Université de Montréal

Recherche sur Hunter & Mole
Implémentation de Cops & Robbers

Par

Mathilde Prouvost (20164315)

Bac Sc. Mathématiques et Informatique
Faculté des Arts et des Sciences

Travail présenté à Sébastien Roy
dans le cadre du cours IFT3150
Projet d'informatique – Initiation à la recherche
Supervisée par Gena Hahn

Automne 2020 - Hiver 2021

Résumé

Les problèmes d'évasion consistent à trouver le nombre minimal de poursuivants nécessaires pour attraper un évadé se déplaçant sur un graphe. Alors qu'originalement le projet consistait en de la recherche sur des variantes de ce problème (principalement Cops & Robbers et Hunter & Mole), le but de ce projet est finalement d'implémenter une application permettant de visualiser le problème. L'application devra permettre de dessiner un graphe, de le modifier, de positionner des policiers et un voleur et de les faire se déplacer selon les règles.

Abstract

In pursuit-evasion problem, the minimal number of cops needed to find a robber on a graph has to be found. The first goal of my project was to do some research around the Cops & Robbers problem and its variants (especially the Hunter & Mole problem). But then it changed to the second goal: implement an app to draw a graph to picture Cops & Robbers strategies.

Table des matières

Introduction : vocabulaire et notations

Les graphes

Un *graphe* non-orienté G est un couple (V, E) composé de l'ensemble V des *sommets* (ou *noeuds*) et de l'ensemble E des *arêtes* (paires de sommets) tel que $E \subset \{\{u, v\}, (u, v) \in V^2\}$. On note $n = \text{Card}(V)$ le nombre de noeuds du graphe. Dans un graphe orienté, les arêtes ne sont pas $\{u, v\}$ mais (u, v), i.e l'ordre importe. On peut aussi noter les arêtes $u \cdot v$ ou plus simplement uv.

Un graphe est *simple* si l'ensemble des arêtes ne contient pas de boucle, c'est-à-dire pour tout $u \in V$, $u \cdot u \notin E$.

Le *graphe trivial* est le graphe simple $(\{u\}, \varnothing)$ où u est un sommet unique.

Deux noeuds u, v sont *voisins* si $u \cdot v \in E$. Le *degré* d'un noeud u est le nombre de ses voisins, c'est-à-dire $\text{Card}(\{v \in V, u \cdot v \in E\})$. Une *feuille* est un sommet de degré 1.

Une *chaîne* $u - v$ entre les noeuds u et v est une séquence de t noeuds voisins c'est-à-dire $(u_k)_{k\in[\![1,t]\!]}$ avec $u_1 = u$, $u_t = v$ telle que $\forall k \in [\![1, t-1]\!],\ u_k \cdot u_{k+1} \in E$. t est appelé la *longueur* de ce chemin. Pour un graphe orienté, on parle de *chemin*.

Un *cycle* est une chaîne dont le point d'arrivée est le point de départ (chaîne $u - u$). Pour un graphe orienté, on parle de *circuit*. Le k-cycle est le cycle de longueur k.

Un graphe est dit *connexe* si pour chaque paire de sommets $(u, v) \in V^2$ telle que $u \neq v$, il existe une chaîne reliant u à v.

Un *arbre* est un graphe connexe sans cycle.

Les problèmes d'évasion

Un problème d'évasion est un problème dans lequel sont donnés un graphe, k poursuivants et un évadé. Les $k + 1$ personnages sont positionnés sur des noeuds du graphe et peuvent se déplacer selon les arêtes. Le but de l'évadé est de rester sur des noeuds sans poursuivant, alors que les poursuivants doivent attraper l'évadé, c'est-à-dire être sur le même noeud. Les poursuivants forment une équipe, l'évadé seul une autre. Le jeu va alors être d'alterner les mouvements de chaque équipe. Le problème répond à la question « Les k poursuivants peuvent-ils attraper l'évadé en un temps fini ? » en essayant de trouver des stratégies qui permettent aux poursuivants d'attraper l'évadé ou au contraire des stratégies permettant à l'évadé de s'échapper.

