UNIVERSITE SIDI MOHAMED BEN ABDELLAH

FACULTE DES SCIENCES ET TECHNIQUES

Licence Sciences et Techniques (LST)

CALCUL SCIENTIFIQUE ET APPLICATIONS

MEMOIRE DE FIN D'ETUDES

Pour l'obtention du Diplôme de Licence Sciences et Techniques

Application de la coloration des graphes aux problèmes de la gestion des examens

Présenté par :

- M^{lle} BOULEFT yousra

Encadré par :

- Pr. EL HILALI ALAOUI Ahmed

Soutenu Le 13 Juin 2013 devant le jury composé de:

- Pr BENCHEIKH Ghizlane
- Pr EL HILALI ALAOUI Ahmed
- Pr EL KHOMSSI Mohammed
- Pr EZZAKI Fatima
- Pr HILALI Abdelmajid

Stage effectué à la FST

Année Universitaire 2012 / 2013

FACULTE DES SCIENCES ET TECHNIQUES FES – SAISS

B.P. 2202 – Route d'Imouzzer – FES

Remerciement

Ce mémoire n'aurait pas être réalisé sans le soutien et l'encouragement de plusieurs personnes que je tiens à remercier.

Je tiens tout d'abord à remercier mon directeur de mémoire Monsieur Ahmed

EL HILALI ALAOUI, professeur à la Faculté des Sciences et Techniques de Fès, pour m'avoir encadré, je tiens à le remercier pour toute sa disponibilité, sa rigueur scientifique, ses précieux conseils.

Mes vifs remerciements s'adressent à Madame EZZAKI Fatima, responsable de notre licence, Madame Ghizlane BENCHEIKH, professeur à la Faculté des Sciences d'économie de Meknès, et aux professeurs Abdelmajid HILALI, et EL KHOMSSI Mohammed, d'avoir accepté à ce jury.

Je ne passerai pas cette occasion sans remercier tous les professeurs de la FST de Fès et particulièrement les professeurs du département des mathématiques.

Je voudrais exprimer mes sincères remerciements à tous mes amis

Je tiens à remercier Ma famille pour m'avoir toujours encouragée dans mes démarches, je remercie tout spécialement ma mère pour la confiance qu'elle m'a accordée, je remercie mon père pour ses conseils judicieux. Mes remerciements chaleureux s'adressent à ma sœur et mes frères pour leur intérêt à ce que je fais.

Table des matières

Introduction générale. 6

Chapitre 1 : Théorie de graphe et optimisation combinatoire 7

1.1 -Introduction 7

1.2 Généralités sur les graphes 7

1.2.1 Définitions et terminologie 8

1.2.2 Représentation d'un graphe 10

La seconde idée permettant une représentation matricielle d'un graphe exploite la relation d'incidence entre arêtes et sommets. 11

1.3 L'optimisation combinatoire 12

1.3.1 Préliminaire 13

C'est l'ensemble des problèmes pour lesquels il existe un algorithme de résolution en un temps polynomiale 14

b)-La classe NP 14

1.3.2 Méthodes de résolution 14

Chapitre 2 : Problème de gestion des examens 16

2.1 Introduction 16

2.2 Problème d'emploi du temps des examens 16

2.3 Etat de l'art 18

Problèmes de l'université de Toronto 18

2.4 Problématique 20

2.5 Représentation des solutions 20

2.5.1 Représentation graphique 20

2.5.2 Représentation matricielle 21

Chapitre 3 : Coloration des graphes 22

3.1 Introduction 22

3.2 Coloration d'un graphe 23

3.2.1 Ensemble indépendant 23

3.2.2 Graphe complet 23

3.2.3 Nombre chromatique 23

3.3 Problème d'ensemble indépendant maximal 24

3.3.1 Relaxation 24

3.3.2 Relaxation Surrogate 25
3.3.3 Problème d'ensemble indépendant maximal 25
3.3.4 Formulation du problème d'ensemble indépendant maximal 25
3.3.5 Formulation Mathématique 26
3.3.6 Amélioration de la solution de la contrainte surrogate 27
3.3.7 Amélioration par une méthode basée sur le multiplicateur w de la contrainte surrogate 31
Chapitre 4 : Application à l'Organisation des examens d'un 33
semestre à la FST de Fès 33
4.1 Introduction 33
4.2 Résolution du problème d'organisation des examens 33
4.2.1 Organisation des examens d'un semestre à la FST de Fès 33
4.2.2 Représentation matricielle 38
Conclusion générale 46
Bibliographie 47

Introduction générale

Les réformes pédagogiques dans les différentes universités mondiales en général et dans les universités marocaines en particulier confrontent des problèmes majeurs au niveau de la gestion des emplois du temps des examens.

Les deux problèmes essentiels qui se posent sont :

1) Un étudiant inscrit dans deux modules différents a le droit de passer les deux examens, comment donc organiser ces deux examens de tel sorte qu'ils ne se passent pas en même temps ?
2) Pour des raisons pédagogiques comment éloigner le maximum possible entre deux examens qui ont des étudiants en commun ?

L'objectif de ce mémoire est de répondre à la première question et ceci en utilisant la méthode de la coloration des graphes.

Le mémoire est organisé en quatre chapitres :

Dans un *premier chapitre,* nous allons rappeler quelques éléments de Théorie de Graphes qui nous seront nécessaire dans la suite. Nous préciserons les définitions, les notations, et nous introduisons les notions préliminaires liées à l'optimisation combinatoire.

Le problème étudié, présenté dans le *deuxième chapitre,* est le problème de planification d'emploi des temps des examens, ce problème qui fait partie du grand problème de la planification dans les établissements d'enseignement est devenu un problème majeur au Maroc particulièrement à partir de l'année universitaire 2003-2004.

Nous abordons dans le *troisième chapitre* le problème de coloration de graphe, et nous essayons de trouver l'ensemble indépendant maximal (MIS), pour cela nous proposons une approche de résolution pour ce problème basée sur une relaxation de la contrainte surrogate.

Dans le *quatrième chapitre* de ce mémoire nous étudions l'organisation des examens dans un semestre à la FST de Fès, et nous essayons de résoudre ce problème on utilisant les propriétés de la coloration des graphes.

Chapitre 1 : Théorie de graphe et optimisation combinatoire

1.1 -Introduction

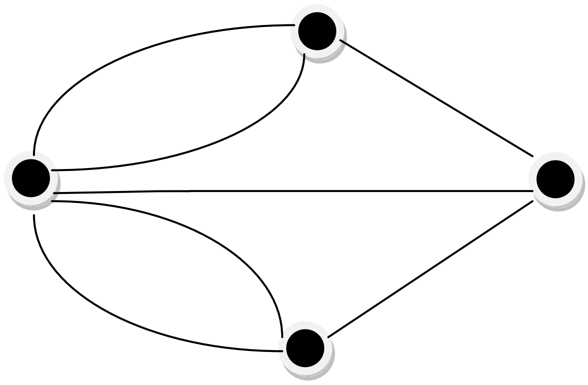
La théorie des graphes présente un domaine des mathématiques qui permet de résoudre efficacement une grande variété de problèmes pratiques en les ramenant à des configurations simples à l'aide de points et de liaisons entre ces points. Elle est aussi l'un des instruments les plus courants et les plus pertinents pour résoudre des problèmes discrets de la recherche opérationnelle (RO).

Plusieurs problèmes de différents domaines peuvent être modélisés sous forme d'un graphe

- Différentes tâches exécutées par une ménagère lors de la préparation d'un déjeuner ;
- Trafic routier ;
- Expédition du pétrole brut depuis les régions productrice jusqu'à la raffinerie des régions consommatrices ;
- Réseaux de voies ferrées ;
- Fils de téléphone ;
- Distribution de marchandises (voyageur de commerce)
- Chimie : Modélisation de structures de molécules.

1.2 Généralités sur les graphes

La théorie des graphes est née en 1736 quand Euler démontra qu'il était impossible de traverser chacun des sept ponts de la ville de Königsberg (Kaliningrad) une fois exactement et de revenir au point de départ.



Le graphe des sept ponts de la ville de Königsberg

1.2.1 Définitions et terminologie

- **Définition d'un graphe :**

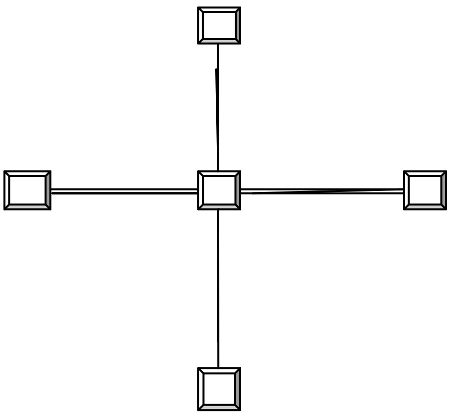
Un graphe G est constitué de deux ensembles :

- Un ensemble X de points appelés **sommets**

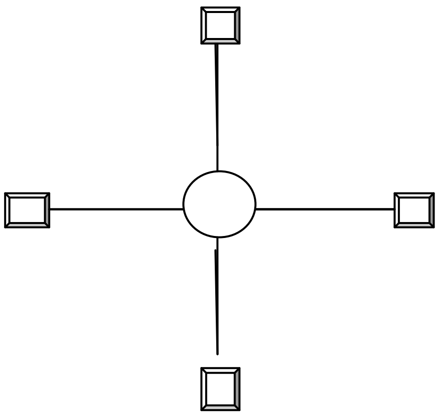
- Un ensemble U de **lignes** reliant chacune deux sommets

On note un graphe par **G = (X, U).** Pour tout (x, y) ∈ X ou x, y ∈ U, x est dite extrémité initiale et y extrémité finale. Le nombre de sommets est appelé ordre du graphe.

