

Université de Montréal

Études du jeu de poursuite dans les graphes

par

Youssef Zine

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Mémoire présenté à la Faculté des études supérieures
en vue de l'obtention du grade de
Maître ès sciences (M.Sc.)
en informatique

août 2005



QA

76

U54

2006

v.007



Direction des bibliothèques

AVIS

L'auteur a autorisé l'Université de Montréal à reproduire et diffuser, en totalité ou en partie, par quelque moyen que ce soit et sur quelque support que ce soit, et exclusivement à des fins non lucratives d'enseignement et de recherche, des copies de ce mémoire ou de cette thèse.

L'auteur et les coauteurs le cas échéant conservent la propriété du droit d'auteur et des droits moraux qui protègent ce document. Ni la thèse ou le mémoire, ni des extraits substantiels de ce document, ne doivent être imprimés ou autrement reproduits sans l'autorisation de l'auteur.

Afin de se conformer à la Loi canadienne sur la protection des renseignements personnels, quelques formulaires secondaires, coordonnées ou signatures intégrées au texte ont pu être enlevés de ce document. Bien que cela ait pu affecter la pagination, il n'y a aucun contenu manquant.

NOTICE

The author of this thesis or dissertation has granted a nonexclusive license allowing Université de Montréal to reproduce and publish the document, in part or in whole, and in any format, solely for noncommercial educational and research purposes.

The author and co-authors if applicable retain copyright ownership and moral rights in this document. Neither the whole thesis or dissertation, nor substantial extracts from it, may be printed or otherwise reproduced without the author's permission.

In compliance with the Canadian Privacy Act some supporting forms, contact information or signatures may have been removed from the document. While this may affect the document page count, it does not represent any loss of content from the document.

Université de Montréal

Faculté des études supérieures

Ce mémoire intitulé

Études du jeu de poursuite dans les graphes

présenté par

Youssef Zine

a été évalué par un jury composé des personnes suivantes :

Petko Valtchev

(président-rapporteur)

Geňa Hahn

(directeur de recherche)

Alain Tapp

(membre du jury)

Mémoire accepté le:

9 décembre 2005

SOMMAIRE

Dans ce mémoire, on étudie les jeux de poursuite policier-voleur dans des graphes, sans aucune contraintes sur les déplacements des deux antagonistes.

Les deux joueurs ont une information complète sur la position actuelle de leurs adversaires, ainsi que sur leurs déplacements possibles après chaque position. L'environnement du jeu sera des graphes non-orientés réflexifs, partiellement réflexifs, ou totalement non réflexifs. Nous verrons des graphes où un policier (ou k policiers) peut gagner contre un voleur (ou l voleurs) en ayant une stratégie qui permet au(x) policier(s) d'occuper le même sommet que le(s) voleur(s); en général, ces graphes sont appelés (k, l) -policier-gagnants. De plus, nous verrons l'influence de la réflexivité de certains sommets sur le résultat d'un jeu optimal (un jeu, où chaque joueur choisit le meilleur déplacement à chaque étape).

Nous aborderons aussi les caractérisations des $(1, 1)$ -policier-gagnants réflexifs graphes qu'ils soient finis ou infinis, et par généralisation, celles des (k, l) -policier-gagnants graphes.

Après avoir examiné tous ces résultats pour des graphes non-orientés et après avoir ajouté quelques résultats simples qui nous sont propres, nous considérerons les graphes orientés. Aucune caractérisation simple n'étant connue dans ce cas-ci, nous verrons l'algorithme de Hahn et MacGillivray qui vérifie si un graphe donné (orienté ou non, avec boucles sur tous, certains, ou aucun sommet) est $(1, 1)$ -policier-gagnant ou non et, par simple extension, s'il est (k, l) -policier-gagnant ou non.

Enfin, pour clore ce mémoire, nous donnons une description de notre implémentation de l'algorithme de Hahn et MacGillivray, une discussion de ses performances,

et certains résultats obtenus pour des graphes dont on voulait connaître leur cop-number (le nombre minimum de policiers pour capturer un voleur dans un graphe donné).

Mots-clés : Jeu de poursuite, Graphe réflexif, policiers et voleurs, Stratégie gagnante, Système de Steiner.

SUMMARY

This thesis considers cop-and-robber pursuit games in graphs, without any constraints on the moves of the two antagonists.

The two players have complete information on each other's current position as well as on the possible moves from each position. The game is played on undirected graphs, reflexive, partially reflexive, or non-reflexives. We look at graphs on which the cop (or k cops) can win against a robber (or l robbers) by virtue of having a strategy which leads to the cop(s) occupying the same point(s) as the robber(s); in general, we call such graphs (k, l) -cop-win. We study the influence of loops on the result of an optimal game (that is, a game in which in player chooses a best move at each step).

We will see also the characterizations of $(1, 1)$ -cop-win reflexive graphs, finite or infinite which generalizes to (k, l) -cop-win graphs.

Having surveyed all these results for undirected graphs and having added a few simple results of our own, we consider directed graphs. As no simple characterization is known in this case, we take a look at an algorithm of Hahn and MacGillivray which decides whether a given graph (directed or not, with no, some or all loops) is cop-win or not and, by simple extension, whether it is (k, l) -cop-win or not.

We close the thesis with a description of our implementation of the algorithm, a discussion of its performance, and some results obtained through it for graphs which are of interest to other researchers who need to know their cop number (the minimum number of cops needed to catch one robber on the given graph).

Key words : Pursuit game, reflexive graph, cops and robbers, winning strategy,
Steiner triple system.

TABLE DES MATIÈRES

Sommaire	iii
Summary	v
Liste des figures	ix
Liste des tableaux	xi
Remerciements	1
Introduction	2
Chapitre 1. État de l'art	4
1.1. Notions de théorie des graphes.....	5
1.1.1. Relation d'ordre	5
1.1.2. Graphes et sous-graphes	6
1.1.3. Voisinage, adjacence	8
1.1.4. Connexité.....	10
1.1.5. Groupes, homomorphisme, contraction.....	11
1.2. Notions de la théorie des jeux.....	14
1.2.1. Description d'un jeu	14
1.2.2. Jeux de poursuite.....	18
1.2.2.1. Le jeu policier-voleur	19
Chapitre 2. Graphes policier-gagnants réflexifs	21
2.1. Caractérisation de graphes policier-gagnants finis.....	23
2.2. Caractérisation de graphes non finis policier-gagnants.....	26

2.3.	Stratégie pour un graphe policier-gagnant	28
2.3.1.	Stratégie policier-gagnante (Clarke et Nowakowski [11, 10])....	28
2.4.	Graphes à pont	30
Chapitre 3.	Graphes généraux	33
3.1.	Un policier, un voleur	34
3.1.1.	Algorithme de Hahn et MacGillivray	35
3.2.	Graphes généraux et le théorème de Nowakowski	40
3.3.	k-policiers, l-voleurs	46
3.4.	Quelques considérations et réflexions	48
Chapitre 4.	L'algorithme de Hahn et MacGillivray	51
4.1.	Les systèmes de triplets de Steiner	51
4.1.1.	Énumération	53
4.2.	Implémentation de l'algorithme	56
4.2.1.	Implémentation matricielle	56
4.2.2.	Implémentation avec listes	59
4.2.3.	Passage d'une représentation à l'autre	59
4.3.	Implémentation	60
4.4.	test de l'implémentation	61
4.5.	Considérations et observations	63
Conclusion	65	
Bibliographie	67	

LISTE DES FIGURES

1.1	3 types de graphes : G est un graphe non orienté, H est un graphe orienté et I est un graphe réflexif	7
1.2	Représentation d'un graphe connexe G , et un graphe H non connexe.	11
1.3	Illustration de points d'articulation dans un graphe	11
2.1	Graphe réflexif policier-gagnant	21
2.2	Graphe réflexif voleur-gagnant	22
2.3	G est policier-gagnant, alors que le sous-graphe induit H est voleur-gagnant	23
2.4	Le graphe du départ est voleur-gagnant, alors que sa rétraction est policier-gagnante	24
2.5	Les étapes de démontage d'un exemple de graphe policier-gagnant....	26
2.6	Exemples de graphes à pont et sous-graphes isométriques	31
3.1	Graphe policier-gagnant	38
3.2	Graphe voleur-gagnant	39
3.3	Ordre linéaire et dominance	41
3.4	Cas de dominance de v par u dans un graphe réflexif	42
3.5	Cas de dominance de v par u dans un graphe partiellement réflexif ...	42
3.6	Graphe avec un ordre linéaire policier-gagnant	43
3.7	Graphe G policier-gagnant, G' voleur-gagnant	44
3.8	Opération d'ouverture d'un sommet réflexif	45
3.9	Opération d'identification en un sommet réflexif	45

3.10	Graphe $(3, 1)$ -policier-gagnant, mais pas $(2, 1)$ -policier-gagnant	49
4.1	Plan de Fano	53
4.2	Construction du graphe de Steiner de 7 sommets	55
4.3	Matrice d'incidence sommets-arcs	56
4.4	Matrice d'incidence sommets-arêtes	57
4.5	Matrice d'adjacence	58
4.6	Liste d'adjacence	59
4.7	Graphe de Peterson	62

LISTE DES TABLEAUX

1.1	La représentation matricielle du jeu d'échecs	16
3.1	Les sommets du graphe de déplacement d'un graphe policier-gagnant.	39
3.2	Les sommets du graphe de déplacement d'un graphe voleur-gagnant ..	39
4.1	Nombre de $STS(n)$	53
4.2	Matrice d'incidence sommets-arcs	57
4.3	Matrice d'incidence sommets-arêtes	57
4.4	Fichier texte du graphe donné en entrée	61
4.5	Temps d'exécution de l'algorithme en secondes sur une machine Athlon 750 MHz avec 512 Mo de RAM	62
4.6	Temps d'exécution de l'algorithme en secondes sur une machine AMD 64 Athlon 3500+ MHz avec 2048 Mo de RAM	63

REMERCIEMENTS

J'aimerais remercier tout ceux qui m'ont soutenus durant mes études et la rédaction de ce mémoire. Plus particulièrement, j'aimerais remercier mon directeur de recherche Mr. Geña Hahn pour son soutien et sa compréhension sans faille, pour sa patience et son aide durant son encadrement. J'ai été grandement honoré de l'avoir comme directeur de recherche, et pour tout cela, je lui suis infiniment reconnaissant.

J'aimerais aussi remercier MITACS pour son soutien financier, de même que le DIRO et la DESI qui m'ont permis de travailler afin de financer mes études, mais aussi d'acquérir une expérience dans l'enseignement, et d'enrichir mes connaissances.

Enfin, j'aimerais remercier toute ma famille, et plus particulièrement mes parents M'Hamed et Zoubida pour leur amour et leur foi en moi, mes frères Karim et Ahmed, mes soeurs Meriam et Rabea pour leur soutien moral et aussi juste le fait d'être là, qui m'a aidé à continuer durant les moments difficiles, enfin j'aimerais remercier mes nièces et neveux pour leur amour inconditionnel qui a été une source d'inspiration pour moi.

INTRODUCTION

Des mineurs perdus dans une mine que des sauveteurs essaient de secourir dans les dédales et les labyrinthes que forment la mine, ou bien des voleurs poursuivis par des policiers en plein centre d'une grande ville, sans pour autant savoir où ils se trouvent ou bien avec une connaissance minimale, ou alors des agents de sécurité qui essaient de localiser un ou plusieurs hackers dans un réseau pour les empêcher de nuire au service. Ce sont là des scénarios qui reflètent la réalité de tout les jours. Pour s'en sortir, chacun des antagonistes aura besoin d'une bonne stratégie, qui lui permettra d'atteindre son but.

Dans un jeu de poursuite, un nombre défini de poursuivant essaient de capturer un ou plusieurs poursuivis qui se déplacent dans un environnement donné avec une connaissance partielle ou totale de l'un ou l'autre des antagonistes du jeu, avec ou sans contraintes sur le déplacement.

Dans ce mémoire, l'environnement où se déroule le jeu de poursuite est un graphe, les antagonistes, qui seront désignés par *policier* et *voleur*, ont une connaissance totale de la position de chacun d'eux, de plus, il n'y aura aucune contrainte sur leurs déplacements. Les jeux de poursuite dans les graphes ont connu plusieurs développements depuis le jour où ils ont été introduits par Breisch [5] et par la suite par Parsons [24].

Le présent mémoire est divisé en 4 chapitres. Dans le premier chapitre, nous allons présenter certaines notions et définitions de la théorie des graphes que nous utiliserons dans ce mémoire, nous allons aussi aborder des notions empiriques de la théorie des jeux, et voir un peu l'historique de son développement, pour finalement parler de la théorie des jeux de poursuite dans les graphes, et plus précisément les graphes policier-voleur, une des différentes variantes de jeux de poursuite dans

les graphes. Dans le chapitre 2, nous allons voir plus en détail ce qui a été fait dans le domaine des jeux de poursuite sur des graphes spécifiques, à savoir des graphes réflexifs, nous allons aussi étudier un des théorèmes fondamentaux dans ce domaine, théorème qui a été introduit par Nowakowski et Winkler [23], qui reste une référence pour ce type de graphes, ainsi que ses différentes utilisations et variantes, nous allons de plus présenter une caractérisation des graphes policier-gagnants qu'ils soient finis ou non finis. Nous aborderons aussi différents travaux faits sur ce types de graphes. Dans le troisième chapitre, nous allons aborder le cas de types de graphes généraux, i.e., non réflexifs ou partiellement réflexifs. Nous verrons aussi l'algorithme de Hahn et MacGillivray [15], qui permet de dire si un graphe (orienté ou non, avec boucles ou sans) est policier-gagnant ou non, de même nous allons voir l'adaptation de cet algorithme pour des graphes k -policiers, l -voleurs. Enfin, nous allons aborder quelques considérations telles que de voir si le théorème de Nowakowski et Winkler peut être adapté pour des graphes autres que réflexifs, de même, nous essaierons de voir si un graphe réflexif policier-gagnant, le restera en étant non réflexif ou partiellement réflexif, et vice-versa, ainsi que de voir si l'ouverture de sommets réflexifs dans un graphe policier-gagnant conserve cette propriété pour le graphe résultant. Finalement, dans le chapitre 4, nous allons voir l'implémentation de l'algorithme de Hahn et MacGillivray, une discussion sur ses performances, et les résultats obtenus par des tests faits sur des graphes, dont on voulait connaître le nombre de policiers minimum, nécessaire pour un gain (tels que les triplets de Steiner).

Dans ce mémoire, par un graphe policier-gagnant, nous désignons un graphe avec un policier et un voleur, de même, tous les graphes considérés sont simples et finis si non spécifiés autrement.

Chapitre 1

ÉTAT DE L'ART

Introduit par Breisch [5] et Parsons [24], le jeu de poursuite dans les graphes a connu plusieurs variantes. De recherche sur les arêtes au tout début on est passé à la recherche sur les noeuds, introduite par Kirousis et Papadimitriou [18], et enfin à des variantes mixant les deux types de recherche [26].

Le jeu de poursuite, comme son nom l'indique, est un jeu de rôle, qui se joue avec deux acteurs : le poursuivant et le poursuivi, qui se déplacent chacun à tour de rôle. Plusieurs variantes du nom donné à l'un ou l'autre sont apparues, ainsi le premier est souvent désigné sous l'appellation : chercheur, policier, gendarme, chasseur, etc. tandis que le deuxième est désigné par : évadé, voleur, lapin, etc. Durant le jeu, la tâche du poursuivant est de capturer en un temps fini de déplacements le second acteur, tandis que le poursuivi essaie de fuir sa capture si possible indéfiniment, sinon, la retarder le plus longtemps possible.

Dans ce mémoire, nous allons étudier certaines variantes du jeu, introduit en premier par Nowakowski et Winkler [23], ainsi que par Quilliot [25].

Le poursuivant sera appelé le policier, et le poursuivi sera appelé le voleur, le nombre de policiers et de voleurs impliqués dans ce jeu peut être un facteur déterminant pour le gain ou la perte du jeu, et par conséquent sera variable. L'information sera complète dans cette variante de jeu, à savoir les deux acteurs seront au courant de la position de leurs adversaires à chaque moment du jeu, et il n'y'a pas de contraintes sur les déplacements permis.

Dans la suite de ce chapitre, nous allons introduire certaines notions nécessairement vagues sur la théorie des graphes et celle du jeu.

1.1. NOTIONS DE THÉORIE DES GRAPHES

1.1.1. Relation d'ordre

Soient deux ensembles A et B . Le *produit cartésien* de A par B , noté $A \times B$ est l'ensemble de tous les couples ordonnés (a, b) , où $a \in A$ et $b \in B$.

Plus formellement,

$$A \times B = \{(a, b) : a \in A, b \in B\}.$$

Plus généralement, le produit cartésien de n ensembles A_1, A_2, \dots, A_n , noté :

$$\prod_{i=1}^n A_i = A_1 \times A_2 \times \dots \times A_n,$$

est l'ensemble de tous les n -uplets (a_1, \dots, a_n) où $a_i \in A_i$.

On note $\pi_j : \prod_{i=1}^n A_i \leftarrow A_j$, la projection de la j -ième composante, $\pi_j(a_1, \dots, a_n) = a_j$.

Une *relation binaire* \mathcal{R} sur un ensemble A est une partie du produit cartésien de A par lui-même, soit $\mathcal{R} \subseteq A^2 = A \times A$. On note x en relation avec y par $x\mathcal{R}y$ pour signifier $(x, y) \in \mathcal{R}$, et par abus de notation, pour tout sous-ensemble non vide $Y \subseteq A$ (respectivement $X \subseteq A$), $x\mathcal{R}Y$ (respectivement $X\mathcal{R}y$), signifie que $\forall y \in Y$ (respectivement $\forall x \in X$) $x\mathcal{R}y$.

La relation \mathcal{R} est dite :

- *réflexive* si $\forall x \in A, x\mathcal{R}x$;
- *irréflexive* si $\forall x \in A, (x, x) \notin \mathcal{R}$;
- *symétrique* si $\forall x, y \in A, x\mathcal{R}y \Rightarrow y\mathcal{R}x$;
- *antisymétrique* si $\forall x, y \in A, (x, y) \in \mathcal{R} \Rightarrow (y, x) \notin \mathcal{R}$;
- *transitive* si $\forall x, y, z \in A, x\mathcal{R}y$ et $y\mathcal{R}z \Rightarrow x\mathcal{R}z$.