Il en existe plusieurs variantes en fonction de différents possibilités sur le graphes et les personnages. Par exemple, le graphe peut être orienté ou non, l'ensemble des arêtes peut être différent pour les policiers que pour le voleur, les personnages peuvent avoir un champ de vision restreint du graphe, les personnages peuvent ne pas se déplacer à la même vitesse, les personnages peuvent avoir la possibilité de ne pas bouger à leur tour, les personnages peuvent connaître la stratégie adverse, etc. Le problème général de Cops & Robbers se situe sur un graphe simple non-orienté. Les poursuivants sont appelés des policiers et l'évadé un voleur. Tous les personnages ont une connaissance entière du graphe. Le problème Hunter & Mole se différencie du problème Cops & Robbers par le fait que les poursuivants (chasseurs) peuvent se déplacer sans contrainte du graphe, mais ne connaissent pas la position de l'évadé (taupe ou lapin), qui lui ne peut pas ne pas se déplacer et connaît la stratégie des poursuivants en avance. C'est principalement ce problème que j'ai étudié, à travers les papiers [3], [4], [5], [6] et [7].

La question principale de tous ces problèmes est : en fonction de caractéristiques du graphe, quel est le nombre minimal de poursuivants suffisants pour qu'ils attrapent l'évadé ? Notamment, la conjecture de Meyniel en évalue une borne supérieure sur le problème Cops & Robbers à $\mathcal{O}(\sqrt{n})$.

Un graphe est dit *n-Cop-Win* ou *n-Hunter-Win* si n poursuivants sont suffisants pour attraper l'évadé pour les problèmes Cops & Robbers et Hunter & Mole respectivement c'est-à-dire s'il existe une stratégie gagnante pour le poursuivant. Si $n = 1$, on dit simplement *Cop-Win* et *Hunter-Win*, et on peut employer les termes *Robber-Win* et *Mole-Win* si l'évadé peut s'enfuir, c'est-à-dire s'il existe une stratégie gagnante pour l'évadé.

Historique de mon travail

Étude des problèmes d'évasion avec un unique poursuivant ([1])

Cops & Robbers avec un unique policier ([2], 1973)

Le sommet u *domine* le sommet v si pour chaque noeud w voisin de v, w est aussi voisin de u. On dit que v est un *coin*.

Un graphe est dit *pliable* s'il contient une séquence de coins qui mènent au graphe trivial.

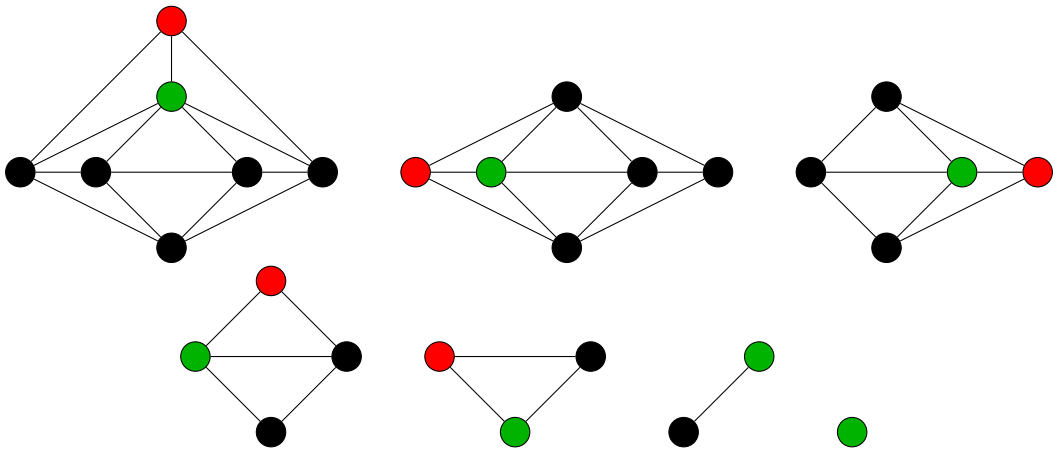


FIGURE 1 **Exemple de construction d'arbre pliable**

Théorème 1. *Pour le problème Cops & Robbers avec un unique poursuivant, un graphe est cop-win si et seulement s'il est pliable.*

Démonstration. $\boxed{\Leftarrow}$ Montrons d'abord que le policier gagne sur les graphes pliables.