Exemples de graphe



étoile

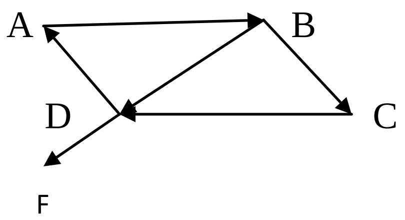


anneau

- **Graphe orienté**

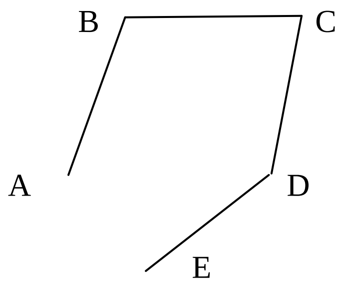
Un graphe est dit orienté, ou bien direct, est un graphe dont on peut distinguer entre les liens (x, y) et (y, x), c'est-à-dire (x, y) ≠ (y, x), et on appelle le lien un arc.

y est un successeur de x tandis que x est un prédécesseur de y



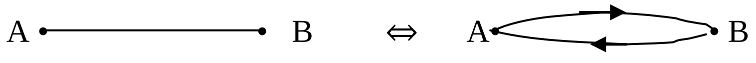
- **Graphe non orienté**

Un graphe est dit non orienté ou bien indirect si la précision de sens de lien (x,y) et la Distinction entre extrémité initiale et extrémité terminale ne jouent aucun rôle. Tout élément (x, y) ∈ U est une arête.



<u>Remarque</u>

- Tout graphe orienté peut être transformé en un graphe non orienté en supprimant l'orientation des arcs.
- Tout graphe non orienté peut être transformé en un graphe orienté en remplaçant chaque arête par deux arcs en sens inverse.
- On note un arc par (A, B) ou $\overrightarrow{AB}$



- Soit X = {x_1, x_2,…….. x_n} un ensemble de sommets.

On peut définir un graphe par la relation binaire R : x_i R x_j ⇔ (x_i, x_j) est un arc du graphe.

Définition

Si x_i R x_j , on dit que les sommets x_i et x_j sont adjacents.

Définition équivalente

x_i et x_j sont adjacents s'il y a un arc qui les relis.

1.2.2 Représentation d'un graphe

Un graphe peut avoir plusieurs représentations, nous utilisons dans ce mémoire la représentation matricielle

1.2.2.1 Matrice d'adjacence

Les outils classiques d'algèbre linéaire peuvent également être utilisés pour coder les graphes. La première idée consiste à considérer chaque arc comme un lien entre deux sommets.

Définition :

Considérons un graphe G = (X, A) comportant n sommets. La matrice d'adjacence de G est égale à la matrice U = (u_{ij}) de dimension n × n telle que :

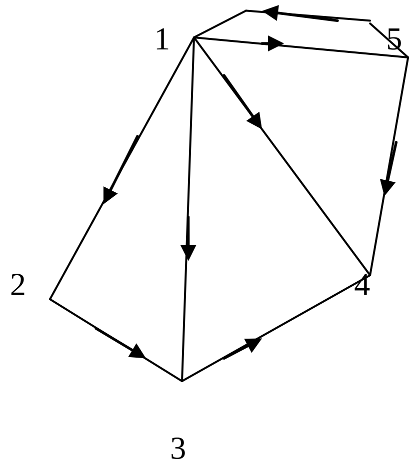
$$u_{ij} = \begin{cases} 1\ si\ (i,j\ )\ \in\ A\ (c'est - à - dire\ (i,j\ )\ est\ une\ arête) \\ 0\ sinon \end{cases}$$

Une telle matrice, ne contenant que des "0" et des "1" est appelée, de manière générale, une matrice booléenne.

Un graphe orienté quelconque a une matrice d'adjacence quelconque, alors qu'un graphe non orienté possède une matrice d'adjacence symétrique. L'absence de boucle se traduit par une diagonale nulle.

Exemple

Considérons le graphe suivant :



La matrice d'adjacence du graphe est la suivante :

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

1.2.2.2 Matrice d'incidence

La seconde idée permettant une représentation matricielle d'un graphe exploite la relation d'incidence entre arêtes et sommets.

Définition :

Considérons un graphe orienté sans boucle G = (X, A) comportant n sommets $x_1, \ldots, x_n$ et m arêtes $a_1, \ldots, a_m$. On appelle matrice d'incidence (aux arcs) de G la matrice M = (m_{ij}) de dimension n × m telle que :

$$m_{ij} = \begin{cases} 1 \text{ si xi est l'extrémité initiale de } a_i \\ -1 \text{ si xi est l'extrémité terminale de } a_j \\ 0 \text{ si xi n'est pas une extrémité de } a_j \end{cases}$$

Pour un graphe non orienté sans boucle, la matrice d'incidence (aux arêtes) est définie par :

$$m_{ij} = \begin{cases} 1 \text{ si xi est une extrémité de } a_j \\ 0 \; sinon \end{cases}$$

La matrice d'incidence du graphe de l'exemple précédente s'´ecrit sous la forme suivante :

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

1.3 L'optimisation combinatoire

L'optimisation combinatoire occupe une place très importante en recherche opérationnelle et en informatique. De nombreuses applications pouvant être modélisées sous la forme d'un problème d'optimisation combinatoire telles que le problème du voyageur de commerce, l'ordonnancement de tâches, le problème de la coloration de graphe, etc. L'optimisation combinatoire comprend un ensemble fini de solutions, où chaque solution doit satisfaire un ensemble de contraintes relatives à la nature du problème, et une fonction objectif pour évaluer chaque solution trouvée. La solution optimale est celle dont la valeur de l'objectif est la plus petite (resp. grande) dans le cas de minimisation (resp. maximisation) parmi l'ensemble de solutions.

Un problème d'optimisation combinatoire peut être défini par :

Vecteur de variables $x = (x_1 ,\ldots\ldots\ldots ,x_n)$

Domaine des variables $D = (D_1, D2,\ldots\ldots,D_n)$, où les $(D_i)_{i=1,\ldots,n}$ sont des ensembles finis

Ensemble de contraintes

Une fonction objectif f à minimiser ou à maximiser

Ensemble de toutes les solutions réalisable possibles est S = {(x_1, x_2…,x_n) ∈ D / x satisfait toutes les contraintes }, l'ensemble S et aussi appelé un espace de recherche.

La résolution des problèmes combinatoires est assez délicate puisque le nombre fini de solutions réalisables croît généralement avec la taille du problème, ainsi que sa complexité. Cela à poussé les chercheurs à développer de nombreuses méthodes de résolution en recherche opérationnelle (RO) et en intelligence artificielle (IA). Ces approches de résolution peuvent être classées en deux catégories : les méthodes exactes et les méthodes approchées.

Les méthodes exactes ont permis de trouver des solutions optimales pour des problèmes de taille raisonnable et rencontrent généralement des difficultés face aux applications de taille importante. En revanche les méthodes approchées ne garantissent pas de trouver une solution exactes, mais seulement une approximation, mais ils peuvent donner des solutions même pour les problèmes de tailles assez grandes.

1.3.1 Préliminaire

1.3.1.1 Optimisation combinatoire

L'optimisation combinatoire ou optimisation discrète, est une branche de l'optimisation liée essentiellement aux domaines suivants :

- mathématique appliquées
- informatique
- la recherche opérationnelle
- l'algorithmique et la théorie de la complexité.

1.3.1.2 Problème d'optimisation combinatoire

Un problème d'optimisation combinatoire consiste à trouver la meilleure solution dans un ensemble discret dit *ensemble des solutions réalisable.* Cet ensemble est souvent fini mais de cardinalité peut être très grande et il est décrit de manière implicite, c'est-à-dire par une liste de contraintes doivent satisfaire les solutions réalisables.

1.3.1.3 La complexité

La complexité analyse l'espace mémoire ou le temps nécessaire pour obtenir une solution.

La théorie de complexité conduit à distinguer entre la complexité des problèmes et la complexité des algorithmes utilisés pur les résoudre.

Complexité d'un problème :

On appelle complexité d'un problème la complexité en temps du meilleur algorithme permettant de le résoudre.

Un problème peut être résolu par différents algorithmes et en général on n'est pas sûr d'avoir trouvé l'algorithme de meilleure complexité. L'analyse des problèmes consiste à les rangés dans des classes de problèmes de complexité.

a)- La classe P

C'est l'ensemble des problèmes pour lesquels il existe un algorithme de résolution en un temps polynomiale

b)-La classe NP

C'est l'ensemble des problèmes pour lesquels il existe un algorithme non déterministe pouvant les résoudre en un temps polynomial.

c)-La classe NP-difficile

Un problème est NP-difficile si l'existence d'un algorithme de complexité polynomiale pour le résoudre implique P = NP.

d)-La classe NP-complet

Un problème est NP-complet s'il est NP-difficile et s'il appartient à la classe NP.

1.3.2 Méthodes de résolution

La résolution du problème d'optimisation combinatoire consiste à trouver une solution optimale dans un ensemble discret et généralement fini, ce qui est une trivialité du point de vue mathématique. Du point de vue informatique et pratique, le temps de recherche la solution optimale est un facteur très important et c'est pour cela les problèmes d'optimisation combinatoire sont réputés si difficiles. On distingue deux grandes classes de méthodes de résolution : les Méthodes exactes et les méthodes approchées

1.3.2.1 Méthodes exactes

Le principe essentiel d'une méthode exacte consiste généralement à énumérer, souvent de manière implicite, l'ensemble des solutions de l'espace de recherche.

Les méthodes exactes ont permis de trouver des solutions optimales pour des problèmes de taille raisonnable. Malgré les progrès réalisés, notamment en matière de la programmation linéaire en nombres entiers, le temps de calcul nécessaire pour trouver une solution, risque d'augmenter exponentiellement avec la taille du problème rencontrent généralement des difficultés face aux applications de taille importante.

1.3.2.2 Méthodes approchées

Les méthodes approchées constituent une alternative très intéressante pour traiter les problèmes d'optimisation de grande taille, leur but est de trouver une solution de bonne qualité en un temps de calcul raisonnable sans garantir l'optimalité de la solution obtenue. Ces méthodes sont fondées principalement sur divers heuristiques, souvent spécifiques à un type de problème donné.

L'objectif « heuristique », signifie « qui facilite la découverte, qui a une utilité dans la recherche (scientifique ou autre) ».

Les méthodes heuristiques sont des critères, des principes ou des méthodes permettant de déterminer parmi plusieurs solutions, celle qui promit d'être la plus efficace pour atteindre un but.

Chapitre 2 : Problème de gestion des examens

2.1 Introduction

Parmi les problèmes d'ordonnancement les plus étudiés dans la littérature, nous trouvons le problème d'emploi du temps dans les établissements d'enseignement. L'objectif est d'affecter un ensemble d'entités (cours, examens, ……etc) à un nombre limité de ressources (tranches horaire, salle,……etc).