La relation \mathcal{R} forme un *ordre partiel* sur l'ensemble A si elle est réflexive, antisymétrique, et transitive. De plus, si tous les éléments de A sont comparables, i.e $\forall x, y \in A, x\mathcal{R}y$ ou $y\mathcal{R}x$, alors on dit que la relation \mathcal{R} forme un *ordre total* sur A . Enfin, si la relation \mathcal{R} est réflexive, symétrique et transitive, alors on dit que

\mathcal{R} est une *relation d'équivalence*. Une relation d'équivalence \mathcal{R} sur A est dite *triviale* si $\mathcal{R} = A \times A$.

1.1.2. Graphes et sous-graphes

Un *graphe orienté* $G = (V, A)$ est un couple d'ensembles, dont le deuxième, A est une relation binaire sur le premier, V .

L'ensemble V est formé de *sommets*, tandis que A est formé d'*arcs*.

Soit un arc $(x, y) \in A$; on dit que les sommets x, y forment ses *extrémités*, x son extrémité initiale, et y son extrémité finale, on dit aussi que le sommet x *domine* y , et que y est un *successeur* du sommet x , tandis que x est un *prédécesseur* de y .

On dit aussi, que x et y sont *adjacents*, et on note $x \sim y$. Par définition, l'arc (x, y) possède une direction, soit de x vers y . Un arc reliant un sommet à lui-même est appelé *boucle*. Si tout sommet du graphe possède une boucle, c'est-à-dire, si la relation A est réflexive, alors le graphe est dit *réflexif*. De même, si la relation A est symétrique (ou transitive), alors le graphe est dit *symétrique* (ou *transitif*).

Plus formellement :

Un graphe orienté G est dit *symétrique* si,

$$\forall x, y \in V(G) : (x, y) \in A(G) \Rightarrow (y, x) \in A(G)$$

De même, un graphe orienté G est dit *transitif* si,

$$\forall x, y, z \in V(G) : (x, y) \in A(G) \text{ et } (y, z) \in A(G) \Rightarrow (x, z) \in A(G)$$

Un graphe *non-orienté* $G = (V, E)$ est un couple où V est l'ensemble des sommets et E l'ensemble des *arêtes*, que l'on pourrait associer à des arcs sans direction, et qu'on notera $\{x, y\} \in E$. Par analogie avec un graphe orienté, on pourrait dire que si la relation E est symétrique, alors on pourrait identifier une arête aux deux paires (x, y) et (y, x) (formellement, ceci correspond à faire des

arêtes, des classes d'équivalence de la relation $(x, y) \equiv (x', y')$ si et seulement si $x' = x, y' = y$ ou $x' = y, y' = x$. De la même manière, un graphe G non-orienté est dit *réflexif*, si tous ses sommets possèdent une boucle, et *simple*, s'il n'en possède aucune. Si $a = \{x, y\}$ est une arête, les sommets x et y sont dits *adjacents* l'un à l'autre, et l'arête a est dite *incidente* à x et y , de plus si x a une boucle, alors $\{x, x\} = \{x\} \in E$.

Exemple 1.1.1 (Illustration de 3 types de graphes). *La figure 1.1 illustre 3 types de graphes, G est non orienté, H est orienté, tandis que I est réflexif.*

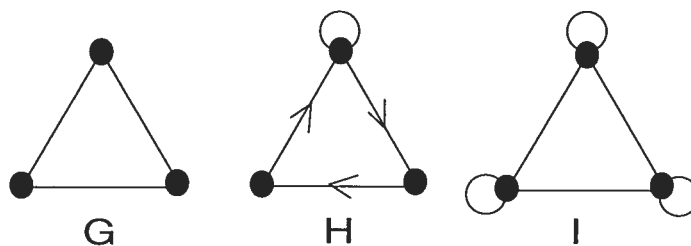


FIGURE 1.1. 3 types de graphes : G est un graphe non orienté, H est un graphe orienté et I est un graphe réflexif

Ainsi, pour un graphe orienté, on peut parler du graphe non-orienté sous-jacent, en passant de (x, y) à $\{x, y\}$ pour tous les arcs, et par comparaison, on peut parler d'un graphe non-orienté comme d'un graphe orienté symétrique correspondant dont chaque arête $\{x, y\}$ est formée des deux arcs (x, y) et (y, x) . Le terme graphe sera employé pour désigner l'un ou l'autre type de graphe, et on les différenciera à partir du contexte. Nous noterons $V(G)$ (respectivement $E(G)$) l'ensemble des sommets du graphe G (respectivement l'ensemble des arcs ou arêtes).

On définira l'*ordre* d'un graphe G , noté $|G|$, comme le cardinal de $V(G)$, soit $|G| = |V(G)|$. Ainsi, le graphe est dit *fini*, si son ordre est fini, sinon, on dit que le graphe est *infini*.

Le graphe, noté \overline{G} , dont l'ensemble des arêtes est le complément de celui de G est appelé le *complément* de G , $\overline{G} = (V(G), V(G)^2 \setminus E(G))$, on dit aussi que les deux graphes sont complémentaires. On remarque que le complément d'un

graphe réflexif est un graphe non réflexif, ainsi, et dépendamment du contexte, on peut ou bien inclure ou exclure les boucles.

Deux graphes simples sur un ensemble de sommets X , d'ordre n , qui sont complémentaires, sont le *graphe vide* et le *graphe complet*.

Un graphe orienté est dit *complet*, si le graphe non-orienté sous-jacent est complet.

On note G^* le graphe orienté symétrique obtenu à partir d'un graphe non orienté G en considérant les deux arcs (x, y) et (y, x) de $A(G^*)$ obtenus de chaque arête $\{x, y\}$ de $E(G)$.

Un graphe H est un *sous-graphe* de G , et on écrira $H \leq G$, si $V(H) \subseteq V(G)$ et $E(H) \subseteq E(G) \cap V(H)^2$. Si, de plus $E(H) = E(G) \cap V(H)^2$, on dit que H est le sous-graphe *induit* de G sur $V(H)$ et on écrira $G \langle V(H) \rangle$ (d'autres notations sont utilisées telles que $G \upharpoonright V(H)$ ou $G[V(H)]$). Autrement dit, le sous-graphe induit de G sur un ensemble X , contient tous les autres sous-graphes de G sur X , i.e $\forall H \leq G, V(H) = X \Rightarrow H \leq G \langle V(H) \rangle$. Si H est un sous-graphe non induit de G , alors il existe un arc $(x, y) \in (E(G) \cap V(H)^2) \setminus E(H)$. Un tel arc s'appelle une *corde* de H dans G . Si $V(H) = V(G)$ alors H est un *facteur*, ou *sous-graphe couvrant* de G .

1.1.3. Voisinage, adjacence

Dans un graphe G non-orienté, on appelle *voisinage* du sommet x , l'ensemble $N_G(x)$ des sommets qui lui sont adjacents, $N_G(x) = \{y \in V(G) : \{x, y\} \in E(G)\}$.

De même, l'ensemble des arêtes incidentes à x est noté :

$$E_G(x) = \{\{x, y\} \in E(G) : y \in N(x)\}.$$

Dans un graphe orienté G on note $N_G^+(x) = \{y \in G : (x, y) \in A(G)\}$ et $N_G^-(x) = \{y \in G : (y, x) \in A(G)\}$, respectivement, le *voisinage sortant* et *voisinage entrant* du sommet $x \in G$. On appelle *voisinage fermé* du sommet x , l'ensemble $N_G[x] = N_G(x) \cup \{x\}$, tandis que $N^+[x] = N^+(x) \cup \{x\}$, et $N^-[x] = N^-(x) \cup \{x\}$ les *voisinages fermé sortant* et *entrant* du sommet x .

Si on considère un graphe non-orienté comme un graphe orienté symétrique, on remarque que $N_G(x) = N_G^+(x) \cup N_G^-(x) = N_G^+(x) = N_G^-(x)$. Si $(x, x) \in A(G)$, alors $N_G^+[x] = N_G^+(x)$ et $N_G^-[x] = N_G^-(x)$. Dans tous les cas, on omet l'indice G quand il n'y a pas de confusion possible.

On appelle le **degré** $d_G(x)$ d'un sommet $x \in V(G)$ dans un graphe $G = (V, E)$, le nombre d'arêtes incidentes à x , c'est-à-dire : $d_G(x) = |E_G(x)| = |N_G(x)|$. Une boucle compte pour deux arêtes.

Par comparaison, dans un graphe G orienté, on appelle **degré sortant** $d_G^+(x)$ (respectivement **degré entrant** $d_G^-(x)$) d'un sommet x , le nombre d'arcs d'extrémité initiale (respectivement d'extrémité finale) x , i.e., les cardinalités des voisinages respectifs.

Le **degré maximum** d'un graphe G est le maximum des degrés des sommets. On le note $\Delta(G) = \max \{d_G(x) : x \in V(G)\}$. De même, pour un graphe G orienté, le degré **maximum sortant** (respectivement **entrant**) est noté $\Delta^+(G)$ (respectivement $\Delta^-(G)$). Dans les deux cas, sauf confusion, on utilise seulement Δ .

Le **degré minimum** d'un graphe G est le minimum des degrés des sommets. On le note $\delta(G) = \min \{d_G(x) : x \in V(G)\}$. De même, le degré **minimum sortant** (respectivement **entrant**) est noté $\delta^+(G)$ (respectivement $\delta^-(G)$). Dans les deux cas, sauf confusion, ici aussi, on utilise seulement δ .

Si tous les sommets du graphe G ont le même degré k , c'est-à-dire $\forall x, y \in V : d_G(x) = d_G(y) = k$, on appelle cette valeur **degré du graphe** et elle est notée $d(G)$, et on dit que G est **régulier**, ou plus précisément **k -régulier**.

Un graphe $G = (V, E)$ est dit **biparti** s'il admet une partition de V en deux sous-ensembles V_1 et V_2 tels qu'aucune arête n'ait les deux extrémités dans le même ensemble V_i , où $i \in \{1, 2\}$.

Si $|V_1| = M$ et $|V_2| = N$ et $\forall u \in V_1, \forall v \in V_2, (u, v) \in E$, le graphe G sera noté $K_{M,N}$.

1.1.4. Connexité

Dans un graphe non-orienté G , une suite $x_0x_1\dots x_k$ de sommets distincts où $\{x_i, x_{i+1}\}$ est une arête dans G pour tout $0 \leq i \leq k - 1$, est appelé un **chemin non-orienté** ou **chaîne** de longueur k .

Un **chemin orienté** de longueur k , dans un graphe orienté G , est une séquence $x_0x_1\dots x_k$ de sommets distincts où (x_i, x_{i+1}) est un arc dans $G \forall i : 0 \leq i \leq k - 1$. Les sommets x_0 et x_k sont appelés les **extrémités** du chemin $x_0x_1\dots x_k$, tandis que les sommets restants sont appelés les **sommets intérieurs**. La **longueur** d'un chemin est le nombre d'arêtes ou d'arcs qui le composent, ou, ce qui revient au même, le nombre de sommets utilisés moins un. si $\{x_0, x_k\} \in E$ (ou $(x_k, x_0) \in A$), on parle plutôt de **cycle** pour un graphe non-orienté et de **circuit** pour un graphe orienté.

Un chemin C entre deux sommets x et y est dit un **plus court chemin**, si sa longueur ne dépasse pas celle de n'importe quel autre chemin reliant les deux sommets.

Soient deux sommets x et y dans un graphe non-orienté G , on appelle **distance** entre x et y , la longueur d'un plus court chemin les reliant, et on la note $d_G(x, y)$. De même, dans un graphe orienté G , la **distance** $d_G(x, y)$ entre deux sommets x et y , est la longueur d'un plus court chemin orienté reliant x à y . Il est à noter que dans un graphe orienté symétrique, la distance d'un sommet x vers un autre sommet y , est la même que celle de y vers x . Par conséquent, on ne fait pas la distinction entre les deux. Par contre, si le graphe orienté n'est pas symétrique, il est possible que $d_G(x, y) \neq d_G(y, x)$.

Le **diamètre** d'un graphe G orienté ou non, noté $diam(G)$, est le maximum des distances entre les sommets, $diam(G) = \max \{d_G(x, y) : x, y \in V(G)\}$.

Si pour tous sommets $x, y \in V(G)$, il existe toujours une chaîne de x vers y ,

on dit que le graphe *connexe*, de même pour un graphe orienté, s'il existe toujours un chemin entre tout paire de sommets, on dit que le graphe est *fortement connexe*, et simplement *connexe* si le graphe non orienté sous-jacent est connexe.

Exemple 1.1.2 (Connexité). *Les deux graphes de la figure 1.2 représentent un graphe connexe G et un deuxième graphe H qui ne l'est pas : il n'existe aucun chemin de 2 à 5, par exemple.*

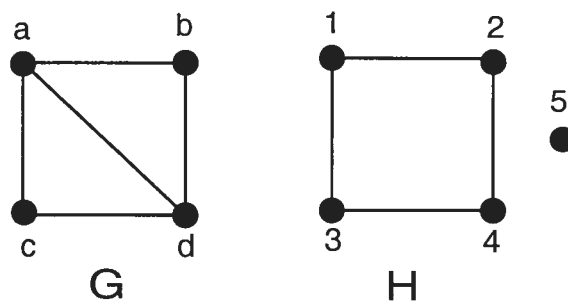


FIGURE 1.2. Représentation d'un graphe connexe G , et un graphe H non connexe

Un *point d'articulation* dans un graphe G est un point dont la suppression sépare le graphe en parties non connexes entre elles.

Exemple 1.1.3 (point d'articulation). *Dans le graphe G de la figure 1.3, on a 3 points d'articulation d , e et f .*

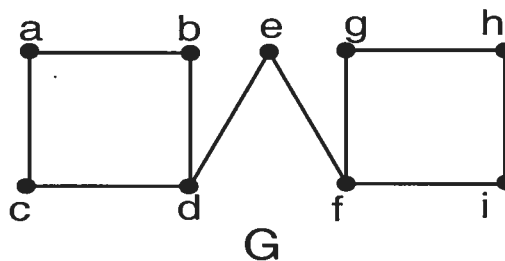


FIGURE 1.3. Illustration de points d'articulation dans un graphe

1.1.5. Groupes, homomorphisme, contraction

Une fonction est un triplet (f, A, B) où :

- A est un ensemble, appelé le *domaine* de la fonction,
- B est un ensemble, appelé le *co-domaine* de la fonction,
- f est une relation de A vers B vérifiant :
 - $\forall a \in A, \exists b \in B$, tel que $(a, b) \in f$;
 - si $a \in A, b_1, b_2 \in B$, et $(a, b_1) \in f$ et $(a, b_2) \in f$, alors $b_1 = b_2$.

Le triplet (f, A, B) est souvent noté $f : A \rightarrow B$. Cette notation signifie que f envoie les éléments de A vers ceux de B . Par abus minime de notation, on pose pour $A \subseteq X$, $f(A) = \{f(a) \mid a \in A\}$.

Une fonction $f : X \rightarrow Y$ est dite *injective*, si $f(x) = f(y)$ implique que $x = y$ pour tout $x, y \in X$.

Une fonction $f : X \rightarrow Y$ est dite *surjective*, si $\forall y \in Y, \exists x \in X$: tel que $f(x) = y$

La fonction f est dite *bijective*, si elle est injective et surjective.

Une conséquence importante de la bijectivité est l'existence d'une fonction inverse.

Soit la fonction bijective $f : A \rightarrow B$, et soit une autre fonction $g : B \rightarrow A$. Si $\forall b \in B, f(g(b)) = b$ alors g est la *fonction inverse* de f , et elle est unique, ce qui permet de la noter f^{-1} . On définit $g(b) = a$ tel que $f(a) = b$.

De plus, une fonction est dite *invertible* si et seulement si elle est bijective.

Ainsi, si $f : X \rightarrow Y$ est bijective alors $\forall A \subset X$, et $\forall B \subset Y$, on a $f \circ f^{-1}(B) = B$ et $f^{-1} \circ f(A) = A$.

Soit $f : X \rightarrow Y$ une fonction. Si A est un sous-ensemble de X , alors la *restriction* de f à A est la fonction $f \upharpoonright A : A \rightarrow Y$ telle que $x \mapsto f(x)$. Un *groupe fini* est un couple $(\Gamma, *)$, où Γ est un ensemble non-vide et $*$ une application de $\Gamma \times \Gamma$ dans Γ , telle que :

- L'application $*$ est *associative* : $\forall a, b, c \in \Gamma, (a * b) * c = a * (b * c)$;
- Elle possède un élément *neutre* e tel que : $\forall a \in \Gamma, a * e = e * a = a$;
- $\forall a \in \Gamma$, il existe un *inverse* a^{-1} pour $*$, tel que : $a * a^{-1} = a^{-1} * a = e$.

L'application $*$ est appelé *opération* du groupe $(\Gamma, *)$ et on parle souvent d'un *produit*.

Le cardinal $|\Gamma|$ de l'ensemble Γ est appelé *ordre du groupe* $(\Gamma, *)$. Un ensemble

$\Gamma' \subset \Gamma$ engendre le groupe $(\Gamma, *)$ si chaque $a \in \Gamma$ peut être exprimé comme un produit d'éléments de Γ' . L'ensemble Γ' est ainsi appelé un *ensemble de générateurs*.

Soient les groupes $(\Gamma, *)$ et (Γ', \bullet) . On appelle un *homomorphisme* de groupes Ψ de $(\Gamma, *)$ dans (Γ', \bullet) une application $\Psi : \Gamma \longrightarrow \Gamma'$ telle que pour tout a et b de Γ , $\Psi(a * b) = \Psi(a) \bullet \Psi(b)$.

Un homomorphisme Ψ est un *isomorphisme*, s'il est bijectif. Si $\Gamma = \Gamma'$, on appelle Ψ un *endomorphisme*. Un endomorphisme bijectif est un *automorphisme*.

