

# **TP de méthodes de résolution de problèmes NP-complets**

# Table des matières

<b>1</b>	<b>Partie théorique</b>	<b>2</b>
1.1	Exercice 1 - Sur le problème de la couverture sommet minimale : trois approches différentes : . . . . .	2
1.1.1	Première approche : la programmation linéaire en nombres entiers .	2
1.1.2	Seconde approche : la recherche d'un couplage maximal . . . . .	4
1.2	Exercice 2 - Sur le problème de la couverture d'ensembles . . . . .	5
1.2.1	Modélisation du problème à l'aide de la PLNE . . . . .	5
1.2.2	Procédure d'arrondis . . . . .	5
1.2.3	Existence d'un algorithme $f$ -approché . . . . .	6
1.3	Exercice 3 - Sur le problème du couplage maximum de poids minimum . .	6
1.3.1	Modélisation du problème . . . . .	6
1.3.2	Modélisation du problème appliquée au graphe de la figure 2 . . . .	6
1.3.3	Solution optimale entière $z(ILP)$ . . . . .	7
1.3.4	Solution optimale $z(LP)$ pour le programme relaxé . . . . .	7
1.3.5	Conclusion sur la pertinence de la formulation . . . . .	7
<b>2</b>	<b>Partie pratique</b>	<b>8</b>

# Chapitre 1

## Partie théorique

### 1.1 Exercice 1 - Sur le problème de la couverture sommet minimale : trois approches différentes :

#### 1.1.1 Première approche : la programmation linéaire en nombres entiers

##### Justification de l'utilisation de la Programmation Linéaire en Nombres Entiers

On considère le problème de la couverture minimale sous la forme suivante :

$$\begin{cases} \min z = \sum_{j=1}^n x_j \\ x_r + x_s \geq 1, \quad \forall \{v_r, v_s\} \in E \\ x_j \in \{0, 1\} \quad j = 1, \dots, n \end{cases}$$

La fonction objectif représente le nombre de sommets utilisés par la solution du problème. Le fait de minimiser la fonction objectif permet d'assurer la couverture minimale. Chacune des clauses est relative à une arête du graphe, et impose qu'au moins un des sommets adjacents à cette arête soit dans la couverture.

On a donc bien un problème de Programmation Linéaire en Nombres Entiers permettant de résoudre le problème de la couverture minimale.

##### Justification des clauses

Considérons le graphe donné par la figure 1.1.1.

Sur ce graphe, le programme linéaire en nombres entiers est le suivant :

$$\begin{cases} \min z = x_A + x_B + x_C \\ x_A + x_B \geq 1 \\ x_A + x_C \geq 1 \\ x_B + x_C \geq 1 \\ x_A, x_B, x_C \in \{0, 1\} \end{cases}$$

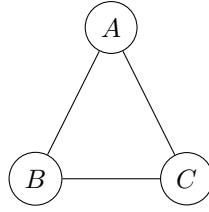


FIGURE 1.1 – Exemple

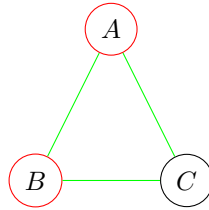


FIGURE 1.2 – Solution

Il est très simple ici de comprendre pourquoi il est impossible de considérer le programme linéaire suivant :

$$\begin{cases} \min z = x_A + x_B + x_C \\ x_A + x_B = 1 \\ x_A + x_C = 1 \\ x_B + x_C = 1 \\ x_A, x_B, x_C \in \{0, 1\} \end{cases}$$

Ce programme ne permet pas de résoudre la couverture minimale sur le graphe donné par la figure 1.1.1. Quelque soit le sommet choisi dans un premier lieu pour appartenir à la couverture minimale, il est impossible d'en choisir un second pour compléter cette dernière. Prenons un exemple, nous forçons le sommet  $A$  à appartenir à la couverture minimale (respectivement  $B$  et  $C$ ). Ce choix force :  $x_B = 0$  et  $x_C = 0$  (respectivement,  $x_A = 0$  et  $x_C = 0$ , et  $x_A = 0$  et  $x_B = 0$ ). Il est donc impossible de respecter la clause  $x_B + x_C = 1$ , le problème (au vu de sa modélisation) n'aurait donc pas de solution, or le graphe de la figure 1.1.1 montre le contraire.

### Une borne inférieure des solutions optimales

On cherche à montrer qu'une solution optimale du programme linéaire en nombres entiers est une borne inférieure de toute solution optimale du programme relaxé. Raisonnons par l'absurde et considérons une solution optimale du programme linéaire, notée  $n^*$  telle qu'il existe  $x^*$  solution optimale du problème relaxé vérifiant  $x^* < n^*$ . Toute solution du programme linéaire est solution du programme relaxé<sup>1</sup>. Ceci implique :  $n^*$  solution du programme relaxé, et donc  $x^* < n^*$  impossible. On a donc :  $n^* \leq x^*$  ce qui est la définition d'une borne inférieure.