La recherche dans le problème d'emploi du temps a commencé par des techniques séquentielles simples pendant les années 60. Les techniques basées sur les contraintes sont apparues plus tard et jouent toujours un rôle significatif à nos jours.

Le problème d'emploie du temps est un problème très étudiés dans la littérature. En effet il a fait le sujet de plusieurs travails de chercheurs dans ces derinières années, nous citons par exemple le travail de Ibtissam Dkhssi et Rachida Abounasser.

Le problème d'emploi du temps d'université qui a eu un grand intérêt le long des dernières décennies, est divisé dans la littérature en deux problématiques :

1) le problème d'emploi des temps de cours.

2) le problème d'emploi des temps des examens.

Ces deux types de problème se ressemblent, mais il y a certaine nombre de points de différences entre eux.

Dans ce chapitre nous étudions le problème d'emploi du temps des examens connu aussi sous le nom de gestion des examens.

2.2 Problème d'emploi du temps des examens

Le problème d'emploi du temps des examens est concerné par l'attribution d'un ensemble $E = \{e_1,\ldots,e_n\}$ des examens dans un ensemble $T = \{T_1,\ldots,T_M\}$ de créneaux horaires sous un certain nombre de contraintes fortes et faibles. Les contraintes fortes doivent être satisfaites afin de produire un planning réalisable, tandis que les contraintes faibles doivent être réduites au minimum.

Les contraintes fortes du problème d'emploi du temps les plus fréquents dans la littérature sont :

1) aucun étudiant ne doit avoir plus d'un examen dans la même tranche horaire.

2) la capacité de l'établissement ne doit pas être dépassée à n'importe quelle tranche horaire donnée.

3) certaines paires d'examens peuvent subir des contraintes de priorité ou de simultanéité. Un ensemble de contraintes fortes moins communes peuvent souvent être considérées dans différents établissements selon leurs exigences particulières.

Les contraintes faibles varient bien plus que les contraintes fortes parce qu'elles définissent essentiellement la qualité de l'emploi du temps afin de donner une mesure de comparaison parmi les emplois du temps réalisable. Afin d'utiliser une fonction objectif simple à optimiser, ces contraintes faibles doivent toutes être pondérées selon leurs importance relative pour le planning produit. Citons quelques exemples des contraintes faibles les plus confrontées dans la littérature :

1) Affectation des créneaux horaire : il peut être désiré qu'un examen soit planifié dans une tranche horaire particulière.

2) Contraintes de précédence entre les événements : un examen doit être planifié avant ou après d'autres (selon le problème, cette contrainte peut être également une contrainte forte).

3) Distribution des événements le long de l'horizon de planification : les étudiants ne doivent pas avoir des examens en périodes consécutives et de préférence séparées au maximum.

4) affectation des ressources : il peut être désiré qu'un examen soit affecté dans une salle particulière.

Certain établissement considèrent également la contrainte concernant le placement de plusieurs examens dans une même salle u la contrainte qui permet que les étudiants inscrits dans un même examen peuvent être placés dans plusieurs salles. Le nombre de tranches horaire dans l'horizon de planification peut être fixé à priori ou il peut intervenir comme objectif à minimiser. Cette contrainte représente un conflit avec la contrainte qui consiste à étaler les examens le long de l'horizon de planification. D'autres contraintes faibles peuvent apparaitre selon la spécifité de l'établissement par exemple une faculté des sciences n'a pas les même contrainte que une FST ou qu'une école d'ingénieure.

2.3 Etat de l'art

L'accroissement de l'intérêt de recherches pour les problèmes d'emploi du temps des examens a mené les chercheurs (Carter et al. [CARTER 96]) à la création d'un ensemble de problèmes qui ont été largement étudiés dans la littérature. Les problèmes établis, sont définis suivant certaines mesures définissant les variantes, qui fournissent une manière pour établir des comparaisons scientifiques. Cependant, il y a eu une manière pour établir des comparaisons scientifiques. Cependant, il y a eu une certaine confusion dans la littérature due à l'apparition de deux versions différentes de huit instances de ces problèmes (de l'université de Toronto). Nous essayons de présenter les différents problèmes apparus dans la littérature :

Problèmes de l'université de Toronto

Carter et al. [CARTER 96] ont présenté un ensemble d'instances du problème d'emploi du temps des examens réels de trois lycée canadiens, de cinq universités canadiennes, d'une université américaine, d'une université britannique et d'une université en Arabie Saoudite. Ces problèmes ont été largement testés dans la littérature par différentes approches et sont considérées comme des références.

Deux variantes des objectifs ont été définies :

1- la minimisation du nombre de tranches horaires requis pour le problème (appelée Torontoa)

2- la minimisation du coût moyen par étudient (appelé Torontob) pour le premier objectif, le but est de construire un planning faisable dans un horizon de longueur minimale. Pour le deuxième objectif, une fonction d'évaluation a été définie pour calculer le coût des plannings produits. Pour un étudiant ayant deux examens séparés par s tranches horaires, le coût associé est défini par la valeur de proximité w_s où $w_1 = 16$, $w_2 = 8$, $w_3 = 4$ $w_4 = 2$ et $w_5 =1$. Le but est de séparer deux examens consécutifs au maximum dans l'horizon de planification fixé.

Les auteurs ont également présenté sept autres instances du monde réel avec d'autres contraintes latérales (par exemple : la capacité maximal des salles par tranche horaire, le nombre maximal d'examen par tranche horaire,….etc)

Burke et al. [BURKE 96] ont modifié l'objectif des variantes du monde réel présentées dans [CARTER 96] en considèrent la contrainte de la capacité maximal des salles par tranches horaires dans les problèmes nt été distinguées en plaçant trois horaires par jour de lundi au vendredi et une tranche horaire samedi. L'objectif est de minimiser le nombre des étudiants passant deux examens consécutif dans le même jour et durant la nuit. Ces deux variantes sont appelées Torontoc et Tonrontod. Terashima-Marin et al. [TERASHIMA-MARIN 99] ont modifié l'ensemble de données en affectant à chaque instance du problème un nombre de tranches horaires estimé et à chaque tranche horaire la capacité maximal des salles. Cette variante est appelée Torontoe

Variantes	Objectifs
Toronto a	Minimisation du nombre des tranches horaires nécessaires
Toronto b	Maximiser de la séparation entre les examens dans un horizon de planification fixé
Toronto c	Minimiser des nombres des étudiants ayant deux examen dans le même
Toronto d	Pareil que dans Toronto c : minimisation des étudiants passant deux examens durant la nuit
Toronto e	Minimisation du nombre des étudiants passant deux examens dans deux tranches horaires adjacents

Problème de l'université de Nottingham

En plus des objectifs étudiés dans les problèmes de Tonronto, Burke et al. [BURKE 98a] ont introduit l'objectif de minimisation du nombre des examens consécutifs durant la nuit. Ces nouvelles variantes des problèmes d'emploi du temps des examens ont été appelés « Nottingham a » et « Nottingham b ».

Problème de l'université de Melbourne

Merlot et al. [MERLOT 03] ont présenté un ensemble de données du problème d'emploi du temps des examens de l'université de Melbourne. Sont objectif est de minimiser le nombre d'examens adjacents dans le même jour.

2.4 Problématique

Les responsables des services des examens confrontent des problèmes au niveau d'organisation les horaires des examens.

Comment organiser les examens dont le minimum de chrénos possible et sous la condition qu'aucun étudiant n'ait à passer deux épreuves en même temps ?

2.5 Représentation des solutions

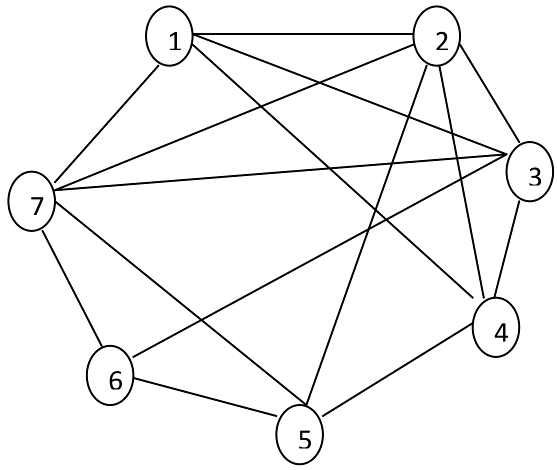
2.5.1 Représentation graphique

La forme la plus simple d'un problème de gestion des examens est équivalente à celle de la coloration des graphes, Ce problème consiste à colorier les sommets d'un graphe G de façon à ce que deux sommets reliés par un arc ne soient jamais de la même couleur. L'objectif de ce problème est la minimisation du nombre de couleurs. Par analogie, en posant les examens comme des sommets et les couleurs comme des périodes, chaque examen doit être assigné à une seule période. De plus, les examens qui ont des élèves en commun, qui sont reliés par un arc, ne doivent pas être assignés à une même période

Exemple :

Une université 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 que les paires de cours suivantes ont des étudiants communs : 1 et 2, 1 et 3, 1 et 4, 1 et 7, 2 et 3, 2 et 4, 2 et 5, 2 et 7, 3 et 4, 3 et 6, 3 et 7, 4 et 5, 4 et 6, 5 et 6, 5 et 7 et 6 et 7.

La présentation graphique de ce problème est donnée par :



2.5.2 Représentation matricielle

Dans le problème d'emploi du temps des examens, nous peuvent transformer le problème sous forme d'une matrice carré symétrique M où chaque élément $m_{ij} = 1$ si l'examen i est en conflit avec l'examen j c'est-à-dire que i et j ont des étudiants en commun, et $m_{ij} = 0$ sinon.

Exemple

Nous considérons le même problème de l'exemple précédente la représentation matricielle est donnée par

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

Où M est une matrice carré symétrique contient 1 et 0,1 si l'examen en i et l'examen j ont des étudiant en commun, 0 sinon

Chapitre 3 : Coloration des graphes

3.1 Introduction

En recherche opérationnelle, le problème de la coloration de graphe fait partie des problèmes de référence. Il consiste à minimiser le nombre de couleurs des sommets d'un graphe donné de telle sorte que deux sommets liés par une arête n'aient pas la même couleur.

La coloration de graphes remonte à 1852, lorsque Francis Lutherie a présenté sa conjecture des quatre couleurs, « chaque carte peut être colorée avec quatre couleurs de sorte que les pays voisins qui partagent une frontière commune reçoivent des couleurs différentes ». Depuis, la coloration de graphes est devenu l'un des domaines les plus étudiés de la théorie de graphes.