Soient G, H deux graphes, l'application $f : V(G) \rightarrow V(H)$ est un *homomorphisme* entre G et H , et noté $f : G \rightarrow H$, si

$$\forall x, y \in V(G)(x, y) \in E(G) \Rightarrow (f(x), f(y)) \in E(H).$$

De façon plus informelle, on dit qu'une fonction f est un homomorphisme de graphe, si elle préserve les arêtes, c'est à dire, si $x, y \in V(G)$ et $x \sim y$, alors $f(x) \simeq f(y)$.

Si de plus f est bijective et si elle, ainsi que son inverse f^{-1} , sont des homomorphismes, alors f est un *isomorphisme*, et on dit alors que G et H sont *isomorphes*, et on note $G \cong H$.

On remarque que si f^{-1} n'est pas un homomorphisme, les graphes peuvent ne pas être isomorphes, contrairement aux groupes : un exemple est celui du passage d'un graphe non complet vers un graphe complet, mais que l'inverse n'est pas vrai.

Si $G = H$, c'est à-dire, si f est un isomorphisme de G avec lui-même, alors on dit que f est un *automorphisme*.

Soient deux graphes G et G' , on dit que G est une *contraction* de G' si on peut passer de G' à G par une suite de *contractions élémentaires*. On appelle ainsi l'opération consistant à choisir une arête (a, b) , à la supprimer en identifiant a et b à un nouveau sommet c , et à faire aboutir à c les arêtes du graphe de départ

qui aboutissaient en a ou en b en éliminant les arêtes multiples.

Formellement, $V' = (V \setminus \{a, b\}) \cup \{c\}$ et $E' = (E \setminus \{\{a, x\}, \{b, x\} \mid x \in V\}) \cup \{\{c, x\} \mid \{a, x\} \in E \vee \{b, x\} \in E\}$.

1.2. NOTIONS DE LA THÉORIE DES JEUX

Les fondements de la théorie des jeux ont été posés par John Von Neumann qui, en 1928, démontra le théorème fondamental du *minimax* (théorème 1.2.1), mais cette théorie ne vit vraiment le jour qu'en 1944 dans la collaboration de John Von Neumann avec Oskar Morgenstern [21], qui ne considèrent pas uniquement les jeux proprement dits, mais également les problèmes de comportement économique et ceux de l'organisation sociale. Pour ces deux problèmes, la théorie des jeux était une source de nombreuses recherches et continue de l'être.

Vers 1950, John Nash a présenté sa vision d'équilibre pour un jeu à plusieurs joueurs, appelée *équilibre de Nash* (ensemble de stratégies tel qu'aucun joueur n'a intérêt à dévier unilatéralement pour améliorer sa position) qui a révolutionné les concepts de la théorie de jeu. Ce résultat a été développé plus tard par Reinhard Selten. La *théorie des jeux* est une branche des mathématiques appliquées qui utilise des modèles pour étudier l'interaction entre des structures formalisées (jeux). Son champ d'application étant extrêmement vaste, la théorie des jeux a commencé par étudier les jeux à deux joueurs. Souvent, il s'agissait de jeux de société, comme les échecs.

1.2.1. Description d'un jeu

N'importe quel jeu peut être décrit de plus d'une façon. En ce qui nous concerne, il est préférable d'utiliser la forme dite *normale*. Toutefois, avant d'expliquer ce qu'il faut entendre par *forme normale*, commençons par expliquer ce que l'on entend par *stratégie*.

Le concept de stratégie est fondamental dans la théorie des jeux. Une *stratégie*

est une description complète de la manière dont on se comportera en toute circonstance possible. Cela ne suppose pas d'habileté particulière. Dans le jeu des petites croix, une stratégie pourrait par exemple consister à jouer dans la colonne qui se trouve le plus à droite possible et à choisir la plus haute des cases qui y restent.

Le nombre de stratégies possibles étant fini, on peut en principe les énumérer. Ainsi, dans le cas du jeu d'échecs, les stratégies pour les pions blancs seront notées par $W1, W2$, etc., et celles pour les pions noirs par $B1, B2$, etc. Puisqu'une stratégie doit rendre compte des mouvements à effectuer dans n'importe quelle situation pouvant survenir, si l'on connaît la stratégie de chaque joueur, on peut prévoir le résultat du jeu.

Supposons qu'avant le commencement du jeu, les joueurs préparent la matrice figurée plus bas (voir tableau 1.1). Ceci est toujours possible, puisqu'un joueur connaît toutes les tactiques possibles dont dispose son adversaire aussi bien que les siennes propres. Les stratégies pour les blancs sont inscrites verticalement à gauche et les tactiques pour les noirs sont inscrites horizontalement au dessus du tableau. À l'intérieur des rectangles formés par l'intersection d'une colonne correspondant à une stratégie pour les noirs et d'une rangée correspondant à une stratégie pour les blancs sont inscrites les lettres G, N , ou P . Par convention, cela signifie que si les joueurs utilisent les tactiques indiquées, les résultats seront soit un gain (G), soit une partie nulle (N), soit une perte (P) pour les blancs. ainsi dans la matrice ci-dessous, si les noirs choisissent $B3$ et les blancs $W4$, alors les blancs perdront.

On dit qu'un jeu est sous la *forme normale* quand toutes les séries de décisions qui peuvent être prises au cours du jeu peuvent être regroupées en une décision unique : le choix d'une stratégie. Dans un seul jeu réel, tel que les dames ou les échecs, un tel procédé de regroupement est purement conceptuel. En pratique, le jeu est en fait joué sous la *forme extensive*, c'est-à-dire, que les décisions sont véritablement prises l'une après l'autre.

		Stratégies pour les noirs			
		<i>B1</i>	<i>B2</i>	<i>B3</i>	<i>B4</i>
Stratégies pour les blancs	<i>W1</i>	<i>G</i>	<i>N</i>	<i>P</i>	<i>P</i>
	<i>W2</i>	<i>N</i>	<i>G</i>	<i>G</i>	<i>P</i>
	<i>W3</i>	<i>P</i>	<i>P</i>	<i>G</i>	<i>G</i>
	<i>W4</i>	<i>N</i>	<i>N</i>	<i>P</i>	<i>G</i>

TABLEAU 1.1. La représentation matricielle du jeu d'échecs

En fait, si on doit définir un *jeu*, on pourrait dire qu'il s'agit d'un ensemble de règles définissant les gains et les pertes d'individus rationnels (les joueurs), suivant les stratégies qu'ils choisissent.

À chaque coup ou tour du jeu, chaque joueur peut choisir entre des actions définies par le jeu. Suivant le type de jeu, ces actions peuvent être consécutives : le joueur *A* joue, puis *B*, puis de nouveau *A* (échecs, bridge), ou simultanées : au premier coup, *A* et *B* jouent, aucun des deux ne connaissant le choix de l'autre, idem au deuxième coup (poker). De façon plus formelle, on pourrait représenter les actions prises par les antagonistes dans un jeu par un arbre : chaque noeud est associé au joueur qui décide, et chaque option constitue une branche. Les gains de tous sont associés aux terminaisons, lorsqu'il est possible de les représenter (fins du jeu).

Le gain d'un joueur *X* pour un tour est fonction de l'action de *X* mais également des actions des autres joueurs. Les choix rationnels dépendent fortement des règles du jeu et on distingue alors le cas d'information complète : chacun des participants connaît ses possibilités d'action, l'ensemble des choix des autres joueurs, toutes les issues possibles et les gains qui y sont associés, les motifs des joueurs et les siens propres, et le cas d'information incomplète.

On suppose généralement que la perspective est non coopérative, c'est-à-dire, que chacun cherche à maximiser son propre gain et le sien seulement - il existe cependant une théorie des jeux coopératifs où les joueurs peuvent passer des accords entre eux : former une coalition, où le non-respect de ces accords est sanctionné.

La théorie des jeux classe les jeux en catégories en fonction de leurs approches de résolution.

- Jeux de stratégie à *somme nulle* et *non nulle* :

- Les jeux *à somme nulle* sont les jeux où la somme des gains des joueurs est constante, c'est-à-dire, ce qu'un joueur gagne, l'autre le perd nécessairement. Les échecs ou le poker sont des jeux à somme nulle.
- Le *dilemme du prisonnier*, les *situations d'affaires*, ou la *vie politique* sont des jeux à *somme non-nulle* car certaines issues sont globalement plus profitables pour tous, ou plus dommageables pour tous.
- Jeux *déterminé* et *indéterminé*
 - Par jeu *déterminé*, il faut comprendre que quelles que soient les actions des joueurs, il y'aura toujours un gagnant et un perdant.
 - Par jeu *indéterminé*, il faut comprendre que l'issue peut être une partie nulle.

Théorème 1.2.1 (Théorème du minimax). *Tout jeu à somme nulle et à deux joueurs, qui ont fait leurs choix dans des ensembles finis de stratégies pures, comporte au moins un équilibre en stratégies mixtes.*

Une *stratégie mixte* est une distribution de probabilités affectée par un joueur à l'ensemble des *stratégies pures*, sachant que les stratégies pures sont des variables certaines.

Le théorème de Zermelo (1912) pour un jeu à issue indéterminé, justifie l'introduction du concept mathématique de jeu à deux joueurs de somme nulle, i.e. tout jeu opposant deux joueurs, sans intervention du hasard, sans possibilité de bluff, avec un nombre fini de coups possibles (par exemple les échecs) possède une stratégie permettant à l'un des deux joueurs de gagner à coup sûr ; par contre, ce théorème ne dit rien quant à la forme de cette stratégie, ni quant un des joueurs gagne. Ainsi, ce théorème peut s'énoncer comme suit :

Théorème 1.2.2. *Au jeu d'échec, ou bien les blancs peuvent gagner, ou bien les noirs peuvent gagner à coup sûr, ou bien chaque joueur peut assurer la partie nulle (i.e. chacun peut garantir qu'il ne perde jamais).*

A priori, ce théorème paraît trivial, il montre que l'issue d'un jeu, tel que celui des échecs, est indéterminée, même si les joueurs sont rationnels. Un autre

théorème de la théorie des jeux de Zermelo concernant les jeux déterminés, sujet de ce mémoire est celui des jeux finis à information complète, et à somme nulle joués à deux, et qui peut s'énoncer comme suit :

Théorème 1.2.3. *Un jeu à deux acteurs, fini, et à information complète et à somme nulle est complètement déterminé.*

En définitif, dans le cadre de la théorie des jeux, il semble difficile d'appliquer des critères scientifiques traditionnels. En effet, cette théorie s'applique à divers domaines, plusieurs sciences : biologie de l'évolution, sciences économiques, stratégie militaire, etc.

Enfin, bien que rencontrant des limites non négligeables, la théorie des jeux n'en demeure pas moins une avancée dans la conceptualisation des relations individuelles ; elle essaie d'y répondre, elle tente de les résoudre et soulève de nouveaux problèmes.

1.2.2. Jeux de poursuite

Les jeux de poursuite ou d'évasion, sont parmi les problèmes fondamentaux de plusieurs axes de recherche, tels que l'informatique, la théorie de jeux, ou la théorie de contrôle. Le but dans un jeu de poursuite est, pour le poursuivant, de trouver une stratégie lui permettant de capturer le poursuivi en temps fini, alors que le poursuivi lui, doit essayer d'éviter sa capture indéfiniment. Il y'a plusieurs variantes de jeux de poursuite, basées sur :

- L'environnement où se joue le jeu : par exemple les graphes ;
- L'information disponible aux joueurs : est-elle complète ? c'est-à-dire, que les joueurs connaissent les positions de leurs adversaires à tout moment, ou bien est-elle partielle ? par exemple, les joueurs connaissent la position de leurs adversaires qui se trouvent juste dans leurs voisinages immédiats ;
- La contrainte sur le déplacement des joueurs : est-ce qu'il y'a une limite sur la vitesse de leurs déplacements ? ou bien une limite sur leurs manières de se déplacer ?

- La définition de capture : est-ce que le poursuivi doit être intercepté, vu, ou bien encerclé pour parler de capture ?
- La contrainte sur le nombre de joueurs : le nombre de poursuivant, et celui des poursuivis, est-il limité ?

Antérieurement, l'étude sur les jeux de poursuite était motivée par les contrôles de tâches tels que l'interception de missiles [3]. Dans cette perspective, le déplacement du poursuivi était modélisé par un processus stochastique. Toutefois, il y'a eu récemment, un intérêt croissant pour la modélisation de jeux où le poursuivi est plus "intelligent" [27]. Les jeux de poursuite dans les graphes [24, 23, 20, 19, 6, 1] ont été étudiés, non seulement pour leurs applications dans la sécurité informatique et l'architecture de protocoles [4, 7], mais aussi pour leurs relations avec les propriétés fondamentales des graphes telles que les points d'articulation [13]. Dans les différentes variantes de jeux de poursuite, différents noms ont été donnés aux protagonistes tels que poursuivant-poursuivi, monstre-princesse, chasseur-lapin, shérif-voleur, ou bien policier-voleur. Dans ce mémoire, nous allons adopter le terme policier-voleur pour nommer les joueurs, le graphe sera bien sûr fini, même si des travaux existent traitant des graphes infinis [14], l'information sera complète, il n'y aura pas de contraintes sur le déplacement des joueurs, ainsi que sur le nombre des joueurs, et la capture se fera quand les deux protagonistes se trouveront au même moment sur le même sommet.

1.2.2.1. *Le jeu policier-voleur*

Le jeu du *policier-voleur* est un jeu de stratégie qui se déroule dans un graphe, et où les joueurs se déplacent à tour de rôle. Le premier joueur, à savoir le policier, choisit un sommet dans le graphe, et s'y place. Après, le deuxième joueur, à savoir le voleur, choisit aussi un sommet où il se place. Ensuite, à tour de rôle, chacun se déplace d'un sommet vers un sommet voisin (si une boucle est présente sur un sommet, le joueur peut s'en servir pour rester sur place). Le but pour le policier est de capturer le voleur, c'est-à-dire, se trouver en même temps que le

voleur sur le même sommet, tandis que celui du voleur est de fuir indéfiniment le policier. Comme le jeu est à information complète, les deux protagonistes sont au courant de la position courante de chacun d'eux, ainsi que des déplacements effectués pour y arriver. Le policier gagne s'il arrive à capturer le voleur en un temps fini, quels que soient les déplacements du voleur. Si le policier peut gagner, on dit que le graphe est *policier-gagnant*. Si le voleur peut indéfiniment fuir sa capture, quelle que soit la stratégie du policier, alors le voleur gagne, et on appelle le graphe *voleur-gagnant*. Le jeu est donc à somme nulle, puisqu'il est complètement déterminé, c'est à dire, qu'il y'aura toujours un gagnant et un perdant.

Plus formellement, et ceci est une conséquence logique du théorème 1.2.3, nous pouvons formuler le théorème suivant qui paraît trivial, mais, reste important :

Théorème 1.2.4. *Pour tout graphe fini G , il existe toujours une stratégie gagnante soit pour le policier, soit pour le voleur.*

DÉMONSTRATION. Soit un graphe fini G . On suppose qu'il n'y a pas de stratégie gagnante pour le policier. Si le policier ne peut pas gagner, cela veut dire que le voleur peut fuir sa capture indéfiniment, donc le voleur gagne.

Si le voleur ne peut pas gagner, cela veut dire qu'à un moment donné quelconque, le policier le capturera, et donc le policier gagne. \square

Comme le nombre de policiers dans un jeu peut être supérieur ou égal à 1 (c'est le cas aussi pour le nombre de voleurs), on définit la notion de copnumber.

Définition 1.2.1. *Par copnumber d'un graphe, on entend le nombre minimum de policiers nécessaire pour capturer un voleur dans ce graphe.*

Chapitre 2

GRAPHES POLICIER-GAGNANTS RÉFLEXIFS

La première fois que le jeu policier-gagnant a été proposé, il était joué par un seul policier et un seul voleur. N'importe quel graphe peut être étiqueté policier-gagnant ou voleur-gagnant, une conséquence du théorème 1.2.4. Les graphes policier-gagnant réflexifs ont été complètement caractérisés dans [23, 25]. Ce sont uniquement les graphes réflexifs que l'on considère dans ce chapitre.

Exemple 2.0.1 (Graphe réflexif policier-gagnant). *Soit le graphe G (voir figure 2.1). Il est policier-gagnant : le policier se place sur le sommet P , ce qui lui permettra de capturer le voleur au coup suivant quelque soit la position initiale du voleur.*

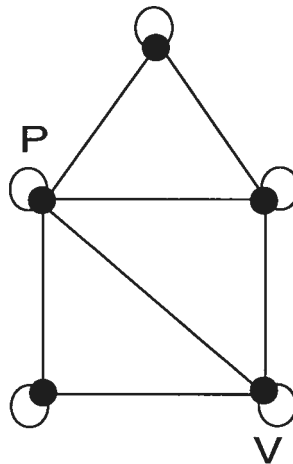


FIGURE 2.1. Graphe réflexif policier-gagnant

Exemple 2.0.2 (Graphe réflexif voleur-gagnant). Soit un graphe G suivant (voir figure 2.2). Il est voleur-gagnant, car quelle que soit la position initiale du policier, le voleur peut toujours se placer 2 sommets plus loin, et ainsi il peut toujours laisser 2 arêtes entre lui et le policier quelle que soit la stratégie du policier.

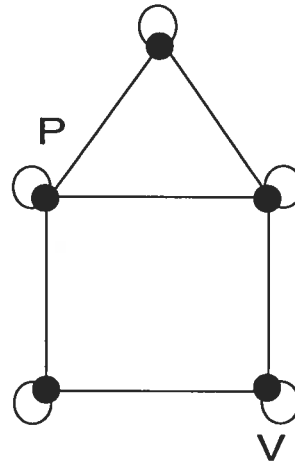


FIGURE 2.2. Graphe réflexif voleur-gagnant

Définition 2.0.2. Soit le graphe réflexif G , et soit H un sous-graphe induit de G . On dit que H est une **rétraction** de G si il existe un homomorphisme f de G vers H tel que la restriction de f à H $f \upharpoonright H$ est la **fonction identité** sur H .

Le type d'argument de la démonstration du théorème qui suit a été utilisé tellement de fois, qu'il sera transcrit tel quel.

Théorème 2.0.5. (Nowakowski et Winkler [23]) Soit H une rétraction d'un graphe G , si G est policier-gagnant alors H est aussi policier-gagnant.

