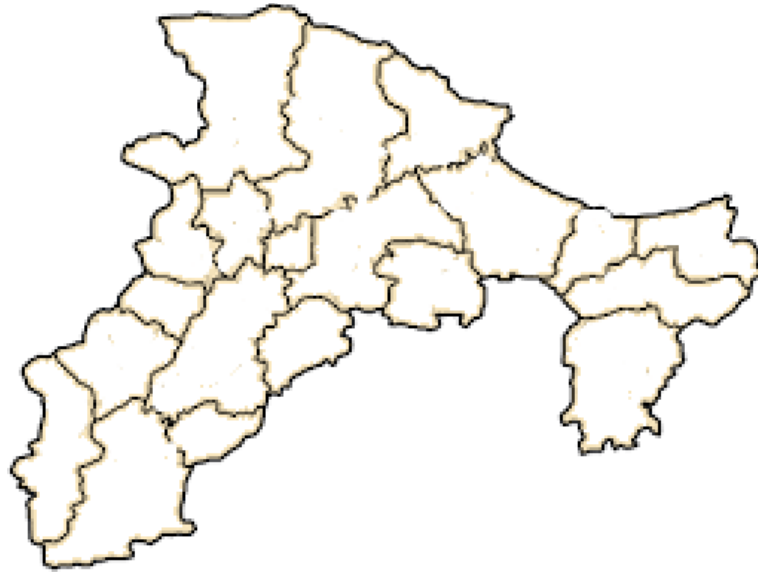
4.3.2.1 Présentation des Daïras de Béjaia

| Daïra | Nombre de communes | Communes | Superficie (km ²) | Population (hab.) |
|----------------|--------------------|---|-------------------------------|-------------------|
| Adekar | 3 | Adekar • Taourirt Ighil • Beni Ksila | 363,10 | 24 105 |
| Akbou | 4 | Akbou • Chellata • Ighram • Tamokra | 212,29 | 78 454 |
| Amizour | 4 | Amizour • Beni Djellil • Semaoun • Ferraoun | 212,88 | 74 455 |
| Aokas | 2 | Aokas • Tizi N'Berber | 80,63 | 28 613 |
| Barbacha | 2 | Barbacha • Kendira | 129,33 | 22 265 |
| Béjaïa | 2 | Béjaïa • Oued Ghir | 168,15 | 197 333 |
| Beni Maouche | 1 | Beni Maouche | 94,86 | 13 412 |
| Chemini | 4 | Chemini • Tibane • Souk-Oufella • Akfadou | 100,27 | 36 766 |
| Darguina | 3 | Darguina • Aït-Smail • Taskriout | 140,67 | 32 582 |
| El Kseur | 3 | El Kseur • Fenaïa Ilmaten • Toudja | 306,40 | 49 496 |
| Ighil Ali | 2 | Ighil Ali • Aït-R'zine | 269,93 | 24 089 |
| Kherrata | 2 | Kherrata • Draâ El-Kaïd | 217,85 | 81 265 |
| Ouzellaguen | 1 | Ouzellaguen | 61,40 | 22 719 |
| Seddouk | 4 | Amalou • M'cisna • Bouhamza • Seddouk | 268,54 | 64 298 |
| Sidi-Aïch | 5 | Sidi-Aïch • Leflaye • Tinabdher • Tifra • Sidi Ayad | 81,69 | 39 839 |
| Souk El-Ténine | 3 | Melbou • Souk El Ténine • Tamridjet | 127,02 | 33 854 |
| Tazmalt | 3 | Tazmalt • Beni Mellikeche • Boudjellil | 176,29 | 48 874 |
| Tichy | 3 | Boukhelifa • Tichy • Tala Hamza | 211,87 | 36 987 |
| Timezrit | 1 | Timezrit | 38,09 | 25 853 |

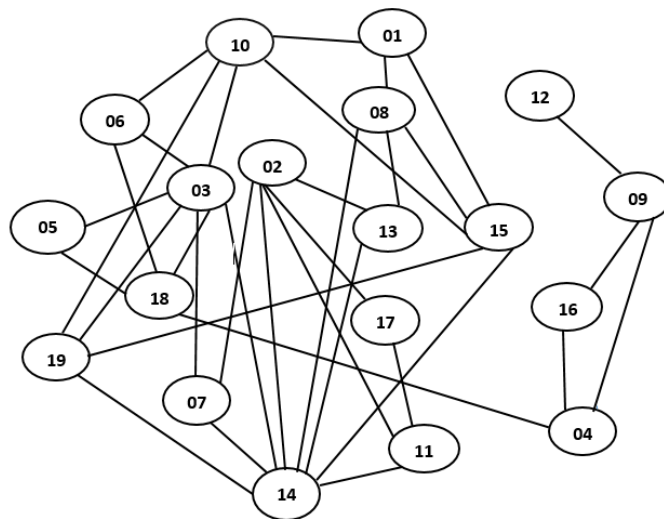
FIGURE 4.5 – Les Daïras de la Wilaya de Béjaïa.