Pour colorier un graphe tel qu'il n'existe pas deux sommets adjacents ayant la même couleur est le principe problème d'étude en coloration de graphe. Le problème de la coloration de graphe est parmi les problèmes d'optimisation combinatoires les plus étudiés en informatique et en mathématique. La coloration de graphes est un sujet très actif de la théorie des graphes, il a fait l'objet de plusieurs applications et problèmes réels dans de nombreux domaines tels que, la télécommunication, la bioinformatique et l'internet. Est lié à de nombreuses applications traditionnelles dans des domaines variés, comme le problème d'emploi du temps, l'ordonnancement des tache,…etc. La coloration de graphes a été aussi appliquée pour modéliser et résoudre des problèmes en mathématiques et statistique.

Dans ce chapitre nous essayons de donner quelque propriété sur le problème de coloration.

Ensuite nous introduisons les méthodes de relaxation, avec en particulier la relaxation surrogate, et nous proposons une heuristique pour la résolution du problème de l'ensemble indépendant maximal (MIS) basée sur la relaxation surrogate de la formulation en programmation entière du problème (MIS).

3.2 Coloration d'un graphe

3.2.1 Ensemble indépendant

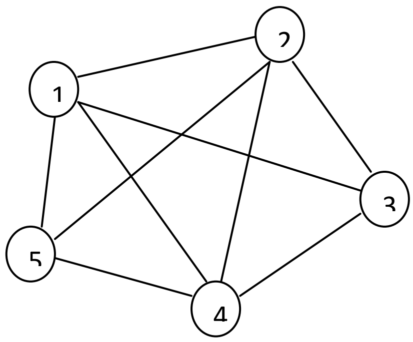
Un ensemble indépendant ou stable d'un graphe G = (V, E) est un sous-ensemble S incluse dans V de sommets deux à deux non reliés par une arête. On note par α(G) (nombre de stabilité) le cardinal maximum d'un stable de G. Notons qu'un stable est le complémentaire d'une clique et que

$\omega(G) = \alpha(G^c)$ ainsi que $\alpha(G) = \omega(G^c)$, ou ω(G) est la taille de la clique maximale de G.

3.2.2 Graphe complet

Un graphe complet ou clique est un graphe dont les sommets sont reliés deux à deux entre eux par une arête.

Exemple d'un graphe complet :



3.2.3 Nombre chromatique

Soit k un entier. On appelle k-coloration du graphe G = (S, A) toute partition de S en k ensembles A_1 ,………A_k tels que pour tout $1 \leq i \leq k$, $G[A_i]$ est un stable. Les ensembles A_i sont appelés les couleurs de G. si $x \in A_i$, on dit qu'on a donné à x la couleur i. notons qu'un graphe à n sommets possède une n-coloration.

On appelle nombre chromatique de G le plus petit entier k tel que G admet une k-coloration. On note χ(G) le nombre chromatique de G.une coloration de G avec χ(G) couleurs est dite optimal. On note ω(G) (resp., α(G)) la taille d'une clique (resp. d'un stable) de G de taille maximal ; autrement dit $\omega(G) := \max\{| C | : C \text{ est une clique de } G\}$ (resp., $\alpha(G) = \max\{| C| : C \text{ est un stable de } G\}$), où C désigne le nombre d'éléments de |C|.

$$\alpha(G).\ \chi(G) \geq n(G) \Rightarrow \quad \chi(G) \geq \frac{n(G)}{\alpha(G)}$$

n(G) étant le nombre de sommets du graphe.

On a aussi :

$$\chi(G) \leq \deg G + 1$$

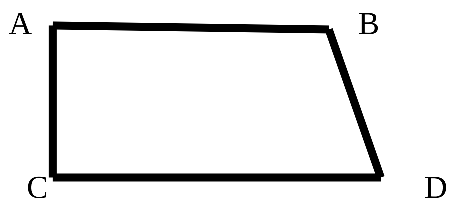
Où deg G est le plus grand degré d'un sommet.

D'où : $\deg G + 1 \geq \chi(G) \geq \frac{n(G)}{\alpha(G)}$

Définition :

L'indice chromatique noté q(G) d'un graphe G est le nombre minimal decouleur snécessaires à la coloration des arêtes de G.

Exemple



3.3 Problème d'ensemble indépendant maximal

3.3.1 Relaxation

Un problème d'optimisation P : *max {f(x), x ∈ S_1}* est une relaxation du problème PR : *max {f(x), x ∈ S_2}* si S_2 incluse dans S_1, une relaxation d'un problème est une formulation qui contient un ensemble de solutions réalisables plus grand que celui du problème original, et donne une borne supérieure du problème PR. Dans le cas d'un problème de minimisation, la relaxation fournit une borne inferieure. Une bonne relaxation fournit une solution optimale proche de l'optimum du problème original.

De manière générale, une relaxation devrait satisfaire les deux conditions suivantes :

- Avoir une structure semblable à celle du problème original
- Plus facile à résoudre que le problème original, de manière à fournir de bonnes bornes.

3.3.2 Relaxation Surrogate

La relaxation surrogte ou bien agrégée a été initialement présentée comme un moyen d'améliorer les règles de décision et borner l'information dans des algorithmes de la programmation entière (GLOVER 1965). La relaxation surrogate consiste à combiner l'ensemble des contraintes en une seule de manière suivante :

$$(IP)\begin{cases} \max c^{t}.x \\ A.x \leq b \\ x \in \{0, 1\}^{n} \end{cases}$$

Où $x = (x_1, \ldots\ldots, x_n)^t$, $c = (c_1, \ldots\ldots, c_n)$, $b = (b_1, \ldots\ldots, b_m)^t$, $A = a_{ij}$ pour $1 \leq i \leq m$ et $1 \leq j \leq n$.

La relaxation surrogate est donnée par :

$$(SC)\begin{cases} \max c^{t}.x \\ w^{t}A.\, x \leq w^{t}b \\ x \in \{0,1\}^{n} \end{cases}$$

Avec $0 \leq w = (w_1, \ldots\ldots w_n)^t$

3.3.3 Problème d'ensemble indépendant maximal

Le problème d'ensemble indépendant maximal est l'un des problèmes centraux de l'optimisation, et un des problèmes classiques montrés pour être NP-difficile, son intérêt augmente en raison de son équivalence au problème de la clique maximale et à celui de couverture. Il est efficace pour la résolution de nombreuses applications pratiques en informatique, en recherche opérationnelle, ou dans le domaine de l'ingénierie, telles que le problème d'ordonnancement, le problème de coloration de graphes, etc.

3.3.4 Formulation du problème d'ensemble indépendant maximal

Nous considérons un graphe non-orienté G = (V, E), où V = {1,……..n} dénote l'ensemble de sommets du graphe G et E dénote l'ensemble des arêtes. ∀ i ∈ V on a

$$\text{nodestar}(i) = \{j : \{i,j\} \in E\}$$

d_i = card(nodestar(i))

$d_0 = |E|$ = nombre des arêtes.

La formulation usuelle de la programmation mathématique pour le problème d'ensemble indépendant maximal associe une variable binaire xi à chaque sommet

$i \in V$ telle que

$$x_i = \begin{cases} 1 & \text{si le sommet est choisi comme élément de l'ensemble indépendant maximal} \\ 0 & \text{sinon} \end{cases}$$

3.3.5 Formulation Mathématique

<u>L'objectif</u>

Maximiser le nombre de sommets d'un stable

<u>Variable</u>

x_i sommet d'un graphe G

<u>Contrainte</u>

L'ensemble indépendant ne contient que les sommets non adjacents c.-à-d. qu'il que soit un arc dont les extrémités sont i, j alors forcement au plus l'un de ce sommets sera dans notre ensemble indépendant maximal.

D'où la formule mathématique est donné par :

$$x_i + x_j \leq 1 \quad (i, j) \in E$$

Donc le problème peut être exprimé sous forme d'un problème de la programmation en nombres entiers comme suit :

$$(IP) \begin{cases} \max x_0 = \sum (x_i : i \in V) \\ x_i + x_j \leq 1, \ \{i, j\} \in E \\ x_i \text{ binaire}, \ i \in V \end{cases}$$

Les variables de décision correspondent aux opération faites sur un graphe tels qu'ajouter ou supprimer des sommets et des arêtes. De même, des implications de la décision telles que forces des variables particulières à prendre la valeur 0 ou 1 et faire exprimer des contraintes redondants peuvent également être reflétées par des changements correspondants à la structure de graphe. Dans le cas actuel, selon le problème considéré, le problème (IP) correspondent aux réductions simples suivantes du graphe :

Poser $x_i = 1$ dans le problème (IP), force $x_j = 0$ pour toutes les variables x_j telles que

$x_i + x_j \leq 1$ ce qui correspond dans le graphe

3.3.6 Amélioration de la solution de la contrainte surrogate

Le problème de l'ensemble indépendant maximal est exprimé comme suit :

$$(IP) \begin{cases} \max x_0 = \sum (x_i : i \in V) \\ x_i + x_j \leq 1, \ \{i, j\} \in E \\ x_i \text{ binaire}, \ i \in V \end{cases}$$

pour produire rapidement une solution approchée de ce type de problème, nous utilisons l'heuristique de la contrainte surrogate, obtenue on remplaçant l'ensemble des contraintes par une seule contrainte combinaison linéaire des originales. Les poids d_i sont obtenus en additionnant le nombre de fois où xi apparait dans les inégalités, et d_0 est le nombre de toutes les inégalités pour le problème (IP).

Pour le problème (IP), nous faisons une simple sommation de toutes les contraintes comme suit :

$$\sum (d_i \, x_i : i \in V) \leq d_0$$

Le problème surrogate associé à (IP) est :

$$(SC) \begin{cases} \max x_0 = \sum (x_i : i \in V) \\ \sum (d_i \, x_i : i \in V) \leq d_0 \\ x_i \text{ binaire}, \ i \in V \end{cases}$$

la valeur de d_0 dans les contraintes du problème surrogate (SC) est égale au nombre de contraintes d'inégalité, et d_i est égale à la somme des coefficients d'unité pour les variables x_i qui apparaissent dans ces contraintes pour avoir une meilleure solution nous pouvons inclure certaines contraintes de (IP) dans le problème (SC) de la manière suivante :

$$(SC_1) \begin{cases} \max x_0 = \sum (x_i : i \in V) \\ \sum (d_i x_i : i \in V) \leq d_0 \\ x_i + x_j \leq 1, \ \{i, j\} \in E' \\ x_i \text{ binaire}, \ i \in V \end{cases}$$

Où E' contient des arêtes de E formées par des paires de sommets disjoints, et par conséquent chaque variable x_i, $i \in V$, apparaît au plus une fois dans la collection des inégalités.