DÉMONSTRATION. Soient G un graphe policier-gagnant et H sa rétraction. Soit f la fonction rétraction de G vers H . Étant donné que G est policier-gagnant, le policier a une stratégie gagnante dans G . Cette stratégie peut être modifiée et utilisée dans le sous-graphe H de la manière suivante : le policier joue l'image sous f de sa stratégie dans G . Ainsi, le policier capturera l'image du voleur dans H . Étant donné que le voleur se déplace dans H et que f est la fonction identité

sur H , l'image du voleur coïncide alors avec sa position actuelle. Ainsi, le voleur est capturé dans H , d'où H est policier-gagnant. \square

Remarque 2.0.1. *On remarquera qu'un sous-graphe induit H d'un graphe G policier-gagnant, n'est pas toujours policier-gagnant.*

Exemple 2.0.3. *Soient les deux graphes G et H suivants (voir figure 2.3), on a G est policier-gagnant : le policier se place au centre et il est capable de capturer le voleur au prochain tour, quelle que soit la position du voleur. Pourtant le graphe H est voleur-gagnant : il y'aura toujours 2 arêtes entre le voleur et le policier.*

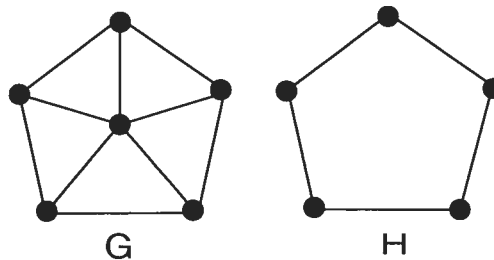


FIGURE 2.3. G est policier-gagnant, alors que le sous-graphe induit H est voleur-gagnant

Remarque 2.0.2. *On remarque que la rétraction d'un graphe voleur-gagnant peut être policier-gagnant.*

Exemple 2.0.4. *Soit un graphe policier-gagnant (voir figure 2.4). La rétraction de ce graphe donne un graphe policier-gagnant.*

2.1. CARACTÉRISATION DE GRAPHES POLICIER-GAGNANTS FINIS

Définition 2.1.1. *On dit qu'un sommet u du graphe G domine un sommet v , si $N[v] \subseteq N[u]$.*

Supposons que le graphe G est policier-gagnant. Afin de déterminer les propriétés qui caractérisent un tel graphe, il est important de considérer le dernier déplacement fait par le voleur avant sa capture. Soit donc v le dernier sommet où le voleur se trouve avant sa capture. Trois déplacements sont possibles : il peut soit rester sur place (le graphe est réflexif), soit il peut se déplacer vers le sommet

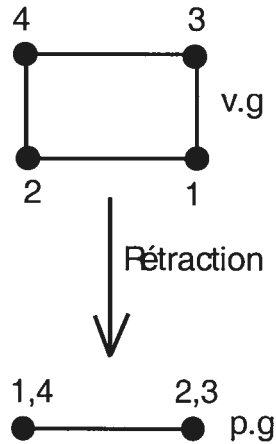


FIGURE 2.4. Le graphe du départ est voleur-gagnant, alors que sa rétraction est policier-gagnante

occupé par le policier, ou bien, il peut se déplacer vers un sommet adjacent à celui du policier. Du moment que tous ces déplacements mènent à une capture du voleur, il est évident que le sommet u où le policier se trouve est adjacent à v et à n'importe quel autre sommet qui lui est adjacent, c'est à dire, (définition 2.1.1) que u domine v , du moment que le voleur n'a aucune échappatoire une fois forcé à se déplacer vers v .

Définition 2.1.2. Soit un graphe G , et soit un sommet $v \in V(G)$. Si il existe un sommet $u \in V(G)$ tel que $N[v] \subseteq N[u]$, on appelle v *irréductible*, *coin* ou bien *dominé*, et u le *dominant*, et on dit alors que u *domine* v .

Ainsi, un graphe sans coins ne peut pas être policier-gagnant. Supposons que le graphe G a un coin v . Le voleur ne se déplacera vers ce sommet que s'il y est forcé. Cela est dû au fait que n'importe quelle stratégie voleur-gagnant qui utilise v peut être modifiée pour utiliser u . La question qui se pose alors, est : est-ce que le policier peut forcer le voleur dans le coin. Répondre à cette question, revient à enlever le coin, et à déterminer si le graphe résultant est policier-gagnant. Intuitivement, l'enlèvement successif de coins d'un graphe policier-gagnant donne un sommet. C'est le principe utilisé par Nowakowski et Winkler [23] pour caractériser les graphes policier-gagnant.

Théorème 2.1.1. (Nowakowski et Winkler [23]) *Soit un graphe G , et soit le sommet c un coin de G . Posons $G' = G \setminus \{c\}$. Le graphe G est policier-gagnant si et seulement si G' est policier-gagnant.*

DÉMONSTRATION. Soit le sommet c un coin du graphe G , et posons $G' = G \setminus \{c\}$. D'après la définition du coin, il existe un sommet d qui domine c . Le graphe G' est une rétraction de G , du fait qu'il existe une fonction rétraction f définie par : $f(c) = d$ et $\forall v \in V(G'), f(v) = v$. Par le théorème (2.0.5), si G est policier-gagnant, alors G' l'est aussi.

Supposons maintenant que G' est policier-gagnant. Le policier a alors une stratégie gagnante qui lui permettra de capturer le voleur. De plus, étant donné que le jeu se joue dans G , la stratégie gagnante du policier dans G' peut être vue comme une capture de l'image du voleur. Supposons maintenant que cette image est capturée sur un sommet u . Si $u \neq d$, alors l'image du voleur dans G' correspond à sa position actuelle sur G étant donné que f est la fonction identité sur G' . Ainsi, le voleur sera capturée, sinon, l'image du voleur est capturée sur d . Comme $f(c) = f(d) = d$, le voleur est soit sur le sommet c , soit sur d dans le graphe G . S'il est sur d , alors sa position actuelle correspond à celle de son image, et il est capturé, sinon, s'il est sur c , alors il sera capturé au prochain déplacement du policier, du fait que le policier est sur d , et que d domine c . \square

Définition 2.1.3. *Un graphe G est appelé **démontable** si ses sommets peuvent être ordonnés linéairement $\{v_1, v_2, \dots, v_n\}$ tels que $\forall \beta < n$, v_β est irréductible dans le sous-graphe induit par $\{v_\beta, v_{\beta+1}, \dots, v_n\}$.*

Le théorème suivant est dû à Nowakowski et Winkler [23] et à Quillot [25].

Théorème 2.1.2. (Nowakowski et Winkler [23]) *Un graphe fini est policier-gagnant si et seulement si il est démontable.*

DÉMONSTRATION. La démonstration de ce théorème reprend celle due au théorème 2.1.1 en enlevant chaque coin, et en appliquant ce théorème par récurrence. \square

Remarque 2.1.1. *L'ordre linéaire des sommets du graphe G de la définition 2.1.3 est appelé **ordre policier-gagnant**.*

Exemple 2.1.1 (Le démontage d'un graphe). *Le graphe représenté par la figure 2.5 est policier-gagnant. Les sommets encerclés sont des coins à chaque étape de démontage. À la fin, on obtient un seul sommet.*

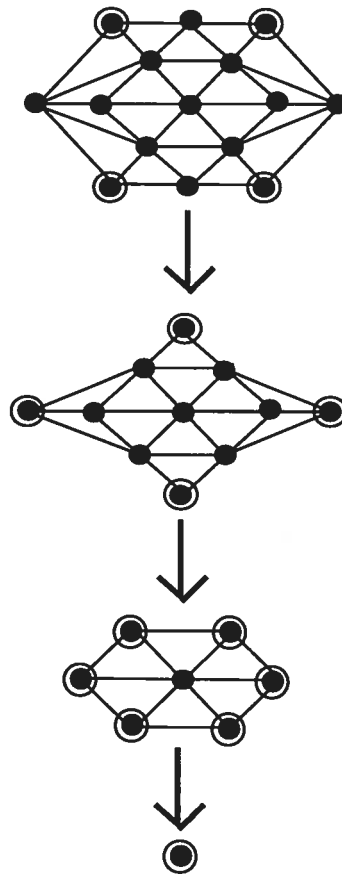


FIGURE 2.5. Les étapes de démontage d'un exemple de graphe policier-gagnant

2.2. CARACTÉRISATION DE GRAPHES NON FINIS POLICIER-GAGNANTS

Il existe aussi des graphes policier-gagnants non finis, Nowakowski et Winkler dans [23] ont étendu leur caractérisation pour englober et les graphes non finis. Ainsi, pour un graphe arbitraire G , il faut définir pour chaque ordinal α une

relation binaire \leq_α dans $V(G)$ récursivement. Pour deux sommets $x, y \in V(G)$, on met $x \leq_0 y$ si et seulement si $x = y$. Pour $\forall \alpha > 0$, on pose $x \leq_\alpha y$ si et seulement si $\forall u \in N[x]$, il existe un sommet $v \in N[y]$ et un $\rho < \alpha$ tel que $u \leq_\rho v$. Finalement, soit α' le plus petit ordinal tel que $\leq_{\alpha'} = \leq_{\alpha'+1}$, et définissons \leq comme $\leq_{\alpha'}$ (α' existe, voir plus loin).

Théorème 2.2.1 (Graphes finis ou non). *Le graphe G est policier-gagnant si et seulement si la relation \leq est triviale, c'est à dire, que : $x \leq y, \forall x, y \in V(G)$.*

DÉMONSTRATION. Avant de démontrer ce théorème, notons certaines propriétés de la relation \leq_α :

- Premièrement, si $p < \alpha$, alors $\leq_p \subseteq \leq_\alpha$. Cela provient de la définition par induction, et elle garantit l'existence de l'ordinal α' . En fait $\alpha' \leq n(n-1)$ si G est fini et possède n sommets ; sinon $\alpha' \leq |V(G)|$.
- Deuxièmement, chaque \leq_α est un quasiordre, i.e., est réflexif et transitif. La réflexivité est immédiate puisque chaque \leq_α contient \leq_0 ; supposons que la transitivité est vraie pour tout $p < \alpha$, et supposons que $x \leq_\alpha y$ et $y \leq_\alpha z$. Soit $u \in N(x)$ et trouvons $v \in N(y)$ et $p < \alpha$ tel que $u \leq_p v$; ensuite, trouvons $w \in N(z)$ et $p' < \alpha$ tel que $v \leq_{p'} w$. Alors $u \leq_{\max\{p, p'\}} w$ et ainsi $x \leq_\alpha z$.

Maintenant, soit α' le plus petit ordinal tel que $\leq_{\alpha'} = \leq_{\alpha'+1}$. Supposons que \leq est triviale, et soit le sommet x_0 la position initiale du policier. Soit y_1 le sommet de départ pour le voleur ; étant donné que $y_1 \leq_{\alpha'} x_0$, alors il existe un sommet $x_1 \in N(x_0)$ et un $\alpha(1) < \alpha'$ tel que $y_1 \leq_{\alpha(1)} x_1$. En général, après le i -ième déplacement du policier, il se trouve sur le sommet x_i et le voleur sur y_i tel que $y_i \leq_{\alpha(i)} x_i$. Quand le voleur se déplace vers $y_{i+1} \in N(y_i)$, il existe un sommet $x_{i+1} \in N(x_i)$ tel que $y_{i+1} \leq_{\alpha(i+1)} x_{i+1}$ et $\alpha(i+1) < \alpha(i)$; le policier se déplace vers x_{i+1} .

Étant donné que les $\alpha(i)$ constituent une séquence strictement décroissante d'ordinaux, cette séquence doit s'arrêter après un nombre fini t de déplacements ; ainsi $\alpha(t) = 0$, $x_t = y_t$ et le policier a gagné.

Réciproquement, supposons que $y_0 \not\leq x_0$ pour une paire de sommets $x_0, y_0 \in$

$V(G)$. Supposons pour le moment que le policier est obligé de commencer sur le sommet x_0 ; le voleur commence sur un sommet $y_1 \in N(y_0)$ tel que pour tout $x \in N(x_0)$, $y_1 \not\leq x$ (si un tel y_1 n'existe pas, alors par définition $y_0 \leq_{\alpha'+1} x_0$, une contradiction). En procédant de cette manière, le voleur garantit que pour tout i il existe un sommet $y_{i+1} \in N(y_i)$ tel que pour tout $x \in N(x_i)$, $y_{i+1} \not\leq x$; x_i et y_i sont les positions respectives du policier et du voleur après le i -ième déplacement du policier. Étant donné qu'il y a toujours un $y_{i+1} \notin N(x_i)$, le voleur ne sera jamais capturé.

Il reste juste à noter que si le policier a une stratégie gagnante nécessitant une position initiale sur le sommet v , alors le graphe G est certainement connecté. Par conséquent, si le policier est forcé de commencer sur x_0 , il peut simplement se déplacer vers v , et gagner à partir de là. \square

2.3. STRATÉGIE POUR UN GRAPHE POLICIER-GAGNANT

Dans cette section, nous allons décrire une stratégie gagnante, en plus de prouver que cette stratégie est effective dans la capture du voleur.

2.3.1. Stratégie policier-gagnante (Clarke et Nowakowski [11, 10])

Soit $\{x_1, x_2, \dots, x_n\}$ l'ordre policier-gagnant d'un graphe G . Définissons le sous-graphe induit $G_i = G_{i-1} \setminus \{x_{i-1}\}$, où $G_1 = G$, et soit $f_i : G_i \rightarrow G_{i+1}$ la rétraction qui envoie x_i vers un sommet qui le domine dans G_i . De plus, si le voleur est sur le sommet x , définissons $F_i(x) = f_{i-1} \circ f_{i-2} \circ \dots \circ f_2 \circ f_1(x)$. Alors $F_i(x)$ est l'image du voleur dans G_i . Le voleur est toujours considéré jouant dans G . Initialement, le policier se déplace dans le sous-graphe G_n , en débutant sur le sommet x_n , sommet qui coïncide avec l'image du voleur sous $F_n(x)$. Supposons après que le policier occupe l'image du voleur dans le sous-graphe G_i sous $f_{i-1} \circ f_{i-2} \circ \dots \circ f_2 \circ f_1(x)$. Le policier se déplace alors vers l'image du voleur dans G_{i-1} .

Le voleur est capturé après n déplacements au plus, comme le prouve le théorème suivant.

Théorème 2.3.1. (Clarke et Nowakowski [11]) *Soit un graphe policier-gagnant G tel que $|V(G)| = n$. En jouant une stratégie policier-gagnante, le policier sera en mesure de capturer le voleur après n déplacements au plus.*

DÉMONSTRATION. Notons d'abord que pour tout i , f_i est un homomorphisme. Ainsi, quand le voleur se déplace de x vers y , on a pour tout j , $F_j(x) = F_j(y)$ ou $F_j(x) \sim F_j(y)$. Notons aussi que quelque soit i , $F_i(x)$ et $F_{i-1}(x)$ sont, soit le même sommet, ou bien deux sommets adjacents.

La preuve se fait par récurrence. Le policier commence sur le sommet x_n qui coïncide avec l'image du voleur sous $f_{n-1} \circ f_{n-2} \circ \dots \circ f_2 \circ f_1$, c'est à dire, que le policier se place sur le même sommet que $F_n(x)$ où il peut rester sur place au prochain tour. Supposons maintenant que pour un $i \leq n$, le policier capture $F_i(y)$, où y est la position du voleur dans G , et que c'est au tour du voleur de se déplacer. Supposons qu'il se déplace sur un sommet z . On doit démontrer que $F_i(y) \sim F_{i-1}(z)$ pour qu'au prochain déplacement du policier, il puisse capturer l'image du voleur dans G_{i-1} . Deux cas se présentent : si $F_i(y) = F_{i-1}(y)$ alors $F_{i-1}(z) \neq F_i(y)$. Sinon $F_i(y) \sim F_{i-1}(y)$. Pourtant $F_{i-1}(y)$ est le coin enlevé de G_{i-1} afin d'obtenir G_i , d'où $N[F_{i-1}(y)] \subset N[F_i(y)]$ et par conséquent $F_i(y) \sim F_{i-1}(z)$. Ainsi et dans tous les cas, l'image du voleur peut être capturée avec un seul déplacement sur le graphe G .

Étant donné qu'il y'a un nombre fini n de graphes G_i , alors l'image du voleur coïncidera avec sa position actuelle après au plus n déplacements. \square

Il a été montré que si le policier joue dans le sous-graphe G_i et s'il occupe l'image du voleur sous F_i , alors le policier peut se déplacer vers l'image du voleur dans G_{i-1} sous $f_{i-2} \circ f_{i-3} \circ \dots \circ f_2 \circ f_1$. Si le policier se déplace dans le sous-graphe G_i , alors le voleur ne peut pas se déplacer vers un sommet de ce sous-graphe

sans être capturé par le policier, c'est pour cette raison que dans [9], la stratégie policier-gagnante est appelée *stratégie non-retour*.

Théorème 2.3.2. (Clarke et Nowakowski [10]) *Supposons que le policier joue la stratégie gagnante dans le sous-graphe G_i , et qu'il occupe l'image du voleur sous $f_{i-1} \circ f_{i-2} \circ \dots \circ f_2 \circ f_1$. Le voleur ne peut pas se déplacer vers un sommet de G_i sans que le policier s'y déplace aussi.*

DÉMONSTRATION. Supposons que le policier joue la stratégie gagnante dans le sous-graphe G_i , et qu'il occupe l'image du voleur sous $f_{i-1} \circ f_{i-2} \circ \dots \circ f_2 \circ f_1$. Le policier peut se déplacer de telle manière à être avec l'image du voleur dans le sous-graphe G_i . De plus, $f_{i-1} \circ f_{i-2} \circ \dots \circ f_2 \circ f_1$ est la fonction identité dans G_i . Ainsi, si le voleur se déplace vers un sommet de G_i , alors son image correspondra à sa position actuelle et il sera capturé. \square

2.4. GRAPHE À PONT

Définition 2.4.1. *Soit un graphe G , et soit H un de ses sous-graphe. On dit que H est un sous-graphe *isométrique* si la distance entre chaque paire de sommets dans H est la même que celle dans G .*

Remarque 2.4.1. *Un sous-graphe isométrique de G est en fait un sous-graphe induit.*

Définition 2.4.2. *Un graphe est appelé *graphe à pont* s'il ne contient aucun cycle isométrique de longueur supérieur à 3.*

Exemple 2.4.1 (Graphe à pont). *Le graphe J est un graphe à pont, le graphe H est un sous-graphe isométrique de G mais pas de I (voir figure 2.6).*

En fait, la définition 2.4.2 veut dire que s'il y'a un cycle C dans un graphe à pont dont la longueur est supérieur à 3, alors il y'a un raccourci entre une paire de sommets dans C .