Pour colorier la carte composée de 19 Daïras en utilisant l'algorithme de Welsh-Powell, on va transformer la carte en graphe tel que les sommets représentent les Daïras. Si deux Daïras sont frontalières on relie les sommets correspondants à ces Daïras par une arête.

Premièrement on va présenter notre carte géographique correspond à la Wilaya de Béjaïa :



Le graphe correspond à cette carte est le suivant :



La matrice d'adjacente notée M associée à ce graphe est :

$$M = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

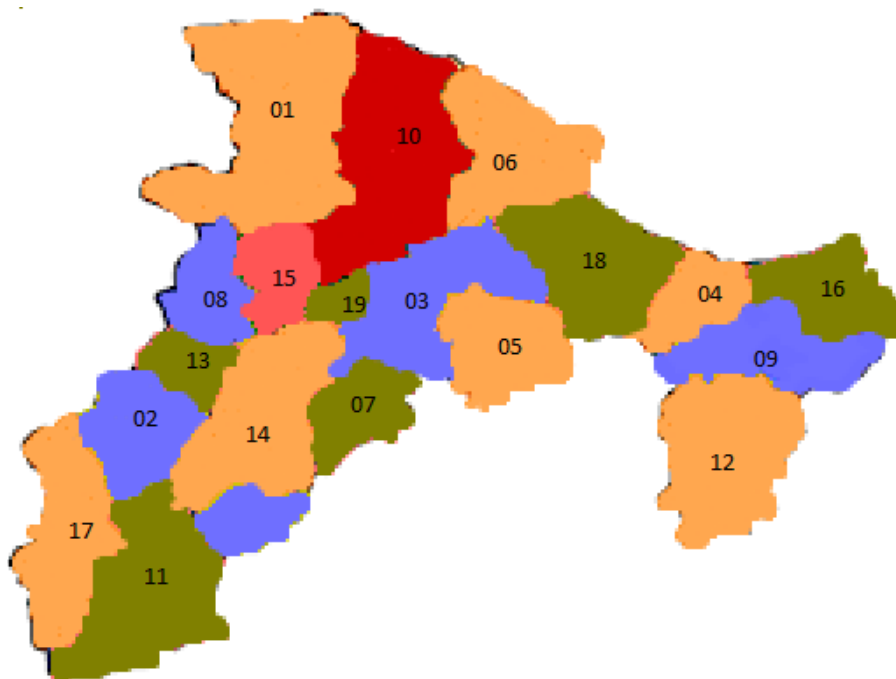
En appliquant l’algorithme de welsh-Powell en langage C :

Le programme marche pour la coloration de tous les graphes, ce n’est pas seulement pour ce graphe, on peut colorer n’importe quelle carte.

La compilation de programme donne les informations suivantes :

```
"C:\Users\Master\Desktop\CHAPITRE 4\powl exemple 1\main.exe"
Couleur 1 pour les sommets : 14 1 4 5 6 12 17
Couleur 2 pour les sommets : 3 2 8 9
Couleur 3 pour les sommets : 10
Couleur 4 pour les sommets : 15
couleur 5 : Pour le reste des sommets
Appuyez sur une touche pour continuer...
Process returned 0 (0x0)   execution time : 798.629 s
Press any key to continue.
```

Et par suite nous colorons notre carte à partir de ces données :



Comme nous voyons 5 couleurs sont suffisantes pour colorier notre carte.

4.4 Conclusion

Dans ce dernier chapitre on a fait deux applications de la coloration des sommets d'un graphe. Pour cela, on a utilisé juste l'algorithme de Welsh-Powell même s'il en existe d'autres, cependant cet dernier est le plus pratique. On a traité dans un premier lieu le problème d'allocation des fréquences, après la modélisation du réseau des stations par le graphe appelé graphe d'interférences, on a essayé de projeter les étapes de notre algorithme sur la matrice d'adjacence du graphe dans le but de colorer les sommets en utilisant le langage C. La seconde application existe au quatrième chapitre (coloration des cartes géographiques), on a choisi de colorer la carte géographique de Béjaia toujours avec l'algorithme de Welsh-Powell en langage C (c'est le plus important dans ce travail).

Il est à noter que notre programme fonctionne pour la coloration des sommets de tous les graphes.

Conclusion et Perspectives

Le problème de coloration des graphes est un domaine très actif de la théorie des graphes du fait de ses nombreuses applications. Dans ce mémoire nous sommes intéressés à la coloration des sommets.