Sous la condition de croissance des di nous définissons :

$$V' = V - \{j \in V: (i, j) \in E' \text{ et } i < j\}$$

Par conséquent l'ensemble V' est obtenu en abandonnant dans chaque arête $(i, j) \in E'$ les sommets du plus grand coefficient. Alors les contraintes auxiliaires dans E' peuvent être enlevées et le problème est réduit à :

$$(SC_2) \begin{cases} \max x_0 = \sum (x_i : i \in V') \\ \sum (d_i x_i : i \in V') \leq d_0 \\ x_i \text{ binaire}, \ i \in V' \end{cases}$$

Nous remarquons que (SC_2) a la même forme que le problème de la contrainte surrogate original (SC), et puisque V' enlève presque la moitié des coefficients de V, donc la solution est plus restreinte et SC_2 donne généralement une borne x_0 de meilleure qualité que (SC).

Exemple 2.1

Nous considérons un graphe non-orienté dans la figure 2.1, nous essayons d'appliquer l'approche de la contrainte surrogate utilisant les deux transformations (SC_1) et (SC_2) afin d'avoir un ensemble indépendant maximal V'. Le nombre entiers associé à ce graphe peut s'exprimer sous la forme :

$$
(IP)\begin{cases}
\max x_0 = \sum_{i \in V} x_i \\
x_1 + x_2 \leq 1 \\
x_1 + x_3 \leq 1 \\
x_2 + x_3 \leq 1 \\
x_2 + x_5 \leq 1 \\
x_2 + x_6 \leq 1 \\
x_3 + x_5 \leq 1 \\
x_3 + x_6 \leq 1 \\
x_4 + x_5 \leq 1 \\
x_4 + x_6 \leq 1 \\
x_5 + x_6 \leq 1 \\
x_i \text{ binaire, } i \in V
\end{cases}
$$

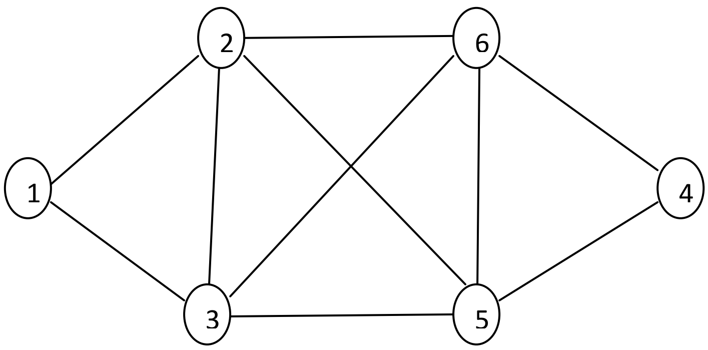


Figure 2.1 Extraction de l'ensemble indépendant maximal

Nous introduisons par la suite l'approche surrogate donnée dans (2.1), ainsi le programme surrogate (SC) associé à (IP) est de la forme :

$$(SC)\begin{cases}\max x_0 = \sum_{i\in V} xi \\ 2x_1 + 4x_2 + 4x_3 + 2x_4 + 4x_5 + 4x_6 \leq 10 \\ x_i \text{ binaire, } i\in V\end{cases}$$

L'ensemble d'arêtes E' ⊂ E constitué de paires de sommets disjoints (les contraintes redondantes) correspond au graphe de la figure 2.1 est E' = {(1, 2), (3,5), (4 ,6)} et donc le problème (SC_1) associé s'écrit sous la forme :

$$(SC_1)\begin{cases}\max x_0 = \sum_{i\in V} x_i \\ 2x_1 + 4x_2 + 4x_3 + 2x_4 + 4x_5 + 4x_6 \leq 10 \\ x_1 + x_2 \leq 1 \\ x_3 + x_5 \leq 1 \\ x_4 + x_6 \leq 1 \\ x_i \text{ binaire, } i \in V\end{cases}$$

Puisque les sommets du graphe sont ordonnés d'une manière croissante suivant leurs degrés, nous éliminons tout sommet j $\in V$ $t$els que (i, j) $\in E'$ et i < j, l'ensemble de sommets enlevé est {2, 6, 5}, par conséquent V'= {1, 3, 4} est l'ensemble de sommets qui sont restés.

Après avoir éliminé toutes les contraintes redondantes du problème (SC_1), nous obtenons donc le problème suivant :

$$(SC_2)\quad\begin{cases}\max x_0 = \sum_{i\in V'} x_i \\ x_1 + x_3 \leq 1 \\ x_i \text{ binaire, } i\in V'\end{cases}$$

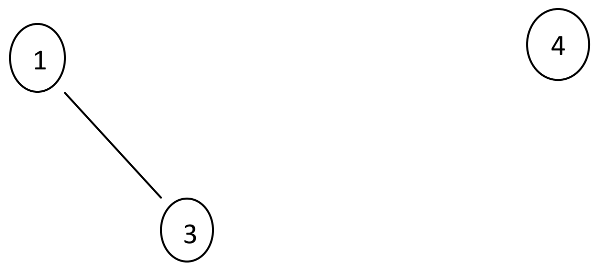


Figure 2.2 Graphe résultant après réduction de variables.

Nous pouvons illustrer cette réduction de variables sous forme d'un graphe de la figure 2.2.

Le problème (SC_2) à la même structure que le problème surrogate (SC), on refait alors les mêmes étapes précédentes on a l'ensemble V' = {1,4}

3.3.7 Amélioration par une méthode basée sur le multiplicateur w de la contrainte surrogate

Pour le cas d'un graphe G = (V, E) non orienté, nous considérons $\omega = (\omega_1,\ldots\ldots,\omega_m)^t$, avec m le nombre d'arêtes. On énonce d'abord le résultat suivant :

Résultat

Soit (SC) la relaxation surrogate du problème (IP), on choisit la variable $x_r = 1$, si $a_{k,j} = 1$ avec j = nodestar (r) alors $w_k = 0$, pour tout $1 \leq j \leq n$ et $1 \leq k \leq m$ (Douiri et El Bernoussi)

Preuve

Pour x_r et j $\in$ nodestar(r) nous posons $\sum_{k=1}^{m} a_{k,j}$ = dj (nombre d'arrêtes reliées au sommet j) nous avons :

$$\omega^t.A.x = \sum_{k=1}^{m} \omega_l \,.\, a_{l,1}.\, x_1 + \ldots\ldots\ldots. + \sum_{l=1}^{m} \omega_l .\, a_{l,n}.\, x_n \qquad (*)$$

Le choix $x_r = 1$ implique $x_j = 0$, donc toutes les arêtes associées à j seront éliminées, autrement dit si $a_{k,j} = 1$ on lui donne la valeur zéro $\forall k \in \{1,\ldots,m\}$,et en utilisant la relation (*) cela s'exprime par $\omega_k = 0$.

Nous utilisant un algorithme de construction d'un ensemble indépendant maximal approché. L'algorithme est donnée par :

1. Poser $\omega = (1,\ldots\ldots..,1)^t$ et V' = Ø
2. Calculer la contrainte surrogate $\omega^t A = \sum_{\omega_{k\neq 0}} (a_k)$
3. Donner i = indice (min $(\omega^t A) : (\omega^t A)_i \neq 0)$, poser $x_i = 1$ et V' = V'∪ $\{i\}$
4. pour tout j ∈ nodestar (i), si $a_{k,j} = 1$ alors $w_k = 0$,si $\sum_{K=1}^{d_0} \omega_K = 0$ stop

   Autrement retourner à l'étape 2

Sur le graphe de l'exemple 2.1, nous appliquons cette l'algorithme pour donner un ensemble indépendant maximal, nous considérons la matrice de graphe $A = (a_{i,j})$ suivante :

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

Pour $\boldsymbol{\omega} = (1,1,1,1,1,1,1,1,1,1)^t$ nous avons SC = $\boldsymbol{\omega}^t A = (2,4,4,2,4,4)$.
indice (min$(\omega^t A) : (\omega^t A)_i \neq 0) = 1$ alors V' = {1}.

nodestar (1) = {2,3} nous cherchons $1 \leq k \leq 12$ tel que $a_{k,2} = 1$ et $a_{k,3} = 1$ donc k ={1,2,3,4,5,6} et $\omega = (0, 0, 0, 0, 0, 0, 0,0, 1, 1, 1)^t$.

Nous recalculons SC = (0, 0, 0, 2, 2, 2) et indice (min $(\omega^t A)$: $(\omega^t A)_i \neq 0) = 4$ donc V'={1,4}.

nodestar (4) = {5,6}, nous identifions les valeurs de k de sorte que $a_{k,5} = 1$ et $a_{k,6} = 1$ ce qui donne k = { 4, 5, 6, 7, 8, 9, 10} alors ω = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0}t

$V \neq \text{nodestar } (1,4) \cup V'$ alors $V' = V' \cup \{ V / \text{ nodestar } (1,4) \cup V \} = \{1,4\}$ correspond à l'ensemble indépendant maximal.

Chapitre 4 : Application à l'Organisation des examens d'un semestre à la FST de Fès

4.1 Introduction

Depuis quelques années le problème de la gestion des examens dans les universités est devenu un problème pour les organisateurs, Vu l'augmentation de nombre de modules à la FST de Fès et vu les réformes pédagogiques qui permettent aux étudiants de s'inscrire dans des modules suivant leurs choix, et vu la capacité des locaux de la FST qui est limité.

Le problème de la gestion des examens est devenu un problème pour les organisateurs,

Dans ce chapitre nous proposons une méthode d'organisation des examens en utilisant la coloration des graphes plus précisément nous proposons une modélisation du problème sous forme d'un graphe non orienté dans les sommets sont les examens, deux sommets i, j sont adjacents dans le graphe si les examens i et j ont des étudiants en commun. Nous assignons une couleur à chaque sommet de tel sorte que deux sommets adjaçants n'aient pas la même couleur. L'objectif est de minimiser le nombre de jours d'organisation des examens qui correspond à minimiser le nombre de couleur du graphe.