Soit $G = (V, E)$ un graphe pliable non trivial. Alors par définition, on a une suite $(u_k)_{k\in[\![0,n-1]\!]}$ de coins menant au graphe trivial au sommet u_{n-1}. On peut alors utiliser les déplacements du policier suivant $(u_{n-1-k})_{k\in[\![0,n-1]\!]}$. A ce moment, le voleur ne pourra pas s'échapper.

En effet, chacun des noeuds visités par le policier ne pourra pas être atteint par le voleur après son passage sans être capturé. On peut supprimer un tel noeud dominé du graphe puisque le voleur ne pourra pas s'y rendre sans être capturé.

On procède alors par induction descendante pour atteindre le cas de base du graphe trivial.

Le policier gagne sur le graphe trivial, car le voleur ne peut pas se positionner ailleurs que sur le sommet déjà occupé par le policier.

$\boxed{\Rightarrow}$ Maintenant, prouvons que si le policier gagne sur un graphe, alors ce graphe est pliable.

Même s'il joue optimalement, le dernier noeud sur lequel le voleur se place est dominé par le policier, sinon il pourrait continuer à jouer. En enlevant ce noeud, le graphe doit toujours pouvoir être gagné par le policier, sinon le voleur n'emprunterait jamais ce noeud et pourrait gagner. On a trouvé notre premier coin.

On procède alors par induction descendante pour atteindre le cas de base du graphe trivial, qui est bien un graphe pliable. □

Hunter & Mole avec un unique chasseur ([3], 2019)

Lemme 2. *Si un graphe H est mole-win, alors tout graphe G ayant H pour sous-graphe est mole-win.*

Démonstration. Si H est mole-win, alors la taupe a une stratégie gagnante sur ce sous-graphe. Il lui suffit de rester sur ce sous graphe pour réussir à échapper au chasseur qui ne pourra donc pas avoir de stratégie gagnante. □

Corollaire 3. *La nature d'un graphe est directement obtenue selon la nature de chacune de ses composantes connexes selon la loi suivante : si une composante est mole-win, alors le graphe est mole-win ; sinon le graphe est hunter-win.*

Démonstration. Si aucune composante connexe n'est mole-win (c'est-à-dire toutes les composantes sont hunter-win), alors le chasseur devra appliquer sa stratégie gagnante sur chaque composante. Ainsi, quelque soit la composante sur laquelle est la taupe, elle se fera attraper, puisqu'elle ne peut pas changer de composante.

Au contraire, si une composante est mole-win, alors on peut utiliser le lemme 2. La taupe pourra utiliser sa stratégie gagnante pour éviter le chasseur, qu'il tire sur sa composante ou non. □

Dans un arbre, on peut définir une *parité* des noeuds. En effet, puisqu'un arbre est connexe, deux noeuds u, v sont toujours connectés donc il existe un chemin entre eux. Si on n'autorise pas les allers-retours sur un chemin, ce chemin est unique puisqu'un arbre ne contient pas de cycle. On peut donc parler de *la* distance entre deux noeuds car celle-ci est unique. Ainsi, en marquant un sommet quelconque appelé *racine*, l'ensemble des noeuds pairs sera l'ensemble des noeuds situés à une distance paire de la racine et l'ensemble des noeuds impairs est l'ensemble des noeuds situés à une distance impaire de la racine. Par les propriétés de la parité dans $\mathbb{N}$, deux noeuds de même parité ne peuvent pas être voisins.

Un *homard* est un arbre constitué d'un chemin $\mathcal{P}$ (le *chemin principal*) et de noeuds situés à une distance maximale de 2 d'un noeud de $\mathcal{P}$.