Anstee et Farber dans [2] ont prouvé que n'importe quel graphe à pont non trivial contient un coin (théorème 2.4.1).

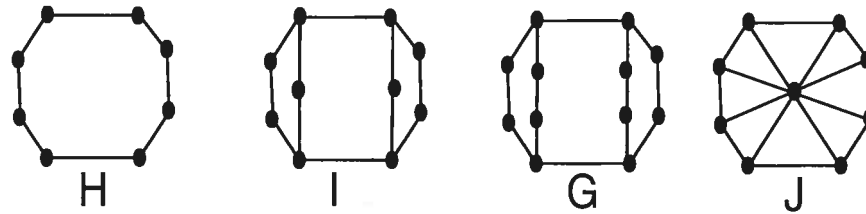


FIGURE 2.6. Exemples de graphes à pont et sous-graphes isométriques

Théorème 2.4.1. (Anstee et Farber [2]) *Chaque composante non triviale d'un graphe à pont contient une paire de sommets u et v telle que $N[u] \subseteq N[v]$.*

Et à partir de ce fait, ils ont prouvé le théorème suivant :

Théorème 2.4.2. (Anstee et Farber [2]) *Soit G un graphe à pont. Il existe un sommet $u \in V(G)$ tel que $G \setminus \{u\}$ est un graphe à pont.*

DÉMONSTRATION. Soit G un graphe à pont. Choisissons une paire de sommets $u, v \in V(G)$ telle que $N[u] \subseteq N[v]$. Soit P le plus court chemin dans G qui contient u et où u n'est pas une feuille ; u peut être remplacé par v dans P . Ainsi $G \setminus \{u\}$ est un sous-graphe isométrique de G . Il est à noter qu'un cycle C est isométrique dans $G \setminus \{u\}$ si et seulement si il est isométrique dans G . D'où $G \setminus \{u\}$ est un graphe à pont. \square

Dans la démonstration donnée du théorème 2.4.2, le sommet u est un coin. Ainsi, ce théorème, en fin de compte, nous apprend que le résultat de soustraction d'un coin d'un graphe à pont reste toujours un graphe à pont.

D'après les théorèmes vus jusqu'à présent, on a quelques informations assez importantes :

- La soustraction d'un coin d'un graphe à pont reste toujours un graphe à pont ;
- Un graphe policier-gagnant est un graphe dont la soustraction successive de coins donne un sommet ;

qui nous mènent à la conclusion suivante :

Théorème 2.4.3. *Tout graphe à pont est un graphe policier-gagnant.*

La démonstration de ce théorème a été donnée par Anstee et Farber [2]. Une preuve algorithmique est due aussi à Chepoi dans [8]. Les deux preuves utilisent l'ordre policier-gagnant introduit par Nowakowski et Winkler [23].

Il est à noter que Anstee et Farber [2] ont prouvé un résultat encore plus intéressant :

Théorème 2.4.4. *Un graphe est à pont si et seulement si il est policier-gagnant sans cycles induits de longueur 4 et 5.*

Chapitre 3

GRAPHES GÉNÉRAUX

Dans ce chapitre, nous allons étudier des graphes non réflexifs avec des boucles sur certains sommets mais pas nécessairement sur tous les sommets. Les graphes sont toujours des graphes finis, et ils peuvent être orientés comme non orientés. À partir des définitions 2.1.2 et 2.1.3, le théorème 2.1.2 de Nowakowski et Winkler [23] peut être reformulé de la manière suivante.

Théorème 3.0.5. *(Nowakowski, Winkler, et Quillot) Un graphe non orienté réflexif G est policier-gagnant si et seulement si ses sommets peuvent être linéairement ordonnés v_1, v_2, \dots, v_n de telle façon que $\forall i \in \mathbb{N}, \exists j > i$ tel que $N_i[v_i] \subseteq N_i[v_j]$.*

DÉMONSTRATION. Supposons que G est policier-gagnant. Cela implique qu'il existe une stratégie permettant au policier c de gagner en un temps fini. Il existe donc une position (le voleur r se trouve sur le sommet x et le policier c sur y , avec $x \sim y$) à partir de laquelle c gagne lors de son prochain déplacement. Forcément $N[v_x] \subseteq N[v_y]$ (G étant policier-gagnant) et on pose $x < y$. On sait que $G \setminus \{x\}$ est policier-gagnant, sinon r a une stratégie gagnante qui l'est aussi dans G , du moment que r pense à c dans x comme s'il se trouvait dans y . Ainsi, en suivant le même raisonnement jusqu'au dernier sommet gagnant (jusqu'à u), on arrive à la fin à trouver un ordre linéaire tel que : $v_1 = x, \dots, v_i = y, \dots$ et $v_n = u$ ($i > 1$) et $S = \{v_1, \dots, v_n\}$ de plus $\forall i, \exists j > i$ tel que $N_i(v_i) \subseteq N_i(v_j)$.

On sait qu'il existe un ordre linéaire tel que $\forall i, \exists j > i$ tel que $N_i(v_i) \subseteq N_i(v_j)$, et on se propose de montrer que G est policier-gagnant. $\forall i, \exists j > i$ tel que v_i est dominé par v_j , cela implique qu'il existe une rétraction f_i définie par : $\forall k > i,$

$f_i(v_k) = v_k$ et $f_i(v_i) = v_j$, qui nous permettra de rétracter les sommets dominés. Ainsi, et en appliquant ce même raisonnement à tous les sommets dominés, on obtient à la fin un seul sommet. D'où G est démontable, et par conséquent G est policier-gagnant. \square

3.1. UN POLICIER, UN VOLEUR

Considérons le graphe orienté G , type de graphes que nous allons plus aborder à partir de cette section, et que nous appellerons dorénavant *digraphe*, et dans lequel un policier et un voleur jouent. Soient les notations suivantes : $c_{x,y}$ et $r_{x,y}$ pour décrire les situations du jeu, et appelées *configurations*. Ainsi, $c_{x,y}$ veut dire que c'est au tour du policier de se déplacer, et qu'il se trouve sur le sommet x et le voleur sur y , et par analogie, $r_{x,y}$ veut dire que le policier se trouve sur x et le voleur sur y , et que c'est au tour du voleur de se déplacer.

À chaque graphe G on associe son digraphe de déplacement $M = M_G$ sur l'ensemble de sommets $V_M = C_M \cup R_M$, où $C_M = \{c_{x,y} : x, y \in V(G)\}$ et $R_M = \{r_{x,y} : x, y \in V(G)\}$. Il y a un arc $(c_{x,y}, r_{w,y})$ si $x \neq y$ et $(x, w) \in E(G)$, et $(r_{x,y}, c_{x,z}) \in E(M)$ si $x \neq y$ et $(y, z) \in E(G)$. Les arcs de M dont l'extrémité initiale est le sommet $c_{x,y}$ représentent les déplacements possibles du policier, et ceux dont $r_{x,y}$ est l'extrémité initiale, ceux du voleur.

Lemme 3.1.1. *Il existe un chemin du sommet $p_{x,y}$ vers $p'_{x',y'}$ dans $M(G)$, où $p \in \{c, r\}$, si et seulement si il existe une suite de déplacements débutant avec le déplacement de p , qui déplace c de x vers x' et r de y vers y' .*

Corollaire 3.1.1. *Si le digraphe G est policier-gagnant, alors il existe un sommet $c_{x,y} \in V_M$ et un chemin de $c_{x,y}$ vers $p_{z,z}$ pour un certain $z \in V(G)$ et $p \in \{c, r\}$.*

En effet, si le voleur joue d'une façon optimale, alors $p = r$.

3.1.1. Algorithme de Hahn et MacGillivray

L'algorithme suivant (algorithme 1), qui a été introduit par Hahn et MacGillivray [15], ressemble à la définition récursive de l'ordre \leq introduite par Nowakowski et Winkler [23].

Chaque sommet $v \in M_G$ est étiqueté par un entier positif $l(v)$ qui indique le nombre de tours de déplacements d'un policier pour gagner. Par exemple, si c'est le tour du voleur de se déplacer, et l'étiquette du sommet est k , alors le policier aura besoin de k -déplacements après que le voleur se soit déplacé. Ainsi, une étiquette d'un sommet indique à chacun des joueurs son déplacement optimal : pour le policier, le plus court chemin pour capturer le voleur, et pour le voleur, le moyen d'éviter d'être capturé le plus longtemps possible, sinon jamais.

Algorithm 1 Algorithme de Hahn et MacGillivray

```

set  $l(v) = \infty$  for all  $v \in V_M$ 
set  $l(c_{x,x}) = l(r_{x,x}) = 0$  for each  $x \in V(G)$ 
repeat
  for each pair  $(x, y)$ ,  $x \neq y$  do
    if  $l(c_{x,y}) = \infty$  then
      set  $l(c_{x,y}) := 1 + \min \{l(r_{x',y}) : x' \in N^+(x)\}$ 
    end if
    if  $l(r_{x,y}) = \infty$  then
      set  $l(r_{x,y}) := \max \{l(c_{x,y'}) : y' \in N^+(y)\}$ 
    end if
  end for
until the no change results

```

Ainsi pour chaque paire (x, y) , deux sommets seulement sont considérés à chaque itération, et puisque l'étiquette d'un sommet n'est jamais changée une fois elle n'est pas ∞ , l'algorithme finit au plus en $O(|V(G)|^2)$ itérations, dont chacune prend au plus $O(|V(G)|^3)$ étapes.

Hahn et MacGillivray [14] ont démontré que l'algorithme fonctionne bien dans le lemme 3.1.2.

Mais tout d'abord, introduisons certaines notions. On appelle ainsi une stratégie

à partir d'une configuration $c_{x,y}$ *optimale* pour le policier (respectivement $r_{x,y}$ pour le voleur) si aucune autre stratégie ne permet au policier de gagner avec moins de déplacements (respectivement si aucune autre stratégie ne prolonge le jeu).

Lemme 3.1.2. *Soit G un graphe orienté et M_G son digraphe de déplacement.*

Alors :

1. *le policier a une stratégie gagnante optimale en t tours à partir de la configuration $c_{x,y}$ si et seulement si $l(c_{x,y}) = t$ quand l'algorithme finit.*
2. *le voleur a une stratégie optimale en t tours ($(t - 1)$ déplacements du voleur) à partir de la configuration $r_{x,y}$ si et seulement si $l(r_{x,y}) = t$ quand l'algorithme finit.*

DÉMONSTRATION. Le cas $t = \infty$ est clair pour les deux configurations $c_{x,y}$ et $r_{x,y}$. Pour un t fini, Hahn et MacGillivray [14] ont procédé par induction sur t . On sait que $l(c_{x,y}) = 0$ si et seulement si $x = y$ et dans ce cas, il n'y a rien à prouver. Si $l(r_{x,y}) = 0$, c'est parce que $l(c_{x,y}) = 0$ pour tout $y' \in N^+(y)$, et de ce fait, il n'y a rien à prouver ici aussi.

1. $l(c_{x,y}) = t$ si et seulement si $l(c_{x,y}) := 1 + \min \{l(r_{x',y}) : x' \in N^+(x)\}$. Ainsi, le policier se déplacera vers un sommet x' à partir d'un chemin qui prend le minimum de déplacements, et à partir duquel, par induction, le policier peut gagner en $t - 1$ tours. Inversement, si le policier peut gagner en t tours à partir de la configuration $c_{x,y}$, il existe un sommet $x' \in N^+(x)$ tel que le policier gagne en $t - 1$ tours à partir de $c_{x',y'}$ pour n'importe quel $y' \in N^+(y)$. Cela veut dire que $l(c_{x',y'}) \leq t - 1$ pour tout $y' \in N^+(y)$ c'est à dire, $l(r_{x,y}) = t - 1$, et par induction $l(c_{x,y}) = t$.
2. Si $l(r_{x,y}) = t$ alors le voleur peut se déplacer vers un sommet $y' \in V(G)$ tel que $l(c_{x,y'}) = t$ mais pas vers z tel que $l(c_{x,z}) > t$. Le policier (voleur) a une stratégie optimale en t tours, c'est à dire, que le voleur échappe à sa capture durant t déplacements. Inversement, supposons que

le voleur a une stratégie optimale de t déplacements à partir de $r_{x,y}$. Alors le policier ne peut pas gagner en moins de t tours, c'est à dire, $\max \{l(r_{x,y'}) : y' \in N^+(y)\} = t$, et ainsi $l(r_{x,y}) = t$.

□

Théorème 3.1.1. *Soit G un graphe orienté et soit $M(G)$ son digraphe de déplacement. Alors G est policier-gagnant si et seulement si il existe un sommet $x \in V(G)$ tel que pour tout $y \in V(G)$, $l(c_{x,y}) < \infty$ dans $M(G)$.*

DÉMONSTRATION. La démonstration du théorème découle du lemme 3.1.2. □

Les corollaires suivants sont une conséquence de la forte connectivité des digraphes.

Corollaire 3.1.2. *Un graphe orienté fortement connexe fini est policier-gagnant si et seulement si, dans $M(G)$, $l(v) < \infty$ pour tout $v \in V_M$.*

Corollaire 3.1.3. *Un graphe fini non-orienté connexe est policier-gagnant si et seulement si, dans $M(G)$, $l(v) < \infty$ pour tout $v \in V_M$.*

L'algorithme donne un moyen facile à chaque joueur d'optimiser son déplacement :

Si le policier se trouve dans x et le voleur dans y , alors le policier doit se déplacer vers x' tel que $l(r_{x',y}) = l(c_{x,y}) - 1$, tandis que le voleur devra se déplacer vers y' tel que $l(c_{x,y}) = l(r_{x,y})$. La longueur maximum du jeu est aussi donnée par l'algorithme, en autant que les deux joueurs jouent leurs stratégies optimales, et c'est l'étiquette maximum non-nulle d'un $c_{x,y}$, $x, y \in V(G)$.

Ainsi, on a un graphe G , auquel on va associer deux autres graphes de déplacement des joueurs G_C et G_R , G_C étant le graphe de déplacement du policier, et G_R celui du voleur.

À partir de ces deux graphes, on va associer deux composantes sous la forme de matrices à 2-dimensions de sommets (voir tableaux 3.1 et 3.2). Ces deux composantes représentent l'étiquetage des sommets pour les graphes de déplacement des deux joueurs et permettent de déterminer si le graphe est policier-gagnant ou

non, ainsi que la stratégie optimale.

L'algorithme commence par étiqueter tous les sommets par ∞ à part les sommets $c_{x,x}$ et $r_{y,y}$, qui sont étiquetés par 0. Ensuite on répète les deux opérations suivantes jusqu'à ce qu'aucune ne soit plus applicable :

1. S'il existe un sommet dans la composante policier dont l'étiquette est égale à ∞ , on regarde ses sommets adjacents dans la composante voleur, et on le réétiquete de la plus petite étiquette finie de ses sommets adjacents plus 1.
2. S'il existe un sommet dans la composante voleur dont l'étiquette est égale à ∞ , on regarde ses sommets adjacents dans la composante policier, et on le réétiquete de la plus grande étiquette finie de ses sommets adjacents.

Pour déterminer si un graphe non-orienté est policier-gagnant ou non, on vérifie que toutes les étiquettes de la composante policier sont inférieures à ∞ . Pour ce qui d'un graphe orienté, on vérifie que les étiquettes d'une ligne de la composante policier soient toutes inférieures à ∞ (voir exemples 3.1.1 et 3.1.2).

Le sommet du graphe qui permet une stratégie optimale est celui qui correspond à la ligne de la composante policier dont la plus grande étiquette reste la plus petite parmi les étiquettes des autres lignes.

Exemple 3.1.1 (Exemple d'étiquetage d'un graphe policier-gagnant non-orienté).

Ceci est un exemple d'étiquetage d'un graphe policier-gagnant non-orienté simple (figure 3.1). On remarque que toutes les étiquettes de la composante policier (tableau 3.1) sont inférieures à ∞ .

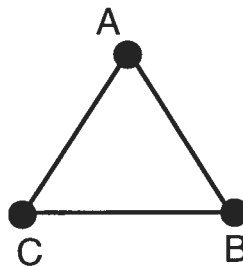


FIGURE 3.1. Graphe policier-gagnant

sommet 1	sommet 2	sommet 3	sommet 10	sommet 11	sommet 12
c(0,0)	c(0,1)	c(0,2)	r(0,0)	r(0,1)	r(0,2)
0	1	1	0	1	1
sommet 4	sommet 5	sommet 6	sommet 13	sommet 14	sommet 15
c(1,0)	c(1,1)	c(1,2)	r(1,0)	r(1,1)	r(1,2)
1	0	1	1	0	1
sommet 7	sommet 8	sommet 9	sommet 16	sommet 17	sommet 18
c(2,0)	c(2,1)	c(2,2)	r(2,0)	r(2,1)	r(2,2)
1	1	0	1	1	0
Composante policier			Composante voleur		

TABLEAU 3.1. Les sommets du graphe de déplacement d'un graphe policier-gagnant

Exemple 3.1.2 (Exemple d'étiquetage d'un graphe voleur-gagnant orienté). *Ceci est un exemple d'étiquetage d'un graphe orienté voleur-gagnant (figure 3.2). On remarque ici que sur toutes les lignes de la composante policier (tableau 3.2), il existe au moins un sommet dont l'étiquette est égale à ∞*

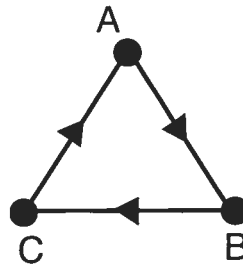


FIGURE 3.2. Graphe voleur-gagnant

sommet 1	sommet 2	sommet 3	sommet 10	sommet 11	sommet 12
c(0,0)	c(0,1)	c(0,2)	r(0,0)	r(0,1)	r(0,2)
0	1	∞	0	∞	0
sommet 4	sommet 5	sommet 6	sommet 13	sommet 14	sommet 15
c(1,0)	c(1,1)	c(1,2)	r(1,0)	r(1,1)	r(1,2)
∞	0	1	0	0	∞
sommet 7	sommet 8	sommet 9	sommet 16	sommet 17	sommet 18
c(2,0)	c(2,1)	c(2,2)	r(2,0)	r(2,1)	r(2,2)
1	∞	0	∞	0	0
Composante policier			Composante voleur		

TABLEAU 3.2. Les sommets du graphe de déplacement d'un graphe voleur-gagnant

3.2. GRAPHERS GÉNÉRAUX ET LE THÉORÈME DE NOWAKOWSKI

En étudiant le théorème de Nowakowski, Winkler, et Quillot (théorème 3.0.5), on pourrait se poser certaines questions, entre autres, est-ce qu'on peut le réadapter à des graphes non réflexifs ou partiellement réflexifs ? Répondre à cette question revient en fait à répondre à une autre question, à savoir que rapporte le fait d'avoir une boucle sur un sommet à l'ordre linéaire du théorème 3.0.5, ou bien au fait que le graphe soit policier-gagnant ?