4.2 Résolution du problème d'organisation des examens

4.2.1 Organisation des examens d'un semestre à la FST de Fès

La Faculté des sciences et Techniques-Fès doit organiser les horaires des examens d'un semestre qui contient (S1, S2, S3, S4), il s'agit cinq spécialité : CAM-S4, CSA-S4, ETI-S4, GIND-S4, GINF-S4, chaque spécialité et semestre (S1, S2, S3) contient quatre module

Trente-deux épreuves donc à planifier, correspondant aux cours numérotés de 1 à 32

Comment organiser ces examens dont le minimum de chrènos possible et sous la contrainte qu'aucun étudiant n'à passer deux épreuves en même temps ?

Le tableau suivant représente les modules de chaque semestre et spécialité :

semestre et Spécialité	Module
S1	M1, M2, M3, M4
S2	M5, M6, M7, M8
S3	M9, M10, M11, M12
CAM-S4	M13(CMA),M14(CMA), M15(CMA), M16(CMA)
CSA-S4	M13(CSA), M14(CSA), M15(CSA), M16(CSA)
ETI-S4	M13(ETI), M14(ETI), M15(ETI), M16(ETI)
GIND-S4	M13(GIND),M14(GIND),M15(GIND), M16(GIND)
GINF-S4	M13(GINF), M14(GINF), M15(GINF), M16(GINF)

Le tableau suivant représente les différents groupes des étudiants ayant un semestre de réserve est ils sont inscris dans un modèle précis :

Ensemble	Module	Ensemble	Module
E1	M1	E17	M13(CSA)
E2	M2	E18	M14(CSA)
E3	M3	E19	M15(CSA)
E4	M4	E20	M16(CSA)
E5	M5	E21	M13(ETI)
E6	M6	E22	M14(ETI)
E7	M7	E23	M15(ETI)
E8	M8	E24	M16(ETI)
E9	M9	E25	M13(GIND)

E10	M10	E26	M14(GIND)
E11	M11	E27	M15(GIND)
E12	M12	E28	M16(GIND)
E13	M13(CAM)	E29	M13(GINF)
E14	M14(CAM)	E30	M14(GINF)
E15	M15(CAM)	E31	M15(GINF)
E16	M16(CAM)	E32	M16(GINF)

Le tableau suivant représente les noms des étudiants qui sont dans le groupe Ei,

i= {1,……,32}

E1	ALLOUCH REDA, BOUNNOUH AMINE, DZIRI HAMZA, ER-RAGRAGUI SALAH EDDINE, MOUHIB OTHMAN, SEBTI MOHAMMED, ABBOU RACHID, AFOURAOU AMAL, AKEL SOUMAYA, BELKHOU JIHANE, BENSLIMANE ASMAE, BERRAHO FADOUA, BOUADLIATTAR AYMEN, BOUCHRI YOUSRA, BOUKHRISS HAMZA, BOUYARMANE MOHAMMED, BOUZAFFOUR MOHAMED, BOUZAID MOHAMED, CHABLI BILAL, CHOUIREF ZINEB, CHOUKRY YOUSSEF, CHRIFI ALAOUI A BBES, EL BARAKA AMAL, EL GANDOULI YOUSSEF, EL KHOMSI SABRINE, ELASSRI AMINE, EL-MRABET AMINA, ERROUDI WAFAE, HOUSNI NAIM, JABER ASMAE, KADDOURI YASSINE, KAINI SAAD, LAAYOUN NAWAL, LEFAF ADIL, MADINI OUALID, MEJAHED KAUTAR , MOUFID RABAB, MOUNAIME RIHAB, MOUNOUAR OTHMANE, MOUSTAHSINE SMAIL, NADAH IBRAHIM, NAH FOUAD, NALI MOHSSINE, OUKICHA NAJWA, RABHI SORAYA, RHALEB BASMA, SEDIKI YASSINE, TAIBI ZINEB, TAMIMI SOUFIANE, TLEMCANI YOUSSEF, NACHIT ZOUHIR)
E2	BOUZIDI MOHAMMED ABDERRAHMANE, MOUHIB OTHMAN, CHRIFI ALAOUI A BBES, EL BARAKA AMAL, ELASSRI AMINE, EL-MRABET AMINA, JABER ASMAE, LEFAF ADIL, MADINI OUALID, MOUSTAHSINE SMAIL, TAIBI ZINEB, ZOUINE MOHAMMED AMINE
E3	LOUDYI KHADIJA, AKEL SOUMAYA, CHOUKRY YOUSSEF, EL BARAKA AMAL, IDIR CHAFIQ, MOUSTAHSINE SMAIL, TAIBI ZINEB, ZOUINE MOHAMMED AMINE
E4	CHOUKRY YOUSSEF, NAH FOUAD
E5	LOUDYI KHADIJA, BOUADLI ATTAR AYMEN, CHABLI BILAL, CHOUIREF ZINEB, EL HABCHI BADR, ELASSRI AMINE, ELKACHCHABI YASSIR, IDIR CHAFIQ, RABAH NABIL, SADIK SOUKAINA

E6	BENMOUSSA MAROUA, DKAKI OUMAYMA, ELHILALI NOURA, LOUDYI KHADIJA, MOUHIB OTHMAN, TIJAHI FATIMA ZAHRAE, BELFAKIH ILHAM, BELMAJDOUB ISMAIL, BENSLIMANE ASMAE, BERRAHO FADOUA, BOUADLI ATTAR AYMEN, BOUCHRI YOUSRA, BOUKHRISS HAMZA, BOURI YOUSRA, CHAIBI FATIMA ZAHRAE, CHOUIREF ZINEB, EL HABCHI BADR, ELASSRI AMINE, ELKACHCHABI YASSIR, IDIR CHAFIQ, JABER ASMAE, MADINI OUALID, MEJAHED KAOUTAR, MOUFID RABAB, MOUNAIME RIHAB, NAH FOUAD, NALI MOHSSINE, OULD AHMED OULD HAMIDOUN MOKHTAR, RABAH NABIL, RABHI SORAYA, RHALEB BASMA, SAADAOUI ISSAM, SADIK SOUKAINA, SEKKAT ABDERRAHIM, STITOU BOUZID, TAIBI ZINEB, TAOUIL MERYEM, TLEMCANI YOUSSEF, TOULOUT SALMA, ZOUINE MOHAMMED AMINE, LAKRAMTI FIRDAOUS, MATTOUHI MERIEM, TELMANI NASSIRA
E7	BELFAKIH ILHAM, BELKHOU JIHANE, BENSLIMANE ASMAE, BOUADLI ATTAR AYMEN, BOUKHRISS HAMZA, BOURI YOUSRA, CHABLI BILAL, CHOUIREF ZINEB, EL GANDOULI YOUSSEF, EL HABCHI BADR, LAAYOUN NAWAL, LEFAF ADIL, MEJAHED KAOUTAR, SAADAOUI ISSAM, SADIK SOUKAINA, SEDIKI YASSINE, TAOUIL MERYEM, TLEMCANI YOUSSEF, ZOUINE MOHAMMED AMINE, TELMANI NASSIRA
E8	SAADAOUI ISSAM
E9	BALLOUK KARIMA, BENISS MOHAMED AMINE, BOUNNOUH AMINE, DJAZI MEHDI, DKAKI OUMAYMA, ECHCHERKI OUSSAMA, EDDARIF FATIHA, ER-RAGRAGUI SALAH EDDINE, ESSAHAL WALID, EZZENATI SARA, HEDDA SARA, LADRAA AICHA, LOULIDI YASSINE, SEBTI MOHAMMED, TIJAHI FATIMA ZAHRAE, ABBOU RACHID, AFOURAOU AMAL, AKEL SOUMAYA, BELKHOU JIHANE, BELMAJDOUB ISMAIL, BERRAHO FADOUA, BOURI YOUSRA, BOUYARMANE MOHAMMED, BOUZAFFOUR MOHAMED, BOUZAID MOHAMED, CHABLI BILAL, CHAIBI FATIMA ZAHRAE, CHOUKRY YOUSSEF, CHRIFI ALAOUI A BBES, DOBLI BENNANI ABDESSALAM, EL BARAKA AMAL, EL GANDOULI YOUSSEF, EL KHOMSI SABRINE, ELKACHCHABI YASSIR, EL-MRABET AMINA, ERROUDI WAFAE, HOUSNI NAIM, IDIR CHAFIQ, KADDOURI YASSINE, KAINI SAAD, LAAYOUN NAWAL, MOUNOUAR OTHMANE, NADAH IBRAHIM, NAH FOUAD, NALI MOHSSINE, OUARD ZINEB, OUKICHA NAJWA, OULD AHMED OULD HAMIDOUN MOKHTAR, RHALEB BASMA, SEDIKI YASSINE, SEKKAT ABDERRAHIM, TAMIMI SOUFIANE, TAOUIL MERYEM, TOULOUT SALMA, FILALI MOUHIM ISSAM, M'HAMDI HAJAR, MOUFID FATIMA ZAHRA, BENBOUBKER HAMZA, BENJELLOUN MOHAMMED SALIM, EL YAMANI MERYEM, MOUSLIH GHIZLANE, IDRISSI MELIANI MOHAMMED, SENHAJI MEHDI, TAHRI JOUTEI ZINEB, ZEHOUANI FATMA, ALIOU DIALLO AOUDI MOHAMED HABIB, CHAOUQUI AKRAM
	ALLOUCH REDA, BENISS MOHAMED AMINE, BOUNNOUH AMINE, BOUZIDI MOHAMMED ABDERRAHMANE, DJAZI MEHDI, DZIRI HAMZA, ECHCHERKI OUSSAMA, EDDARIF FATIHA, ESSAHAL