Les noeuds à distance 1 du chemin principal sont appelés des *genoux*, les feuilles à distance 2 de ce même chemin des *pieds* et les sommets du chemin principal reliés à des genoux des *hanches*.

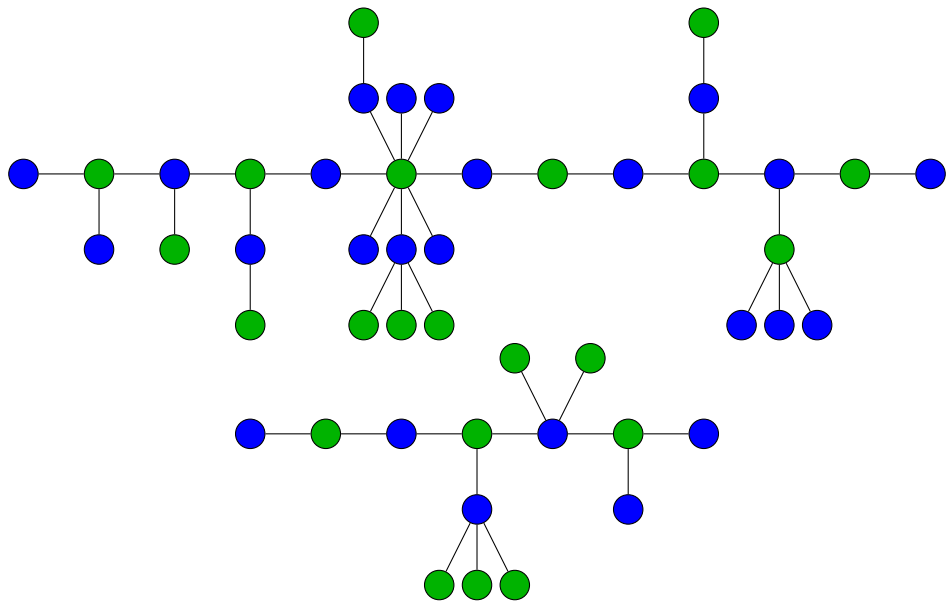


FIGURE 2 **Exemples de homards coloriés selon la parité des noeuds** Les chemins principaux sont horizontaux.

Théorème 4. *Pour le problème Hunter & Mole avec un unique poursuivant, les homards sont hunter-win.*

Démonstration. Montrons que le chasseur gagne sur les homards, en explicitant sa stratégie.

Remarquons d'abord que, puisque la taupe doit se déplacer d'une distance exactement 1 à chaque fois, sa position changera de parité à chaque tour.

Sans perte de généralité, on va fixer la racine à une extrémité du chemin principal. On va faire une disjonction de cas, selon si au premier tour la proie est sur un sommet pair ou impair.

On va d'abord supposer qu'elle est sur un sommet d'une certaine parité et éliminer toutes les possibilités. Si on attrape la taupe, c'est qu'elle était de cette parité. Sinon, c'est qu'elle était sur l'autre parité. Dans ce cas, il faut avoir fait un nombre impair de tour (au besoin, gâcher un tour sans cible apparente) pour se retrouver à la situation initiale avec la taupe dans la bonne parité. Il suffira alors de refaire notre stratégie gagnante pour l'attraper.

Explicitons notre stratégie gagnante sur une taupe de la bonne parité. Un exemple est illustré à la figure 3.