On remarque ainsi, que si on veut avoir un ordre linéaire avec la dominance précitée, il faut vérifier que les deux règles suivantes soient toujours vraies :

- si le dominant u est le voisin de son dominé v , alors u doit avoir une boucle, sinon, on ne peut pas parler de dominance puisque u est un voisin de v mais pas de lui-même,
- si le dominant et le dominé ne sont pas voisins, alors tous les voisins du dominés sont voisins du dominant, et le dominant peut être réflexif, mais pas le dominé, sinon le dominé est le voisin de lui-même, mais pas de son dominant,

À partir de la première règle, il est évident que pour avoir un ordre linéaire, il faut avoir au moins un sommet avec boucle, et il doit être le dernier par cet ordre, étant donné qu'il doit dominer l'avant-dernier sommet (voir figure 3.3 de l'exemple 3.2.1).

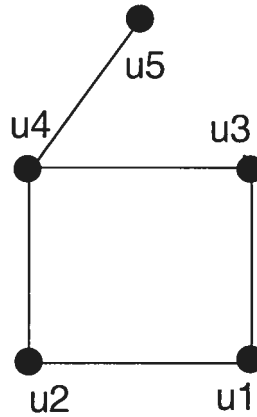


FIGURE 3.3. Ordre linéaire et dominance

Exemple 3.2.1. Soit le graphe G de la figure 3.3, on remarque que les sommets de G ne sont pas tous ordonnés linéairement. Ainsi, u_1 est dominé par u_4 dans le sous-graphe G_1 , u_2 (respectivement u_3) est dominé par u_5 dans le sous-graphe G_2 (respectivement G_3), mais u_4 n'est pas dominé par u_5 (u_5 n'est pas réflexif).

On pourrait aussi vérifier si le théorème 3.0.5 est toujours valide pour n'importe quel type de graphe (non réflexif ou partiellement réflexif). En fait, pour un graphe non réflexif, on ne peut pas parler d'ordre linéaire, puisqu'on sait qu'il faut avoir au moins un sommet réflexif.

Il reste quand même le cas d'un graphe partiellement réflexif (par graphe partiellement réflexif, on entend un graphe avec $(n - k)$ sommets non réflexifs, où n est le nombre de sommets et $0 < k < n$).

Il est évident que si on peut ordonner linéairement les sommets d'un graphe G v_1, v_2, \dots, v_n de telle façon que $\forall i \in \mathbb{N}, \exists j > i$ tel que $N_i(v_i) \subseteq N_i(v_j)$ (v_n est réflexif), alors G est policier-gagnant; démontrer cela, revient à rétracter le sommet réflexif, et tout sommet réflexif obtenu à partir de cette opération de rétraction, et cela jusqu'à l'obtention d'un seul sommet (définition d'un graphe démontable).

Pour ce qui est de l'autre sens, à savoir, si on a un graphe, avec au moins un sommet réflexif, policier-gagnant, cela veut-il dire qu'il existe un ordre linéaire policier-gagnant qui vérifie la dominance de sommets précités plus haut? à première vue, il apparaît qu'il est valide, et sa démonstration reprendra celle du théorème 3.0.5 (voir la démonstration 3) avec quelques modifications dues aux

sommets réflexifs. (voir les figures suivantes 3.4 et 3.5). Ainsi la stratégie gagnante, contrairement au cas des graphes réflexifs qui consiste à capturer le voleur au dernier coup, serait de forcer le voleur à se déplacer vers un sommet, à partir duquel, quelque soit son déplacement final, il se place dans une position qui mène à sa capture.

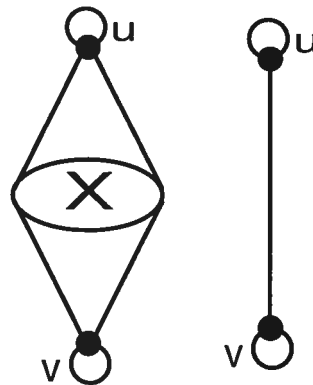


FIGURE 3.4. Cas de dominance de v par u dans un graphe réflexif

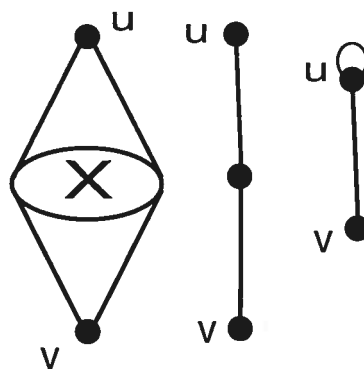


FIGURE 3.5. Cas de dominance de v par u dans un graphe partiellement réflexif

Ainsi, on peut formuler le théorème suivant.

Théorème 3.2.1. (*Théorème de Nowakowski, Winkler, et Quillot modifié*) Soit un graphe G partiellement réflexif, G est policier-gagnant si et seulement si ses

sommets peuvent être linéairement ordonnés v_1, v_2, \dots, v_n de telle façon que $\forall i \in \mathbb{N}, \exists j > i$ tel que $N_i(v_i) \subseteq N_i(v_j)$.

En revenant à l'exemple 3.2.1, on remarque que si la boucle avait été sur le sommet u_4 (figure 3.6) ou bien sur le sommet u_5 (deux sommets qui peuvent être les derniers dans l'ordre linéaire), on aurait eu un graphe policier-gagnant, ce qui reprend en fait ce qu'on a dit plus haut.

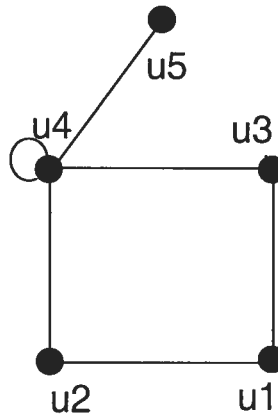


FIGURE 3.6. Graphe avec un ordre linéaire policier-gagnant

Une autre question qu'on pourrait se poser, est si on a un graphe réflexif policier-gagnant, est-ce que ce même graphe sans boucles ou sur certains de ses sommets est toujours policier-gagnant et vice-versa ?

En fait pour le cas où on a un graphe non réflexif ou partiellement réflexif policier-gagnant, le même graphe, mais réflexif n'est pas policier-gagnant (voir figure 3.7), contrairement à l'autre sens, d'où le théorème 3.2.2. Mais avant, introduisons le lemme suivant :

Lemme 3.2.1. *Soit G un graphe réflexif policier-gagnant. Un graphe modifié G' partiellement réflexif peut-être policier-gagnant.*

DÉMONSTRATION. Soit G un graphe réflexif policier-gagnant, alors d'après le théorème 3.0.5, il existe un ordre linéaire qui vérifie la dominance. Si on enlève les boucles sur certains sommets du graphe G , à part sur le dernier sommet dans l'ordre linéaire qui doit rester réflexif, sinon la dominance n'est plus vérifiée (voir

théorème 3.2.1), alors, par ce même théorème, le graphe modifié G' est aussi policier-gagnant. \square

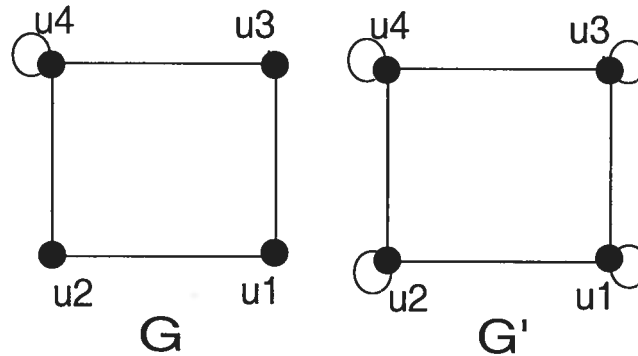


FIGURE 3.7. Graphe G policier-gagnant, G' voleur-gagnant

Théorème 3.2.2. *Soit G un graphe policier-gagnant réflexif, alors le graphe partiellement réflexif G^{**} est aussi policier-gagnant.*

DÉMONSTRATION. Soit le graphe G réflexif policier-gagnant, d'après le lemme 3.2.1, le graphe partiellement réflexif G^{**} , obtenu à partir de la suppression de boucles sur certains sommets (le dernier sommet, par ce même ordre linéaire qui reste valide, doit rester réflexif) est aussi policier-gagnant. \square

Si on a un graphe réflexif ou partiellement réflexif policier-gagnant, et si on ouvre les sommets avec boucles (d'un sommet réflexif on obtient deux sommets avec les mêmes voisins que le sommet d'origine (voir figure 3.8)), est-ce que le graphe non réflexif modifié resterait policier-gagnant? La réponse est oui, donnée par le théorème suivant :

Théorème 3.2.3. *Soit G^o un graphe réflexif ou partiellement réflexif policier-gagnant. Le graphe G^+ non réflexif représentant le graphe G^o , mais dont les boucles ont été ouvertes est aussi policier-gagnant.*

DÉMONSTRATION. Soit le graphe G^o réflexif ou partiellement réflexif policier-gagnant, cela implique qu'il existe une stratégie policier-gagnante. Cette stratégie est alors appliquée au graphe G^+ non réflexif obtenue. En effet, le policier joue

dans G^+ comme s'il aurait joué dans G^o (de même que le voleur), sur tout sommet non réflexif. Si par contre le policier doit rester sur place à un moment donnée (un sommet réflexif u , comme dans la figure 3.8), à ce moment-là, il peut se déplacer entre les deux sommets jumeaux obtenus à partir de l'opération de l'ouverture de la boucle autant de fois que cela est nécessaire. Ainsi, le graphe G^+ est policier-gagnant aussi. \square

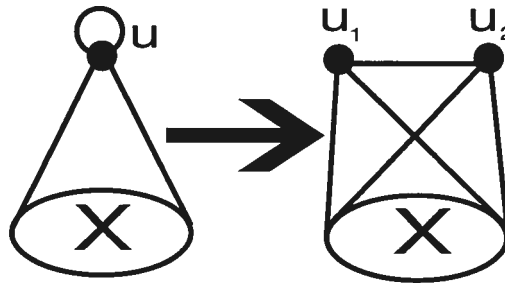


FIGURE 3.8. Opération d'ouverture d'un sommet réflexif

La question qu'on pourrait se poser est : l'inverse est-il vrai ? mais avant de répondre à cette question, introduisons un lemme et une nouvelle notion : Soit un graphe G , s'il existe un sous-graphe complet dont tous les sommets ont les mêmes voisins, alors on peut *identifier* tous ces sommets pour en former un seul sommet réflexif (voir figure 3.9) en sauvegardant leur voisinage.

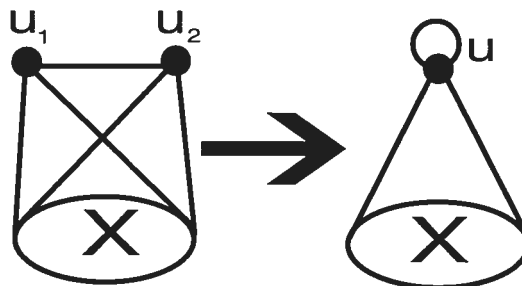


FIGURE 3.9. Opération d'identification en un sommet réflexif

Lemme 3.2.2. *Soit un graphe G policier-gagnant. Si G possède un sous-graphe complet qu'on peut identifier pour former un seul sommet réflexif, alors le graphe résultant G^- partiellement réflexif est policier-gagnant.*

DÉMONSTRATION. En fait, il suffit de jouer la stratégie gagnante optimale du graphe G dans le graphe résultant G^- de l'opération d'identification. Ainsi, le policier se déplace dans le graphe G^- comme il se déplacerait dans G , et le voleur aussi. Quand le policier se déplace dans le sous-graphe complet dans G , cela revient à rester sur le sommet identifié réflexif dans le graphe G^- (un sommet u , comme dans la figure 3.9); comme le policier doit sortir du sous-graphe du G à un moment donné, il lui suffira de quitter le sommet réflexif à ce moment-là dans le graphe G^- . Ainsi, le policier capturera le voleur dans G^- , comme il l'aurait fait dans G , et donc le graphe G^- est policier-gagnant \square

Le Théorème suivant est une généralisation du lemme 3.2.2.

Théorème 3.2.4. *Soit G un graphe possédant des sous-graphes qui peuvent être identifiés en sommets réflexifs. Si G est policier-gagnant, alors le graphe obtenu G^- partiellement réflexif est policier-gagnant.*

DÉMONSTRATION. On suppose que le graphe G d'origine est policier-gagnant, et on suppose aussi que G possède un ou plusieurs sous-graphes complets qui vérifient le lemme 3.2.2. Ainsi la stratégie gagnante pour G peut être adaptée pour le graphe obtenue G^- de la manière suivante : le policier reste sur place sur le sommet crée qui correspond au sous-graphe complet si telle est sa stratégie dans G , sinon, il se déplace normalement comme dans G . Cette étape est répétée autant de fois qu'il y'a de de sous-graphes identifiés, et cela jusqu'à la capture du voleur. D'où le graphe G^- est policier-gagnant. \square

3.3. K-POLICIERS, L-VOLEURS

L'algorithme de Hahn et MacGillivray (algorithme 1) peut être utilisé pour dire si un graphe est (k, l) -policier-gagnant pour tout $k \geq l > 0$ où k est le

nombre de policiers et l celui de voleurs. Il faut aussi spécifier que les policiers gagnent lorsque chaque voleur partage un sommet avec au moins un policier. Pour que k -policiers (par analogie, pour que l -voleurs) soient sur au plus k sommets, cela veut dire qu'ils sont sur un point de V^k . Pour un point $u \in V^k$, on écrit $\bar{u} = (u_1, u_2, \dots, u_k)$ (par analogie $\bar{v} = (v_1, v_2, \dots, v_l)$). On pose $0 < l \leq k$ et soit le digraphe $G = (V, A)$ avec un degré sortant minimum 1. On définit $C_k(G) = (V^k, A_k)$ en posant $A_k = \{(u, v) : u, v \in V^k, v_i \in N_G^+(u_i), i = 1, \dots, k\}$. De même, on définit $R_l(G) = (V^l, A_l)$. Ainsi un déplacement de policiers ou de voleurs dans G correspond à des déplacements par un policier dans $C_k(G)$ ou bien par un voleur dans $R_l(G)$. Le (k, l, G) graphe de jeu \mathcal{G} est la réunion disjointe des copies de $C_k(G)$ et $R_l(G)$. Formellement $\mathcal{G} = (V^k \times \{c\} \cup V^l \times \{r\}, A(c) \cup A(r))$ où $A(c) = \{(u, v) : u, v \in V^k \times \{c\}, (u, v) \in A_k\}$, et $A(r) = \{(u, v) : u, v \in V^l \times \{r\}, (u, v) \in A_l\}$. Le jeu peut maintenant être joué dans \mathcal{G} : les policiers se déplacent d'un sommet vers un autre de $V^k \times \{c\}$ tandis que les voleurs se déplacent dans $V^l \times \{r\}$ (comme pour le jeu 1-policier, 1-voleur, ce sont les policiers qui choisissent les premiers leurs positions, suivis par les voleurs, et durant le jeu, ils alternent leurs déplacements). Les policiers gagnent si à un certain moment du jeu, ils se trouvent dans un certain (u, c) et les voleurs dans (v, r) tel que $\{v_1, v_2, \dots, v_l\} \subseteq \{u_1, u_2, \dots, u_k\}$. On appelle ce jeu dans \mathcal{G} le *jeu ombre* du (k, l) jeu dans G . Le lemme suivant est évident.

Lemme 3.3.1. *Un digraphe G est (k, l) -policier-gagnant si et seulement si k policiers peuvent gagner le jeu ombre avec l voleurs dans \mathcal{G} .*

Remarque 3.3.1. *Si on insiste pour que le jeu soit joué dans un graphe connexe, on peut augmenter le modèle cité plus haut ; le (k, l, G) digraphe de jeu augmenté \mathcal{G}^- est construit à partir de \mathcal{G} en ajoutant deux nouveaux sommets contrôles α, β avec quelques arcs, ensuite, en ajoutant les arcs qui indiquent comment les policiers contrôlent l'espace autour d'eux dans G . Ainsi $V(\mathcal{G}^-) = V(\mathcal{G}) \cup \{\alpha, \beta\}$ ($\alpha, \beta \notin V(\mathcal{G})$) et $A(\mathcal{G}^-) = A(\mathcal{G}) \cup \{(\alpha, x) : x \in V^k \times \{c\}\} \cup \{(\beta, y) : y \in V^l \times \{r\}\} \cup$*

$\{(u, v) : u = ((x_1, \dots, x_k), c), v = (y, r), y \in \prod_{i=1}^k N^+(x_i)\}$. On remarque que le digraphe de jeu adapté n'est jamais réflexif (α et β n'ont pas de boucles) et que \mathcal{G} est réflexif si et seulement si G l'est aussi.

Théorème 3.3.1. *Un digraphe G est (k, l) -policier-gagnant si et seulement si le digraphe de jeu adapté \mathcal{G}^- est policier-gagnant.*

Avant de démontrer ce théorème, introduisons une notation dont on aura besoin, à savoir L_G un digraphe qui représente les positions des policiers dans un graphe jeu.