E10	WALID, EZZENATI SARA, HEDDA SARA, LADRAA AICHA, ABBOU RACHID, AFOURAOU AMAL, AKEL SOUMAYA, BELMAJDOUB ISMAIL, BOUCHRI YOUSRA, BOUZAFFOUR MOHAMED, BOUZAID MOHAMED, CHAIBI FATIMA ZAHRAE, DOBLI BENNANI ABDESSALAM, EL KHOMSI SABRINE, ELKACHCHABI YASSIR, ERROUDI WAFAE, HOUSNI NAIM, KADDOURI YASSINE, KAINI SAAD, MOUNAIME RIHAB, NALI MOHSSINE, OUARD ZINEB, RAFIE SOUMIA, SEKKAT ABDERRAHIM, TOULOUT SALMA, M'HAMDI HAJAR, MOUFID FATIMA ZAHRA, EL YAMANI MERYEM, TCHICH GHITA
E11	ALLOUCH REDA, BALLOUK KARIMA, BENMOUSSA MAROUA, BOUZIDI MOHAMMED ABDERRAHMANE, DJAZI MEHDI, DKAKI OUMAYMA, ECHCHERKI OUSSAMA, EDDARIF FATIHA, ELHILALI NOURA, ESSAHAL WALID, EZZENATI SARA, HEDDA SARA, LADRAA AICHA, LOULIDI YASSINE, TIJAHI FATIMA ZAHRAE, ABBOU RACHID, BELMAJDOUB ISMAIL, BERRAHO FADOUA, BOUCHRI YOUSRA, CHAIBI FATIMA ZAHRAE, CHRIFI ALAOUI A BBES, EL GANDOULI YOUSSEF, HOUSNI NAIM, KADA OUMAIMA, KADDOURI YASSINE, KAINI SAAD, LAAYOUN NAWAL, MOUNAIME RIHAB, NADAH IBRAHIM, OUARD ZINEB, OUKICHA NAJWA, OULD AHMED OULD HAMIDOUN MOKHTAR, RAFIE SOUMIA, STITOU BOUZID, TAMIMI SOUFIANE, TOULOUT SALMA, DAROUICHE BAHIJA, MOUFID FATIMA ZAHRA, TCHICH GHITA, ALIOU DIALLO AOUDI MOHAMED HABIB, CHAOUQUI AKRAM
E12	ALLOUCH REDA, BALLOUK KARIMA, BENISS MOHAMED AMINE, BENMOUSSA MAROUA, BOUNNOUH AMINE, BOUZIDI MOHAMMED ABDERRAHMANE, DJAZI MEHDI, DKAKI OUMAYMA, DZIRI HAMZA, ECHCHERKI OUSSAMA, EDDARIF FATIHA, ELHILALI NOURA, ER-RAGRAGUI SALAH EDDINE, ESSAHAL WALID, EZZENATI SARA, HEDDA SARA, LADRAA AICHA, LOUDYI KHADIJA, LOULIDI YASSINE, MOUHIB OTHMAN, SEBTI MOHAMMED, TIJAHI FATIMA ZAHRAE, SBAI AMIN, ELMEKAOU OUSSAMA, BOUYARMANE HOUDA, MOUFID FATIMA ZAHRA
E13	HADDAOUI SABAH, LAHLOU DRISS
E14	HADDAOUI SABAH, LAHLOU DRISS
E15	
E16	HADDAOUI SABAH, LAHLOU DRISS
E17	ASSEMLAL ISSAM, BAKHRI ASSINE, BOUMAIS MOHAMMED, SKIREDJ MOHAMED, ZGHIMRI ACHRAF, GHAZOUANI FATIMA ZAHRA
E18	SBAI AMIN, ASSEMLAL ISSAM, BAKHRI YASSINE, BERTAL HANANE, BOUMAIS MOHAMMED, IDRISSI HANANE, SKIREDJ MOHAMED, ZGHIMRI ACHRAF, ELANSARI MOHAMMED, MEJBAR MOHAMMED REDA
E19	ASSEMLAL ISSAM, BAKHRI YASSINE, BOUMAIS MOHAMMED, IDRISSI HANANE, SKIREDJ MOHAMED, MEJBAR MOHAMMED REDA
E20	BOUMAIS MOHAMMED, SKIREDJ MOHAMED, EL HAMMOUMI MOHAMMED
E21	ELMEKAOUI OUSSAMA

E22	ELMEKAOUI OUSSAMA, CHRAIBI MERYEME, REZKI KHADIJA
E23	REZKI KHADIJA, SALIM MOHAMMED
E24	ELMEKAOUI OUSSAMA
E25	AHOUFI ANAS, LEGRA SOW MAHMOUD
E26	
E27	OUENJLI SOUMAYA, AHOUFI ANAS, LEGRA SOW MAHMOUD
E28	OUENJLI SOUMAYA
E29	BOURAKKADI MOHAMMED, DAROUICHE BAHIJA, EL ARAB YAHIA, HAMEDOUN MOHAMMED JABER
E30	EL ARAB YAHIA, EL ISSAOUI NAOUFAL, FILALI MOUHIM ISSAM, LAHRICHI SAFAE
E31	BOURAKKADI MOHAMMED, DAROUICHE BAHIJA, EL ARAB YAHIA, HAMEDOUN MOHAMMED JABER
E32	BOURAKKADI MOHAMMED, EL ARAB YAHIA, FILALI MOUHIM ISSAM, HAMEDOUN MOHAMMED JABER, LAHRICHI SAFAE

4.2.2 Représentation matricielle

Pour résoudre le problème d'organisation des examens en utilisant la coloration des graphes, nous proposons une modélisation sous forme d'une matrice carré symétrique contient 0 et 1, 1 si i et j sont deux modules de même spécialité où s'ils ont au moins un étudiant réserve en commun et 0 sinon avec i,j = {1,……………,32}

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

4.2.3 Étapes de résolution

Pour colorier un graphe nous proposons une présentation sous forme d'une matrice symétrique qui contient 0 et 1, à partir de cette matrice nous proposons un programme qui permet de colorier tous les sommets du graphe en déterminant à chaque fois un ensemble indépendant qui nous donne une couleur.

Les étapes de l’exécution :

étape1 : Initialisation de l’ensemble indépendant par le sommet M1

étape2 : Chercher le premier sommet j non adjacent avec le sommet de l’ensemble puis l’ajouter a notre ensemble

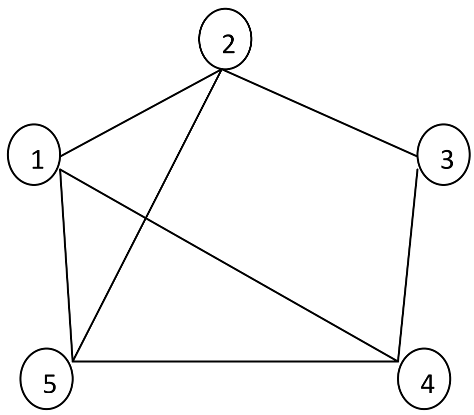
étape3 : Chercher un sommet j qui n’est adjacent avec aucun sommet de l’ensemble indépendant, si tous les sommets restant sont adjacents avec au mois un sommet de l’ensemble indépendant, on passe à l’étape d, sinon on répète cette dernière

étape4 : donner une couleur à cet ensemble puis barrer en ligne et en colonne tous les sommets sélectionnés dans l’ensemble

étape5 : Revenir à l’étape1 jusqu’à ce que tous les sommets des graphes soient coloriés

Exemple

Considérons le graphe suivant :



Sur ce graphe, nous allons appliquer l’algorithme précédent pour le colorier, nous considérons la matrice de graphe A= (a_{ij}) suivante :

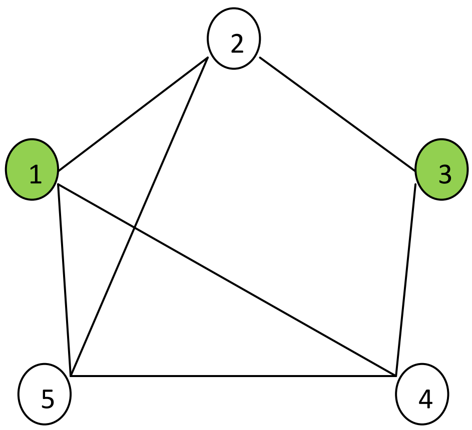
$$A=\begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

L’ensemble indépendant à été initialise par le sommet 1

On note par L notre ensemble indépendant, qui est jusqu’à maintenant L={1}

Puis on cherche le premier sommet non adjacent avec le sommet « 1 », c’est-à-dire on cherche le premier 0 dans la ligne qui correspond au sommet « 1 », dans ce cas c’est le sommet « 3 », donc L devient L= {1,3}.

Puis on cherche un autre sommet, qui ne doit être adjacent avec aucun sommet de L, dans ce cas il n’existe pas, donc on va donner à cet ensemble la couleur 1.



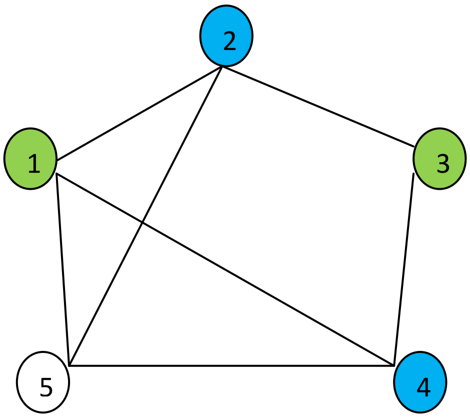
Puis on supprime en ligne et en colonne les éléments qu’on a coloriés.

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

La matrice obtenue est

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

On refait le même travail, On a « 2 » et « 4 » non adjacent donc L= {2, 4}, de même cet ensemble est devenu comme un ensemble indépendant maximal, car on ne peut ajouter aucun sommet a cet ensemble, et on lui donne la couleur 2.

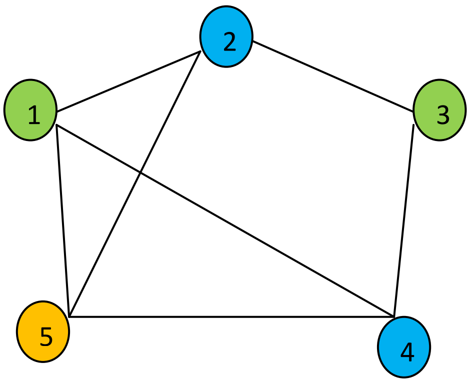


On supprime ces deux sommets de la matrice

$$A=\begin{pmatrix} \cancel{0} & \cancel{0} & \cancel{1} \\ \cancel{0} & \cancel{0} & \cancel{1} \\ \cancel{1} & \cancel{1} & 0 \end{pmatrix}$$

On trouve une matrice qui contient un seul élément, alors évidement c'est le dernier ensemble indépendant, et on lui donne la couleur 3.