1. Une solution appartenant à  $\mathbb{N}$  appartient aussi à  $\mathbb{R}$

## A propos de la relaxation de contrainte

Pour démontrer que la relaxation des contraintes d'intégrité implique  $x_r \geq \frac{1}{2}$  ou  $x_s \geq \frac{1}{2}$ , le raisonnement par l'absurde sera utilisé. Soient  $x_s$  et  $x_r$  les variables relatives aux sommets  $r$  et  $s$  adjacents à l'arête  $(rc)$  et telles que, après relaxation des contraintes, on a :  $x_r < \frac{1}{2}$  et  $x_s < \frac{1}{2}$ . On en déduit donc que  $x_r + x_s < 1$  et donc la contrainte liée à l'arête  $(rs)$  est violée, l'hypothèse de départ est donc fausse. On a donc,  $\forall (rs) \in V : x_r \geq \frac{1}{2}$  et  $x_s \geq \frac{1}{2}$ .

## Une 2-approximation

Mettons en évidence le pire des cas pouvant se présenter : pour une arête  $(rs) \in V$ , un seul sommet est nécessaire pour la couverture de cette dernière dans le cas de la couverture minimale, mais l'algorithme approché retourne :  $x_r = x_s = \frac{1}{2}$ . Après la phase d'arrondis, on a  $x_r = x_s = 1$  et donc les deux sommets appartiennent à la solution approchée, cette phase multiplie donc au pire le nombre de sommets (pour chaque clause par 2), ce qui implique que le cardinal de la solution approximée est au plus 2 fois la solution optimale.

Cet algorithme est donc une 2-approximation.

## Dans le cas d'un graphe valué

### 1.1.2 Seconde approche : la recherche d'un couplage maximal

## Une 2-approximation

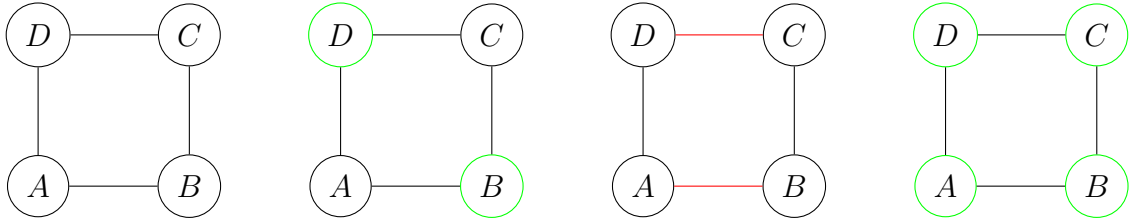
Commençons par prouver que l'algorithme retourne une couverture des arêtes par les sommets. Considérons donc une arête  $(rs)$  non couverte par l'ensemble de sommets retourné par l'algorithme, par définition du couplage, il serait donc possible d'ajouter  $(rs)$  au couplage. Or le couplage calculé par l'algorithme est maximal, on en déduit que l'arête  $(rs)$  telle qu'elle est définie ne peut exister et donc que l'ensemble de sommets obtenu couvre l'ensemble des arêtes du graphe.

Appelons  $c$  le couplage calculé par l'algorithme et  $x^*$  la solution optimale du problème de la couverture par les sommets, on sait que  $\text{Card}c \leq \text{Card}x^*$ , or pour construire la solution approchée, on ajoute à  $C$  les deux extrémités des arêtes utilisées pour le couplage. On a donc :

$$\begin{aligned} \text{Card}C &= 2 \times \text{Card}c \leq 2 \times \text{Card}x^* \\ \implies \frac{\text{Card}C}{\text{Card}x^*} &= 2 \end{aligned}$$

**Exemple de graphe foireux**

Le graphe suivant met en évidence la borne 2 de l'algorithme.



Le graphe      Une couverture minimale      Un couplage maximale      La solution renvoyée

**Application de l'algorithme****1.2 Exercice 2 - Sur le problème de la couverture d'ensembles****1.2.1 Modélisation du problème à l'aide de la PLNE**

On considère le problème de la couverture d'ensemble, défini par : Soit  $E = \{e_1, \dots, e_n\}$ , soient  $S_1, \dots, S_m$  des sous-ensembles non vides de  $E$  tels que  $\forall i \in \{1, \dots, m\}$ , on a :  $S_i \subset E$ . On associe à chaque ensemble  $S_j$  un poids  $w_j \geq 0$ . Le problème consiste à trouver une collection de sous-ensemble de poids minimum et telle que  $\bigcup_{i=1}^m S_i = E$ .