DÉMONSTRATION. Supposons que les policiers ont une stratégie gagnante dans G . On décrit une stratégie gagnante pour le policier dans un graphe policier-gagnant dans \mathcal{G}^- . Le policier commence par choisir α . Ceci force le voleur à choisir β ou à commencer dans G , sinon, il est immédiatement capturé. Le policier se déplace alors vers un sommet de L_G correspondant au déplacement initial des policiers dans G . Le déplacement du voleur le met dans G quelque soit son point de départ, étant donné que β n'a pas de boucle, et qu'un déplacement de β le mène dans G . Le policier se déplace dans L_G en suivant la stratégie gagnante dans G . Dû à la construction de \mathcal{G}^- , le voleur est éventuellement capturé. D'où \mathcal{G}^- est policier-gagnant.

Supposons que les voleurs ont une stratégie gagnante dans G . Dans \mathcal{G}^- , le policier doit choisir un sommet dans L_G ou bien α , sinon le voleur joue dans L_G , et ainsi il ne sera jamais capturé. Le voleur lui, choisit β si le policier se met dans α , ou bien, sa position initiale dans G , si le policier se met dans L_G . Dans le premier cas, le prochain déplacement du policier le mène vers un sommet dans L_G , tandis que le voleur se déplace vers sa position initiale dans G . Le voleur joue alors sa stratégie gagnante dans G . \square

3.4. QUELQUES CONSIDÉRATIONS ET RÉFLEXIONS

Dans cette section, nous allons voir quelques considérations concernant des graphes avec k policiers et l voleurs. De plus, nous allons étudier les possibles

versions de capture des voleurs et leurs influences sur l'issue du jeu.

Ainsi, soit le jeu joué par k policiers et l voleurs dans un graphe G . Il est facile de voir que G est k -policier-gagnant pour $l > 1$ si et seulement si il l'est pour $l = 1$ dans le cas où les policiers peuvent capturer les voleurs 1 par 1, en mettant les voleurs capturés hors du jeu. Appelons cette version de jeu A . On remarquera qu'un graphe k -policier-gagnant pour $l > 1$ n'est pas forcément k -policier-gagnant pour $k - 1$ (voir figure 3.10 d'un graphe qui est $(3, 1)$ -policier-gagnant, mais qui n'est pas $(2, 1)$ -policier-gagnant).

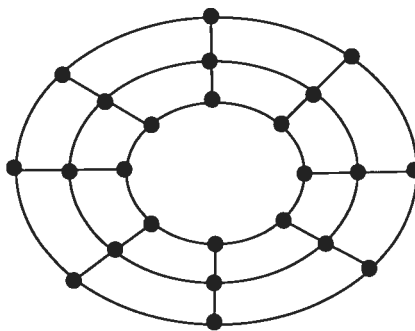


FIGURE 3.10. Graphe $(3, 1)$ -policier-gagnant, mais pas $(2, 1)$ -policier-gagnant

Est-ce que c'est vrai aussi, si la règle de capture est *tous les voleurs sont capturés en même temps*? (version B). Il est évident que si $k < l$ et si un policier ne peut capturer qu'un voleur, les policiers ne peuvent pas gagner (même s'ils peuvent gagner dans la version A), il est tout aussi évident que même si $k > l$, dans la version B , les policiers ne peuvent pas gagner, puisqu'il suffit qu'un voleur *se suicide* (se laisse capturer dès le début du jeu, en se mettant sur le même sommet qu'un policier pour arrêter le jeu). Il reste quand même les cas où un policier peut capturer plusieurs voleurs, et où k est supérieur à l ; à ce moment là, on revient à la version A du jeu, mais en plus, on doit ajouter une autre contrainte, à savoir, combien de voleurs un policier peut capturer en même temps (par exemple, un policier peut capturer 2 voleurs au maximum), en particulier si $k < l$; il est évident de remarquer que si un policier peut capturer n voleurs, alors le graphe n'est pas

(k, l) -policier-gagnant si $l > kn$.

Bien sûr, il y'a d'autres contraintes qui influencent sur le résultat du jeu, entre autres, la contrainte sur le déplacement des policiers (ils doivent se déplacer par exemple en tandem ou paire de telle façon qu'il y'ait toujours au plus une longueur entre eux comme cela a été décrit par Clarke et Nowakowski [12]), ou bien la contrainte sur l'information que les antagonistes ont du jeu ; dans ce mémoire on a supposé qu'elle était complète, mais elle peut être partielle ou limitée (les joueurs ont une connaissance limitée de leurs voisinages par exemple), et à ce moment-là, les joueurs se déplacent en ayant une connaissance sur leurs voisinages immédiats d'une longueur ou deux par exemple, ce qui fait qu'ils se déplacent en aveugle : ne connaissant pas où se trouvent leurs adversaires, ni vers quels sommets ils vont se déplacer (ce qui correspond en fait à la réalité). On peut aussi bien combiner toutes ces contraintes : information incomplète et contraintes sur le déplacement.

Chapitre 4

L'ALGORITHME DE HAHN ET MACGILLIVRAY

Dans ce chapitre, nous allons voir l'implémentation de l'algorithme de Hahn et MacGillivray (voir algorithme 1) qui a été faite dans le cadre de la partie pratique de ce mémoire.

Mais avant d'aborder la partie implémentation, nous allons voir certaines notions telles que l'implémentation d'un graphe et les systèmes de triplets de Steiner, graphes dont nous voulons déterminer les copnumbers, étant donné qu'on veut regarder les tournois (graphe orienté obtenu à partir d'un graphe non-orienté complet en donnant à chaque arête une direction) et trouver leurs copnumbers. Étant donné que les systèmes de triplets de Steiner correspondent aux décompositions des arêtes d'un graphe complet en triangles (i.e. triangulation), ils sont, par conséquent, les mieux adaptés à ce genre de graphes, et on peut alors donner une direction à chaque arête en orientant les triplets de base cycliquement et en les faisant tourner.

4.1. LES SYSTÈMES DE TRIPLETS DE STEINER

Ainsi, le *système de triplets de Steiner* communément appelé **STS** consiste en un ensemble A de n éléments, et d'une collection de sous-ensembles de A , appelés blocs ou triplets, tel que chaque bloc contient exactement 3 éléments, et que deux éléments quelconques se trouvent ensemble dans exactement un bloc.

Le nombre n est appelé **ordre** de STS.

Le théorème suivant a été introduit par Kirkman [17].

Théorème 4.1.1. (Kirkman) *Un STS non trivial de l'ordre n , qu'on notera $STS(n)$, existe si et seulement si n est congru à 1 ou à 3(mod 6).*

La démonstration de ce théorème donnée par Kirkman a été faite par étapes.

Il a ainsi introduit les 4 faits suivants :

- (a) $STS(1)$ existe,
- (b) si $STS(n)$, alors $STS(2n + 1)$ existe,
- (c) si $STS(n)$ existe et $n > 1$, alors $STS(n - 2)$ existe,
- (d) si $STS(n)$ existe, alors $STS(2n - 1)$ existe.

Ainsi, de (a)-(d), il a montré que $STS(n)$ existe pour tout entier positif $n \equiv 1, 3 \pmod{6}$. On considère un STS comme trivial si son ordre est égal à 0, 1 ou 3. Le nombre n est appelé **admissible** s'il satisfait aux conditions du théorème 4.1.1 de Kirkman. Ainsi, le plus petit ordre non trivial est 7, et le STS obtenu est unique, et il peut être représenté de plusieurs façons :

- Une représentation cyclique : l'ensemble de points est $\{0, 1, \dots, 6\} = \mathbb{Z}_7$ et les triplets sont l'ensemble $\{1 + i, 2 + i, 4 + i\}, i = 0, \dots, 6$.
- Une représentation binaire : les points sont tous les triples d'éléments binaires excepté 000, et les blocs sont tous les ensembles de trois tels triplets dont la somme par position "par points" est zéro (par exemple $\{110, 101, 011\}$).
- Interprétant ce qui précède comme nombres entiers dans la base 2, nous obtenons la représentation de Nim : l'ensemble de point est $\{1, 2, \dots, 7\}$, et les blocs sont $\{1, 2, 3\}, \{1, 4, 5\}, \{1, 6, 7\}, \{2, 4, 6\}, \{2, 5, 7\}, \{3, 4, 7\}, \{3, 5, 6\}$. (ce sont toutes les positions gagnante dans le jeu de Nim [22] se composant de trois piles avec au plus sept compteurs dans chacun.).

Le $STS(7)$ est unique, i.e., les systèmes sont isomorphes. Figure 4.1 montre ce système sous la forme du plan projectif d'ordre 2 dit *de Fano* [16].

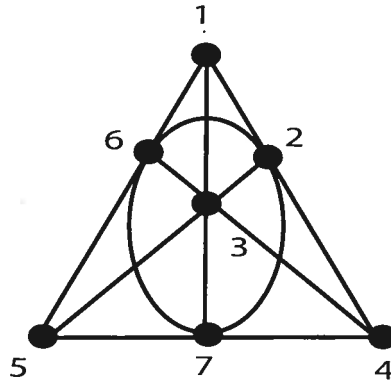


FIGURE 4.1. Plan de Fano

On remarque que le plan de Fano a 7 points et 7 lignes ; chaque ligne est formée de 3 points ou sommets, chaque point se trouve sur 3 lignes et chaque paire de points se trouve exactement sur une seule ligne, les lignes sont : 124, 235, 346, 457, 156, 267, et 137.

4.1.1. Énumération

Pour un nombre admissible petit ($n < 20$), le nombre des $STS(n)$ (par isomorphisme) est donné dans le tableau suivant (voir tableau 4.1). Les données pour $n = 15$ et $n = 19$ sont dus à White, Cole et Cummings, et à Kasri et Ostergard respectivement.

n	7	9	13	15	19
Nombre de STS	1	1	2	80	11084874829

TABLEAU 4.1. Nombre de $STS(n)$

Pour les tests que nous avons faits dans le cadre de ce mémoire, nous avons regardé la représentation de STS sous forme de graphes orientés engendrés par un ou plusieurs triplets suivants le nombre de sommets. Ainsi, le STS consiste de A l'ensemble de sommets du graphes, et de collection de blocs de 3 sommets. De plus, on remarquera qu'aucun arc ne se répète dans le graphe (voir l'exemple 4.1.1 de construction d'un graphe de Steiner d'ordre 7).

Exemple 4.1.1 (Construction d'un graphe de Steiner d'ordre 7). *La figure 4.2 représente les 2 premières étapes de construction d'un graphe de Steiner de 7 sommets engendré par le triplet $(1, 2, 4)$, ainsi que le graphe obtenu à la fin.*

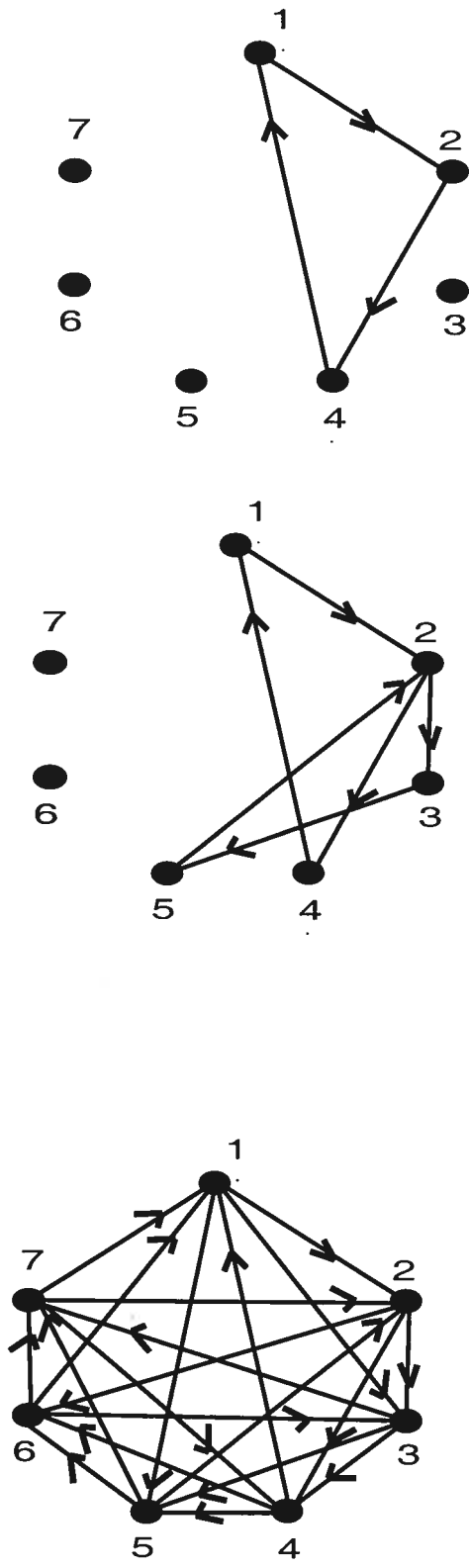


FIGURE 4.2. Construction du graphe de Steiner de 7 sommets

4.2. IMPLÉMENTATION DE L'ALGORITHME

Deux implémentations des graphes sont couramment utilisées ; le choix se fait en fonction des opérations que l'on veut faire, et de la densité du graphe, c'est à dire, du rapport entre le nombre d'arcs (arêtes) et le nombre de sommets.

4.2.1. Implémentation matricielle

Il existe deux types de représentations matricielles d'un graphe : matrice d'adjacence utilisée dans ce mémoire et matrice d'incidence.

La *matrice d'incidence sommets-arcs* d'un graphe $G = (V, A)$ est une matrice $M_G = (a_{i,u})$ (où $(i, u) \in [1, n] \times [1, m]$ et n le nombre de sommets et m celui d'arcs) à coefficients entiers appartenant à l'ensemble $\{0, +1, -1\}$ telle que chaque colonne correspond à un arc de G , et chaque ligne à un sommet de G ; si $u = (i, j) \in A$, la colonne u a tous les termes nuls sauf $a_{i,u} = 1$ et $a_{j,u} = -1$ (voir tableau 4.2). D'une façon évidente, si on considère une ligne i quelconque (correspondant au sommet i) alors :

$$w^+(i) = \{u \mid a_{i,u} = 1\}$$

$$w^-(i) = \{u \mid a_{j,u} = -1\}$$

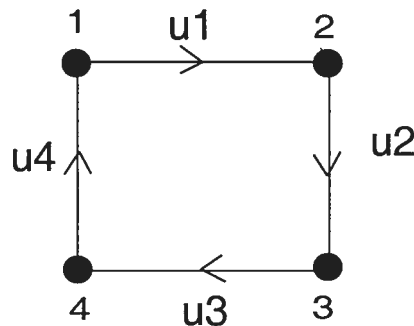


FIGURE 4.3. Matrice d'incidence sommets-arcs

L'arc $u1$ relie le sommet 1 au sommet 2 donc $a_{1,1} = 1$, $a_{2,1} = -1$ et tous les autres $a_{i,1} = 0$. De même pour les autres arcs.

	u1	u2	u3	u4
1	1	0	0	-1
2	-1	1	0	0
3	0	-1	1	0
4	0	0	-1	1

TABLEAU 4.2. Matrice d'incidence sommets-arcs

La *matrice d'incidence sommets-arêtes* d'un graphe $G = (V, E)$ est une matrice à coefficients 0 ou 1, où chaque ligne i correspond à un sommet i de G , et chaque colonne à une arête $u = \{i, j\}$ de G . Si $u = \{i, j\}$ alors la colonne u a tous ses éléments nuls sauf $a_{i,u} = 1$, et $a_{j,u} = 1$ (voir tableau 4.3).

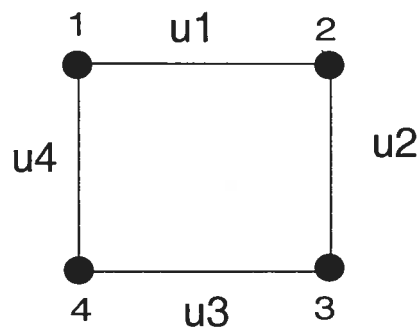


FIGURE 4.4. Matrice d'incidence sommets-arêtes

	u1	u2	u3	u4
1	1	0	0	1
2	1	1	0	0
3	0	1	1	0
4	0	0	1	1

TABLEAU 4.3. Matrice d'incidence sommets-arêtes

La *matrice d'adjacence* M_G d'un graphe G est une matrice carrée, indexée par les sommets (donc de dimension $n \times n$, où n est le nombre de sommets), dont les éléments indiquent l'existence d'un arc(arête). Dans le cas de notre implémentation, nous avons utilisé représentation sous forme d'une matrice booléenne, i.e., dont les coefficients sont égaux à 0 ou 1 ($M_G[u][v] = 1$ si et seulement si u est voisin de v , et $M_G[u][v] = 0$ dans le cas contraire) (voir exemple 4.2.1). La représentation d'un graphe par sa matrice d'adjacence est préférée quand le graphe est dense (beaucoup d'arcs), car elle comporte toujours les n^2 éléments d'un tableau bidimensionnel; elle est bien adaptée aux algorithmes qui s'expriment à l'aide d'opérations matricielles. Par contre, l'inconvénient majeur de ce type de structure de données, c'est qu'elle tient une place mémoire considérable. En fait, pour coder un graphe avec $O(n)$ de données, on utilise une place mémoire en $O(n^2)$ puisqu'on est obligé de coder tous les arcs (arêtes), qu'ils existent ou non.

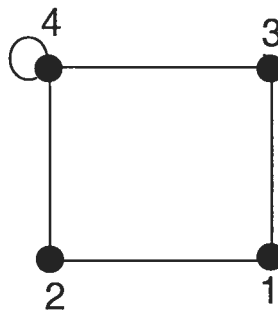


FIGURE 4.5. Matrice d'adjacence

Exemple 4.2.1 (Implémentation matricielle d'un graphe). *L'implémentation matricielle du graphe de la figure 4.5 est :*

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

4.2.2. Implémentation avec listes

Pour palier aux problèmes de mémoire de la représentation matricielle, on se tourne vers des structures dynamiques : les listes chaînées, et une représentation sous forme de *liste d'adjacence*. Un graphe apparaît alors comme une liste d'éléments, chaque élément étant composée d'un sommet et de la liste des arcs (arêtes) qui ont ce sommet comme origine (voir exemple 4.2.2). Dans ce cas, la "gymnastique" à effectuer pour implémenter la structure est plus poussée, mais on ne code plus que pour les arêtes effectives donc le gain d'espace mémoire est considérable. Cette représentation est préférée quand le graphe est creux (peu d'arcs ou d'arêtes) (voir exemple 4.2.2).