En ne se déplaçant que d'une distance 1, le chasseur va changer de parité de la même façon que la taupe. La stratégie consiste à suivre le chemin principal en se déplaçant d'un sommet à son voisin. En présence de genou, il y a deux possibilités : soit le genou n'a pas de pied, soit il en a un. Si le genou n'a pas de pied, puisqu'il n'y a qu'un sommet à un distance impaire dans la branche du genou, la taupe ne peut pas y être, et donc le chasseur continue. Sinon, la taupe peut se trouver dans un pied. Dans ce cas, le chasseur va devoir aller sur le genou. La taupe n'en étant pas à une distance 1 sur le chemin principal, elle ne va pas pouvoir profiter de ce "détour" pour aller sur un sommet antérieur du chemin principal. Après cela, si elle était effectivement sur le pied, elle se fait attraper sur le genou, sinon, le chasseur devra repasser par la hanche pour conserver sa parité. En présence de plusieurs pieds sur un même genou, tous les pieds arrivent sur le genou du chasseur. En présence de plusieurs genoux à pied, le chasseur devra éliminer les pieds de chacun séparément, mais puisqu'il ne changera jamais de parité, la taupe ne pourra pas en profiter pour s'échapper.

Après avoir parcouru tout le chemin principal et les genoux, le chasseur aura éliminé toutes les positions possibles pour la taupe. S'il ne l'a pas encore attrapée, c'est qu'elle a débuté sur une parité opposée. En attendant un nombre de tours impair depuis le début, la taupe sera alors sur la bonne parité, et refaire la stratégie assure d'attraper la taupe.

Ainsi, avec cette stratégie, le chasseur est assuré d'attraper la taupe en un temps fini, quelle que soit la stratégie de la taupe pour essayer de s'enfuir. □

Lemme 5. *Les cycles de taille $n \geq 3$ sont mole-win.*

Démonstration. D'après les règles, la taupe connaît la stratégie du chasseur. Dans un graphe de taille $n \geq 3$, tous les sommets sont de degré 2. Si le chasseur tire sur un sommet, la taupe avait au tour d'avant la possibilité d'aller sur l'autre sommet qui était relié à sa position.

En fait, l'impossibilité pour le chasseur d'enfermer la taupe dans un cul-de-sac (chemin qui termine sur des feuilles) permet à la taupe de toujours s'évader. □

Lemme 6. *Le graphe $S_{3,3}$, décrit à la figure 4, est mole-win.*

Démonstration. Je ne vais pas expliciter une preuve rigoureuse. Celle-ci est bien écrite dans [3]. Je vais tout de même en expliquer la principale logique.

La démonstration consiste en une disjonction de cas décrite par un diagramme d'états fini, qui montre que quelque soit la stratégie du chasseur, la taupe peut s'échapper. Voici quelques résultats tout de même intéressants à noter.