Ce problème peut s'exprimer à l'aide de la Programmation Linéaire en Nombres entiers de la manière suivante :

$$\begin{cases} \min z = \sum_{i=1}^m w_i x_i \\ \sum_{x_j/e_i \in x_j} x_j \geq 1 \quad \forall e_i \in E \\ x_i \in \{0, 1\} \end{cases}$$

**1.2.2 Procédure d'arrondis**

Soit  $f = \max_{i=1, \dots, n} f_i$ , avec  $f_i$  le cardinal de l'ensemble des sous-ensembles de  $E$  contenant  $e_i$  :  $f_i = |\{j : e_i \in S_j\}|$ .

Définissons la procédure d'arrondis suivante :

$$x_i = \begin{cases} 1 & \text{si } x_i \geq \frac{1}{f} \\ 0 & \text{sinon} \end{cases}$$

Cherchons à démontrer que cette procédure d'arrondis garantit une solution réalisable. Pour ce faire, nous allons procéder par l'absurde. Supposons qu'il existe un sommet  $k$  qui ne respecte pas sa contrainte d'intégrité associée, à savoir :

$$\sum_{x_j/e_k \in x_j} < 1$$

Les variables  $x_j$  de cette contrainte étant définies positives et entières, le cas pris en considération si dessus implique que toutes les variables de l'inéquation sont nulles pour le programme en nombres entiers et donc :

$$\forall x_j/e_k \in x_j < \frac{1}{f}$$

dans le cas de la version relaxée du problème. Or ceci n'est possible que si le nombre de sous-ensemble contenant  $x_k$  est supérieur à  $f$ , ce qui est impossible par définition. Tous les sommets respectent donc leur contrainte d'intégrité et on en déduit que la procédure d'arrondis garantit une solution réalisable.

### 1.2.3 Existence d'un algorithme $f$ -approché

## 1.3 Exercice 3 - Sur le problème du couplage maximum de poids minimum

### 1.3.1 Modélisation du problème

Soit un graphe  $G = (V, E)$  avec  $V$  l'ensemble de ses sommets et  $E$  l'ensemble de ses arêtes. Soit  $\{\forall(i, j) \in E, X_{(i,j)}\}$  un ensemble de variables booléennes qui indiquent le choix de l'arête  $(i, j)$  correspondante dans le couplage. Soit  $P_{(i,j)}$  le poids de l'arête  $(i, j)$ .

Le problème du couplage maximum de poids minimum peut être modélisé de la façon suivante :

*Minimiser*

$$\sum_{(i,j) \in E} (X_{(i,j)} \times P_{(i,j)})$$

*Sous contraintes*

$$\forall i \in V, \sum_{(i,j) \in E} X_{(i,j)} = 1$$

$$\forall(i, j) \in E, X_{(i,j)} \in \{0, 1\}$$

$$\forall(i, j) \in E, P_{(i,j)} \geq 0$$

En effet, les contraintes forcent le couplage à être maximum tandis que la fonction objectif le force à tendre vers le poids minimum.

### 1.3.2 Modélisation du problème appliquée au graphe de la figure 2

*Minimiser*

$$\epsilon \times X_{ab} + \epsilon \times X_{bc} + \epsilon \times X_{ac} + M \times X_{ae} + M \times X_{cd} + M \times X_{bf} + \epsilon \times X_{df} + \epsilon \times X_{de} + \epsilon \times X_{fe}$$

*Sous contraintes*

$$X_{ab} + X_{ac} + X_{ae} = 1$$

$$\begin{aligned}
X_{ab} + X_{bc} + X_{bf} &= 1 \\
X_{ac} + X_{bc} + X_{cd} &= 1 \\
X_{cd} + X_{de} + X_{df} &= 1 \\
X_{ae} + X_{de} + X_{df} &= 1 \\
X_{ef} + X_{df} + X_{bf} &= 1 \\
\forall (i, j) \in E, X_{(i,j)} &\in \{0, 1\} \\
\epsilon &\geq 0 \\
M &\geq 0
\end{aligned}$$

### 1.3.3 Solution optimale entière $z(ILP)$

Sur un exemple de cette taille, il est facile de trouver une solution à la main. Il y a plusieurs solutions optimales de poids total  $M + 2\epsilon$  sur cet exemple ; l'une d'entre elles est le couplage  $\{(a, b), (c, d), (e, f)\}$ .

La résolution de ce PLNE par *glpsol* (solveur de *GLPK*) donne bien la même solution.

### 1.3.4 Solution optimale $z(LP)$ pour le programme relaxé

Relaxer le programme revient à transformer la contrainte d'intégrité des  $X_{(i,j)}$  en la contrainte suivante :

$$\forall (i, j) \in E, 0 \leq X_{(i,j)} \leq 1$$

En rentrant le PL relaxé dans *glpsol*, nous obtenons le couplage de poids total  $3\epsilon$  :

$$\{X_{ab} = 0.5; X_{ac} = 0.5; X_{bc} = 0.5; X_{df} = 0.5; X_{ef} = 0.5; X_{de} = 0.5\}$$

### 1.3.5 Conclusion sur la pertinence de la formulation



## Chapitre 2

# Partie pratique