Donc le graphe est colorié de la façon suivant :

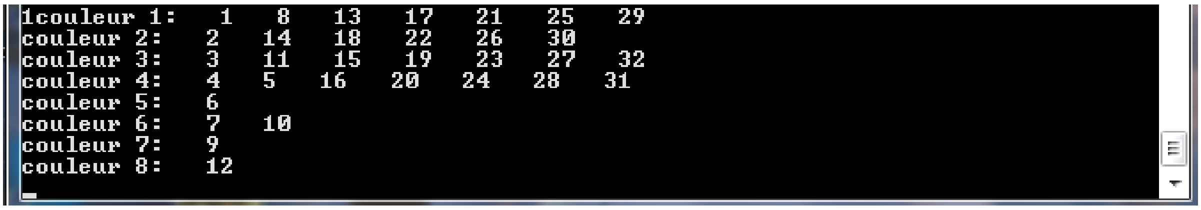


Voila le résultat donnez par le programme :

```
1couleur 1:   1   3
couleur 2:   2   4
couleur 3:   5
```

4.2.4 Résultats expérimentaux

Dans cette section, nous présentons les résultats expérimentaux après l'execution de notre programme, on obtient la meilleure solution affiché par la figure suivante :



```
1couleur 1:   1   8   13   17   21   25   29
couleur 2:   2   14   18   22   26   30
couleur 3:   3   11   15   19   23   27   32
couleur 4:   4   5   16   20   24   28   31
couleur 5:   6
couleur 6:   7   10
couleur 7:   9
couleur 8:   12
```

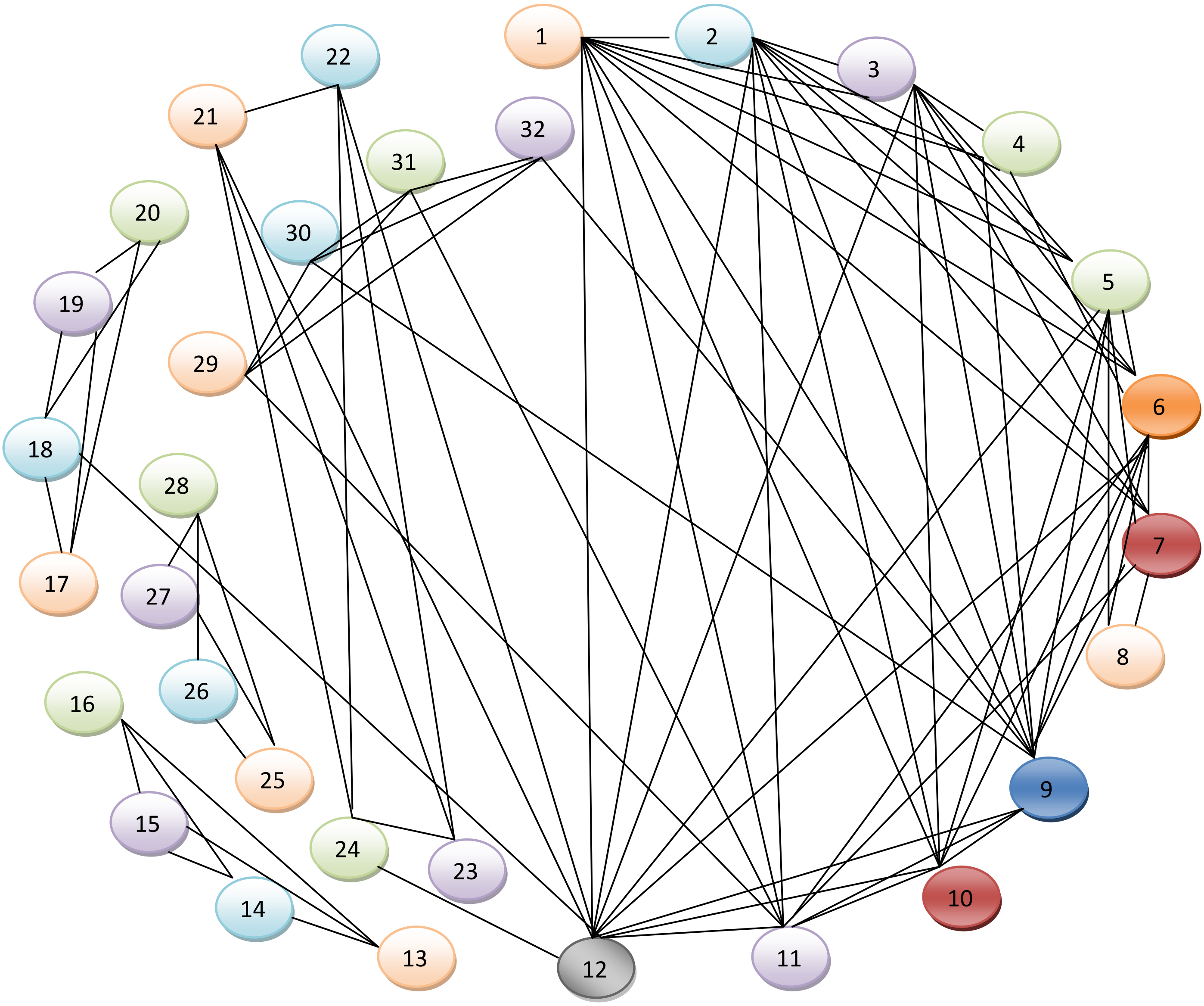
Solution obtenue pour l'organisation des examens : un semestre à la FST de Fés

Les examens peuvent être répartis en 8 périodes, de la manière suivante :

- 1ère période, épreuve des cours : M1, M8, M13 (CAM), M13 (CSA), M13 (ETI), M13(GIND), M15 (GINF).
- 2ème période, épreuve des cours : M2, M14(CAM), M14(CSA), M14(ETI), M14(GIND), M14(GINF).
- 3ème période, épreuve des cours : M3, M11, M15(CAM), M15(CSA), M5(ETI), M15(GIND), M16(GINF).
- 4ème période, épreuve des cours : M4, M5, M16(CAM), M16(CSA), M16(ETI), M16(GIND), M15(GINF).
- 5ème période, épreuve de cours : M6.
- 6ème période, épreuve des cours : M7, M10.
- 7ème période, épreuve des cours : M9.
- 8ème période, épreuve des cours : M12.

4.2.5 Représentation sous forme de graphe

La représentation de la solution peut s'exprimer sous forme d'un graphe non orienté dans les sommets sont les examens, deux sommets i, j sont adjacents dans le graphe si les examens i et j ont des étudiants en commun. Nous assignons une couleur à chaque sommet de tel sorte que deux sommets adjaçants n'aient pas la même couleur.


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32

Conclusion générale

Dans ce mémoire, nous avons étudié le problème de la coloration de graphe, qui est un problème central de l'optimisation combinatoire, ce problème à de nombreuses applications pratiques, telles que les problèmes d'emploi du temps, les problèmes des gestions d'entrepôts, ou encore les problèmes d'allocation de fréquences.

Nous avons présenté au deuxième chapitre le problème de gestion des examens, et nous avons donné par la suite une représentation matricielle est graphique de ce problème.

Dans le chapitre 3 nous avons introduit le problème de la coloration de graphe, et puis le problème d'ensemble indépendant maximal, nous avons proposé une heuristique pour la résolution de ce problème basée sur la relaxation surrogate de la programmation entière.

Finalement dans le dernier chapitre nous avons appliqué la coloration de graphe à un problème réel de l'emploi du temps des examens de la Faculté des Sciences et Technique de Fès. Après l'exécution de notre programme faite sur C++ nous avons obtenue une solution qui respect les contraintes en particulier la contrainte qu'aucun étudiant n'à passer deux épreuves en même temps.

Bibliographie

1. Ahmed EL HILALI ALAOUI, Ghizlane BENCHEIKH, Fatima ELKHOUKHI. Livre-Initiation à la recherche opérationnelle. 2009.
2. Ahmed EL HILALI ALAOUI. Youssef BENNADADA. livre-programmation mathématique de la modélisation a la resolution. 2012.
3. Btissam DKHISSI.thèse-Problèmes de planification horaire : problèmes d'emploi du temps et problèmes de transport, modélisation et résolution par les métaheuristiques. 2011.
4. Hassania MESSAOUD.mémoire de master- Les problèmes de tournées de véhicules multi-dépôts modélisation par les graphes splits reslution par les metaheuristique. 2011.
5. Sidi Mohamed DOUIRI.Thèse-Contribution à la résolution des problèmes de la coloration et la somme coloration de graphes. 2012.
6. Mohamed HOSSEINI DOLAMA.thèse- Contribution à l'étude de quelques problèmes de coloration de graphes. 2005.
7. Pascal Côté.Thèse-Optimisation Multi-objective des problèmes combinatoires Application à la génération des horaires d'examens finaux. 2004.
8. Rachida ABOUNACER.thèse- Probèmes de planification des tournées de véhicules et planification des horaires des examens : Modélisation et résolution par les métheuristiques.2010.

[BURKE 96] Burke E.K., Newall J.P., Weare R.F., ‘A memetic algorithm for university Exam timetabling’. In : Burke E.K., Ross P. (eds). (1996). Practice and Theory of Automated Timetabling : Selected Papers from the 1 st International Conference. Springer Lecture Notes in Computer Science, vol. 1153. 241-250, 1996.

[BURKE 98a] Burke E.K., Newall J., ‘Initialization strategies and diversity in evlutionar timetabling ’. Evolutionary Computation, 6(1), 81- 103, 1998 .

[CARTER 96] Carter M.W., Laporte G., Lee S.Y., ’Examination timetabling : Algorithmic strategies and applications’. Journal of Operational Research Society, 47(3) : 373-383, 1996.

[GLOVER 1965] muliphase-dual algorithm for the zone-one integer programming problems. Operations Research, No. 6, 13 :879-893,1965.

[MERLOT 03] Merlot L.T.G., Boland N.,Hughes B.D., Stuckey P.J. , ‘A hybrid algorithm for the examination timetabling problem’. In : E .K. Burke and P.De Causmaecker (eds). (2003). Practice and Theory of Automated Timetabling : Selected Papers from the 4th International Conference. Springer Lecture Notes in Computer Science, vol. 2740. 207-231,2003.

[TERASHIMA-MARIN 99] Terashima-Marin H., Ross P., Valenzuela-Rendon M., ‘Evolution of constraint satisfaction strategies in examination timetabing’. In : Proceedings of the Genetic and Evolutionary Conference, Orlando, Florida, 635-642, 1999.