Dans ce manuscrit, nous avons présenté les notions de bases de la théorie des graphes. Dans un second temps, nous nous sommes intéressés aux notions de la coloration des sommets et ses différentes propriétés.

nous avons abordé quelques techniques de coloration des sommets en présentant les quatre algorithmes. Ces derniers fournissent des colorations qui diffèrent selon que la coloration obtenue est optimale ou non.

Dans le quatrième chapitre, nous avons abordé une application de la coloration des graphes dans le monde réel et nous avons appliqué l'algorithme Welsh-Powell implémentés en C aux deux application que nous avons présenté.

Nous avons abordé quelques applications de la coloration des graphes dans le monde réel.

Pour celà, nous avons utilisé juste l'algorithme de Welsh-Powell même s'il en existe d'autres algorithmes comme : algorithme de Glouton, algorithme DSATUR, algorithme LexBFS.

Cependant, l'algorithme de Welsh-Powell marche mieux en pratique.

En termes de perspectives, nous visons à :

- Développer d'autres algorithmes qui seront plus efficaces et plus performants que ceux qui ont été présenté précédemment.
- Améliorer des algorithmes d'approximation de temps polynomial.
- Développer des algorithmes heuristiques ou métaheuristique.
- Chercher des algorithmes fournissant des solutions optimales exactes pour des cas particuliers de graphe.

BIBLIOGRAPHIE

- [1] S. Agueniou L. Djerroud A. Rebehi. Problème de cheminement dans les graphes. mémoire Université Béjaia, (2010/2011). DEF CHAP1
- [2] R. L. Brooks. On colouring the nodes of a network. Proc. Cambridge Phil. Soc, 37 :194-147, (Cité en page 14).
- [3] J.A. Bondy and U.S.R. Murty. Théorie des Graphes. 2008. (Cité en page 11 et 15).
- [4] P., Zhang, Color-Induced Graph Colorings, Springer Briefs in Mathematics, 2015.
- [5] P., Zhang, Graphs and digraphs, 6^{ème} édition, Taylor et Francis Group, 2016.
- [6] C. Berge. Graphs and hypergraphs. Dunod, Paris, 1970.
- [7] D. B. West, Introduction to Graph Theory, Pearson education, 2001.
- [8] M., Kubale, Graph colorings, American Mathematical Society, 2000.
- [9] A. Ghouila-Houri, Sur une conjecture de Berge (mimeo), 1960.
- [10] C. Berge, Les problèmes de coloration en théorie des graphes, Publications de l'Institut de Statistiques de l'Université de Paris (1960) 123-160.
- [11] M. Grötschel, L. Lovász, A. Schrijver, The ellipsoid method and its consequences in combinatorial optimization, Combinatorica 1 (1981) 169-197.
- [12] L. Lovász, A characterization of perfect graphs, Journal of Combinatorial Theory B13 (1972) 95-98.
- [13] L. Lovász, Normal hypergraphs and the perfect graph conjecture, Discrete Mathematics 2 (1972) 253-267.
- [14] M. Chudnovsky, N. Robertson, P. Seymour, R. Thomas, The strong perfect graph theorem, Annals of Mathematics 164 (2006) 51-229.

- [15] M. Chudnovsky, G. Cornuéjols, X. Liu, P. Seymour, K. Vučković, Recognizing Berge Graphs, *Combinatorica* 25 (2005) 143-186.
- [16] K. Appel and W. Haken, Every map is four colorable, *Discharging?* Illinois Journal of mathematics, 1977.
- [17] N. L. Biggs, E. K. Lloyd, R. J. Wilson, Graph theory 1736 - 1936, clarendon press oxford, 1976.
- [18] M. Karonski, T. Łuczak et A. Thomason, Edge weights and vertex colours, *Journal of Combinatorial Theory, Series B*, 91 (2004), pp. 151–157.
- [19] P. Zhang. Graphs and disgraphs. 6^{ème} édition, Taylor et Francis Group, 2016. (Cité en pages 14 et 15).

Résumé

La coloration de graphes est un problème classique de la théorie des graphes. Dans ce mémoire, nous nous intéressons à aborder certainement l'un des plus fameux sujets de la théorie des graphes : la coloration des sommets. Il s'agit de colorer les sommets d'un graphe afin que deux sommets adjacents n'aient pas la même couleur. En particulier, nous montrons quelques techniques de coloration des sommets utilisées dans ce cadre afin d'obtenir des meilleurs résultats.

Un intérêt particulier est porté aux algorithmes cités dans notre projet vu leur grande efficacité face aux problèmes de la coloration.

Abstract

Coloring of graphs is a classic problem in graph theory. In this thesis, we are interested in tackling one of the most famous subjects of graph theory : the coloring of vertices. This involves coloring the vertices of a graph so that two adjacent vertices do not have the same color. In particular, we show some vertex coloring techniques used in this setting in order to obtain the best results. A particular interest is brought to the algorithms cited in our project given their great efficiency in the face of coloring problems.