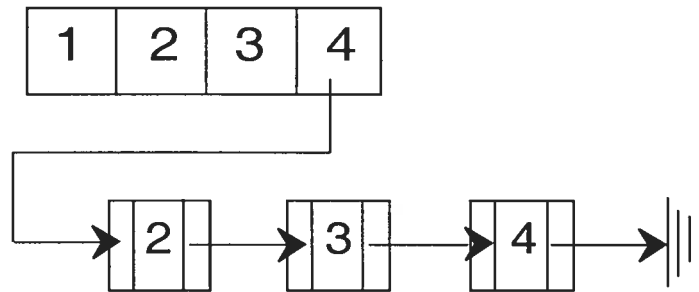


FIGURE 4.6. Liste d'adjacence

Exemple 4.2.2 (Implémentation avec listes). *Pour le graphe de la figure 4.5, la liste d'adjacence pour le sommet 4 est représentée dans la figure 4.6.*

4.2.3. Passage d'une représentation à l'autre

En pratique, on utilise la représentation avec des listes pour stocker le graphe, car c'est elle qui occupe le moins d'espace mémoire, et dès qu'il s'agit de passer aux calculs, on transforme le graphe en matrice pour éviter des parcours de listes redondants et coûteux; d'où la nécessité de définir une fonction de *matricialisation* pour transformer nos listes en matrice, et dans le cas de notre implémentation en vecteurs qui sont plus dynamiques que des tableaux.

4.3. IMPLÉMENTATION

Comme le graphe de départ peut être dense et comme sa taille ne sera pas vraiment très grande ($n \leq 100$), son implémentation a été représentée sous forme de matrice d'adjacence booléenne ou tableau à deux dimensions. Par contre, la représentation des graphes de déplacement des policiers et des voleurs a été réalisée sous la forme de listes chaînées, car la taille de ces graphes augmentait considérablement avec les nombres de policiers et de voleurs. Ainsi, pour un graphe de départ de n sommets, et de k -policiers et l -voleurs, la taille des graphes de déplacement des policiers (ou voleurs) est dans $\Omega(n^{k+l})$ sommets.

De plus, même si ces graphes de déplacement n'étaient pas vraiment denses, le problème résidait dans le fait que durant la phase d'étiquetage des sommets des graphes, on devait les parcourir en entier pour tester si un sommet est voisin d'un autre ou non. Comme le parcours se faisait en $O(n^{(k+l) \times 2})$ pour chaque graphe de déplacement, et comme la phase d'étiquetage se fait tant que les deux conditions de l'algorithme de Hahn et MacGillivray ne sont pas atteintes (voir algorithme 1), cela devenait énorme avec des graphes initiaux de plus de 10 sommets.

L'autre tâche réalisée durant l'implémentation de l'algorithme, a été la construction des graphes de déplacement pour k -policiers et l -voleurs. Cette opération s'est faite en 3 étapes :

- vérifier que le déplacement était possible,
- si oui, ajouter le sommet à la liste chaînée et l'ajuster après,
- enfin, l'étape de la matricialisation ou plus précisément, dans notre cas, de vectorisation (changer la liste en vecteur) au lieu de matrice.

Enfin la dernière étape consistait à tester si le graphe est policier-gagnant ou non, pour cela on devait, premièrement vérifier si le nombre de sommets n du graphe est supérieur ou non au nombre de policier k , car dans le cas contraire, le graphe est (k, l) -policier-gagnant. Ensuite, on vérifiait si le nombre de voleurs l est supérieur à k , car dans ce cas-là le graphe n'est pas (k, l) -policier-gagnant (sauf si on prend pour acquis que lorsqu'un voleur est capturé, il est mis hors-jeu, et le

policier peut se déplacer pour capturer le suivant). Enfin, suivant que le graphe à tester est orienté ou pas, on appelle la méthode appropriée qui nous retournerait vrai si le graphe donné en entrée était (k, l) -policier-gagnant et faux dans le cas contraire.

Les graphes à tester sont stockés dans des fichiers texte sous formes de listes d'adjacence pour chaque sommet, en plus du nombre de sommets qui se trouve être sur la première ligne (voir tableau 4.4). Lors de l'exécution du programme, on donne en paramètres le nom de fichier texte où est stocké le graphe, en plus du nombre de policier et celui de voleurs.

Exemple 4.3.1 (Fichier texte du graphe donné en entrée). *La représentation du graphe de la figure 4.5 dans un fichier texte est la suivante :*

```

4
2 3
1 4
1 4
2 3 4

```

TABLEAU 4.4. Fichier texte du graphe donné en entrée

On remarque que chaque $(i + 1)$ -ième ligne correspond aux voisins du i -ième sommet (à partir de $i = 1$), puisque la première correspond au nombre de sommets du graphe à tester.

4.4. TEST DE L'IMPLÉMENTATION

Durant l'étape du test de l'implémentation de l'algorithme, on a commencé par tester des petits graphes qu'on savait policier-gagnant, et au fur et à mesure, on est passé à des graphes plus grands. Le temps d'exécution pour ces tests était insignifiant (on parle ici de quelques secondes pour des graphes dont le nombre de sommets ne dépasse pas 15 sommets), ensuite le temps augmentait avec l'augmentation du nombre de sommets (voir tableau 4.5 et 4.6). Pour ce qui

a été de tester les graphes (k, l) -policiers-gagnant, on a commencé à tester des graphes simples, ensuite, on a utilisé des graphes du système de Steiner (ce sont des graphes orientés), et plus précisément un graphe de 7 sommets et engendré par $(1, 2, 4)$, i.e. les sommets $1, \dots, 7$, les arcs $(1, 2), (2, 4)$ et puis ceux obtenus en ajoutant 1 à chaque sommet modulo 7. Lors du test, on a pu vérifier que ce graphe n'était pas policier-gagnant, mais $(2, 1)$ -policier-gagnant en un temps d'exécution relativement petit (273 secondes). Un autre graphe qu'on a testé est le graphe de Peterson (voir figure 4.7). On a pu confirmer que ce graphe-là n'était pas policier-gagnant (étant donné que le graphe possède des cycles de longueur 5), mais $(2, 1)$ -policier-gagnant, et cela en un temps de 5943 seconds.

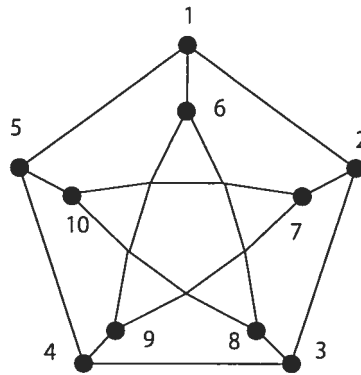


FIGURE 4.7. Graphe de Peterson

Graphes / Nombres de sommets	4	5	7	14	20
(1,1)-policier-gagnant	0 sec	0 sec	0 sec	19 sec	681 sec
(2,1)-policier-gagnant	1 sec	5 sec	273 sec		
(3,1)-policier-gagnant	83 sec				

TABLEAU 4.5. Temps d'exécution de l'algorithme en secondes sur une machine Athlon 750 MHz avec 512 Mo de RAM

Graphes / Nombres de sommets	4	5	7	14	20
(1,1)-policier-gagnant	0 sec	0 sec	0 sec	3 sec	176 sec
(2,1)-policier-gagnant	0 sec	1 sec	73 sec		
(3,1)-policier-gagnant	14 sec				

TABLEAU 4.6. Temps d'exécution de l'algorithme en secondes sur une machine AMD 64 Athlon 3500+ MHz avec 2048 Mo de RAM

4.5. CONSIDÉRATIONS ET OBSERVATIONS

La première fois que nous avons décidé de coder l'algorithme de Hahn et MacGillivray, nous avons dans l'idée de le tester sur les graphes STS dont le nombre de policiers ne dépassera pas 3 et le nombre de voleurs restera 1. C'est pour cela que nous avons utilisé des matrices et listes pour représenter nos graphes. Il existe bien sûr d'autres possibilités, mais qui ne sont utiles dans notre cas que pour un nombre de policiers ou de voleurs très grand. Ainsi, parmi les approches auxquelles nous avons pensées pour la construction des graphes de déplacement des policiers et des voleurs (graphes qui demandaient plus de temps de calcul, même s'ils n'étaient pas très denses) est de créer une collection (liste) de paires formées d'un sommet u_i et d'un entier n_i ($i = 0, 1, 2, \dots, n$), représentant le nombre de sommets qui lui sont voisins. Ensuite, chaque paire serait associée à une liste de sommets qui représentent les voisins du sommet u_i . Cette association se ferait dans les deux sens, i.e., on peut tout aussi bien trouver les successeurs d'un sommet donné, que le prédécesseur d'un sommet. Cette implémentation prend plus d'espace mémoire (on parle ici d'une liste doublement chaînée), mais lorsqu'on veut tester des (k, l) -graphes, dont k ou l est grand, cette approche est plus adéquate, car le temps de traitement pour étiqueter les sommets est de $\Omega\left(\frac{n^{(k+l)}}{(k+l)!}\right)$ comparativement au $\Omega(n^{(k+l) \times 2})$ de la méthode utilisée dans notre implémentation.

Une autre approche à laquelle on pourrait penser, serait d'étiqueter les sommets en même temps qu'on remplit les graphes de déplacements, on y gagnerait en temps de calcul, puisque les deux étapes se feraient en même temps, mais il faut savoir que l'étape d'étiquetage se fait tant que les deux conditions de l'algorithme

ne sont pas remplies. Donc le gain serait moindre, car au lieu de faire le traitement en $c \times (n^{(k+l) \times 2})$, on le fait en $(c - 1) \times (n^{(k+l) \times 2})$, si bien sûr on remplit les graphes de déplacement une seule fois, et non pas à chaque étiquetage, sinon, le temps d'exécution serait encore plus grand.

On pourrait utiliser d'autres approches, mais comme on l'a remarqué pour les approches abordées ci-dessus, si chacune a des avantages, elle a aussi ses propres inconvénients. Ce qu'on pourrait gagner en temps dans, par exemple, la construction des graphes, on risque de le perdre dans le temps d'étiquetage, et vice-versa. De plus, on remarque que le temps d'exécution dépend aussi de la machine sur laquelle on teste le programme, ce qui est logique.

D'après les tests faits, on a remarqué que le temps d'exécution du programme augmentait durant la phase d'étiquetage, ce qui est logique en soi même, étant donné que cette étape se fait tant que les deux conditions d'arrêt ne sont pas atteintes, ce qui fait qu'elle se répète durant c fois, et cette opération se fait en même temps que le parcours des graphes de déplacements des voleurs et des policiers. On pourrait alors se poser la question de savoir si on n'aurait pas dû s'y prendre autrement pour coder cette partie de l'algorithme, seulement, comme on était obligé de parcourir les graphes de déplacement autant de fois que nécessaire, parcours qui consommait le plus de mémoire, et augmentait le temps d'exécution du programme, nous n'avons pas trouvé une autre approche pour contourner ce problème.

CONCLUSION

Dans ce mémoire nous avons étudié les jeux de poursuite dans les graphes, en présentant d'abord une synthèse de ce qui a été réalisé dans ce domaine. Nous avons ainsi donné certaines définitions et notions de la théorie des jeux et des graphes en général, et des graphes policier-gagnants en particulier. Notre contribution à ce mémoire est l'introduction de quelques résultats et quelques directions de recherche futur sur le sujet, plus particulièrement pour des graphes non réflexifs ou partiellement réflexifs, ainsi que la démonstration de certains théorèmes laissés sans preuve par leurs auteurs, telle que le théorème 3.0.5. De plus, nous avons implémenté l'algorithme donné par Hahn et MacGillivray pour les graphes à un policier et un voleur, ainsi que sa généralisation pour les graphes avec k policiers et l voleurs.

Comme on l'a constaté dans ce mémoire, la plupart des travaux abordés considèrent des graphes non-orientés réflexifs. Nous avons essayé de généraliser ces travaux pour n'importe quel type de graphe, qu'il soit réflexif, partiellement réflexif, ou non réflexif, en réadaptant certains théorèmes déjà faits sur ce type de graphe ou en introduisant de nouveaux théorèmes et concepts. De plus, nous avons étudié l'influence des boucles sur le résultat du jeu, et par conséquent, comment utiliser cette propriété de la réflexivité des sommets.

Nous nous sommes bornés aux jeux de poursuite avec information complète, et sans aucune contrainte (que ce soit sur les déplacements des joueurs ou sur la définition de capture). Il serait intéressant de voir ce qui se passe dans le cas où les déplacements des joueurs sont limités ou bien dans le cas où l'information n'est pas complète; certains travaux ont déjà été réalisés dans ce sens là, comme ceux entrepris par Clarke et Nowakowski [12] pour la contrainte sur le déplacement

des joueurs par exemple.

L'implémentation de l'algorithme de Hahn et MacGillivray a été aussi un défi, du fait qu'il fallait optimiser son exécution. Pour cela, il fallait choisir les bonnes structures de données au départ ; on pense qu'on aurait pu faire mieux pour cette partie là du mémoire, étant donné que le temps d'exécution augmentait de façon considérable avec la taille du graphe et le nombre de policiers et de voleurs. Vu que l'algorithme reste valable pour des graphes avec contraintes, il serait intéressant de le tester sur de tels exemples.

BIBLIOGRAPHIE

- [1] M. ADLER, H. RACKE, N. SIVADASAN, C. SOHLER, AND B. VOCKING, Randomized Pursuit-Evasion in Graphs, *Proceedings of the International Colloquium on Automata, Languages and Programming (ICALP)*, (2002).
- [2] R.P. ANSTEE AND M. FARBER, On Bridged Graphs And Cop-win Graphs, *J. Combin. Theory (Series B)*, **44** (1988), 22-28.
- [3] T. BASAR AND G. J. OLSDER, Dynamic Noncooperative Game Theory, *SIAM*, (1998).
- [4] T. BASAR AND P.R. KUMAR, On Worst Case Design Strategies, *Computers and Mathematics with Applications : Special Issue on Pursuit-Evasion Differential Games*, (1987), 13(1-3) : 239-245.
- [5] R.BREISCH, An Intuitive Approach to Speleotopology. *A publication of the Southwestern Region of the National Speleological Society*, (1967), VI : 72-78.
- [6] G. BRIGHTWELL AND P. WINKLER, Gibbs Measures and Dismantlable Graphs. *J. Combin. Theory (Series B)*, **78** (2000).
- [7] I. CHATZIANNAKIS, S. NIKOLETSEAS, AND P. SPIRAKIS, An Efficient Communication Strategy for Ad-Hoc Mobile Networks. *In Proc. of 15th Symposium on Distributed Computing (DISC'2001)*, (2001), 285-299.
- [8] V. CHEPOI, Bridged Graphs are Cop-win Graphs : An Algorithmic Proof, *J. Combin. Theory (Series B)*, **69** (1997), 97-100.
- [9] V. CHEPOI, On Distance-Preserving and Domination Elimination Orderings, *SIAM J. Discrete Math.*, **11** (1998), no. 3, 414-436.
- [10] N.E. CLARKE, The Effects of Replacing Cops and Searchers with Technology, *M.Sc. Thesis, Dalhousie University*, 1999.
- [11] N.E. CLARKE AND R.J. NOWAKOWSKI, Cops, Robber, and Traps, *Utilitas Mathematica*, **60** (2001), 91-98.

- [12] N.E. CLARKE AND R.J. NOWAKOWSKI, Tandem-Win Graphs, (2000).
- [13] J. A. ELLIS, I. H. SUDBOROUGH, AND J. S. TURNER, The Vertex Separation and Search Number of a Graph, *Information and Computation*, (1994), 113(1) : 50-79.
- [14] G. HAHN AND G. MACGILLIVRAY, A Note on k-cop, l-robber Games on Graphs, *Discrete Mathematics*, (2005) .
- [15] G. HAHN AND G. MACGILLIVRAY, A Characterization of k-cop-win graphs and digraphs, (2002).
- [16] P. KASKI AND P. R. J. OSTERGARD, One-Factorizations of Regular Graphs of Order 12, *Electronic Journal of Combinatorics* (2005).
- [17] T. P. KIRKMAN, On a problem in combinations, Cambridge and Dublin, *Math. J.*2(1847). 191-204.
- [18] L. M. KIROUSIS AND C.H. PAPADIMITRIOU, Searching and Pebbling, *Theor. Comp. Sc.*, (1986), 47 : 205-218.
- [19] A. S. LAPAUGH, Recontamination does not help to search a graph, *J. ACM*, (1993), 40 : 224-245.
- [20] N. MEGIDO, S. L. HAKIMI, M. R. GAREY, D. S. JOHNSON, AND C. H. PAPADIMITRIOU, The Complexity of Searching of Graph, *J. ACM*, (1988).
- [21] J. V. NEUMANN, AND O. MORGENSTERN, Theory of Games and Economic Behaviour, *Princeton University Press*(1944).
- [22] C. L. BOUTON NIM, a game with a complete mathematical theory, *Ann. Math.* (1902) 3, 35-39.
- [23] R.J. NOWAKOWSKI AND P. WINKLER, Vertex to Vertex Pursuit in a Graph, *Discrete Math*, **43** (1983), 23-29.
- [24] T. D. PARSONS, Pursuit Evasion in a Graph, In Y. Alavi and D. R. Lick, editors, *Theory and Application of Graphs*, Berlin (1976), Springer Verlag, 426-441.
- [25] A. QUILLIOT, *Thèse d'Etat, Université de Paris VI*, (1983), 1-45.

- [26] A. TAKAHASI, S. UENO, AND Y. KAJITANI, Mixed-searching and Proper-path-width, *Theor. Comp. Sc.*, (1995), 137 : 253-268.
- [27] R. VIDAL, O. SHAKERNIA, J. KIM, D. SHIM, AND S. SASTRY, Probabilistic Pursuit-Evasion Games : Theory, Implementation and Expeimental Evaluation, *IEEE Tramsactions on Robotics and Automation*, (2002), 18 : 662-669.