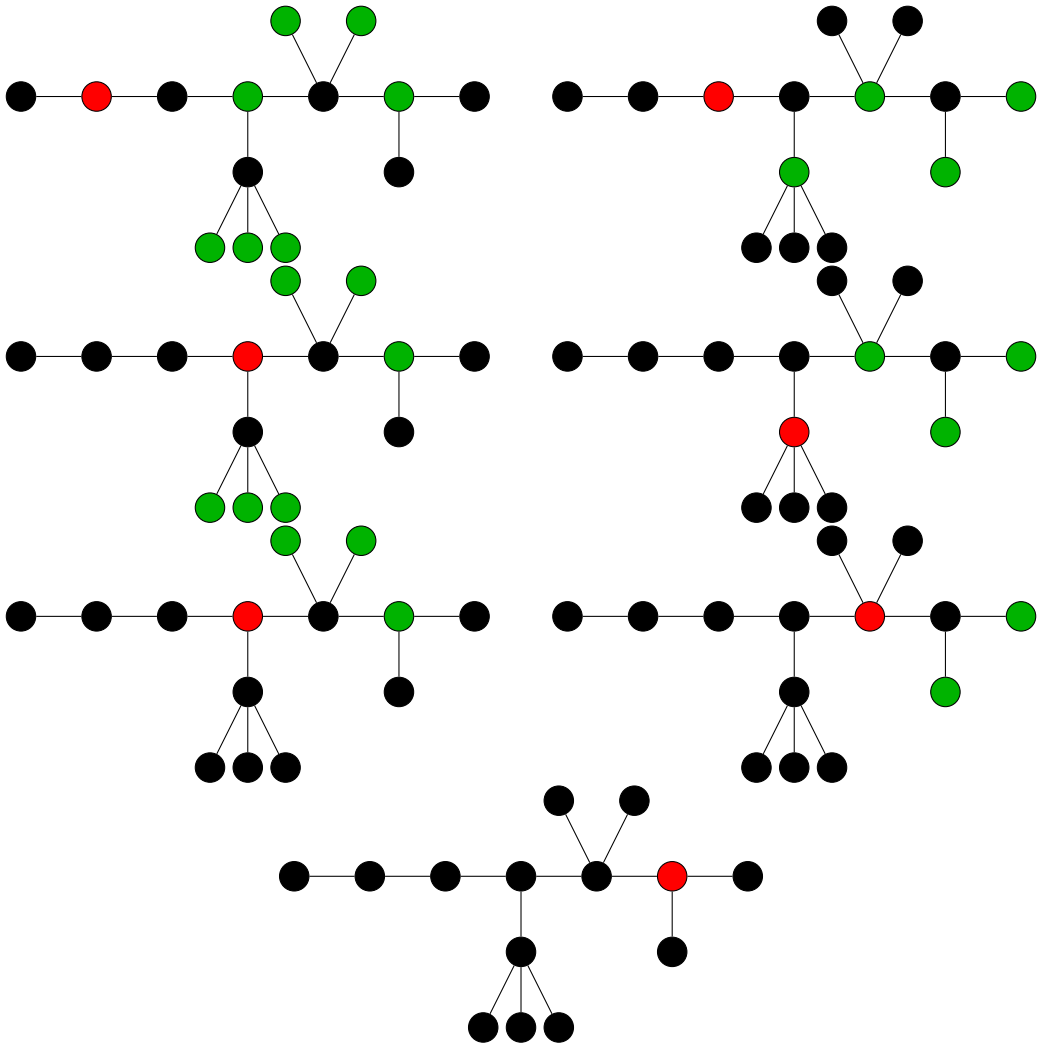


FIGURE 3 **Stratégie gagnante pour le chasseur** En rouge est la position du chasseur, en vert les positions possibles de la taupe. On a commencé par supposer que la taupe était au tour $t = 0$ sur un noeud pair de la même parité de le sommet rouge. Si ce n'était pas le cas, il faut alors refaire la stratégie puisqu'on a joué un tour impair de tours, la taupe qui était à $t = 0$ sur un sommet impair est maintenant sur un sommet pair. Si ça n'avait pas été le cas, refaire le tour $t = 0$ avant de reprendre à ce même tour une fois supplémentaire permet de changer la taupe de parité. Finalement, la taupe ne peut être sur aucun sommet sans avoir été attrapée.

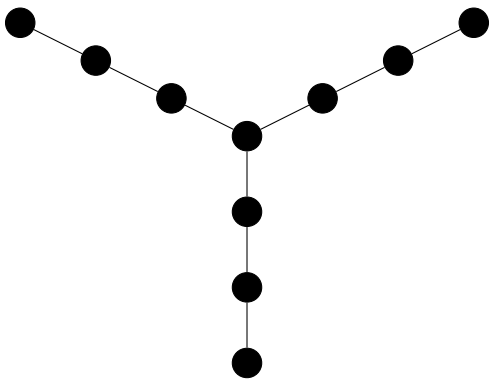


FIGURE 4 Le graphe $S_{3,3}$, aussi appelé *araignée à 3 pattes*

La taupe a ou bien 1 degré de liberté (sur les feuilles), ou bien 2 (au milieu des branches) ou bien 3 (au centre).

Le chasseur peut empêcher la taupe de changer de branche en tirant toujours au centre. Cependant, la taupe a assez de liberté au sein d'une branche pour y rester libre.

Le chasseur peut épuiser les possibilités d'une branche. La taupe pouvait s'assurer de ne pas être sur cette branche précise. En s'acharnant sur une unique branche, le chasseur ne peut pas empêcher la taupe de changer de branche entre temps, en passant par le centre.

Contrairement à un homard, si on considère deux branches de l'araignée comme le chemin principal, rien n'empêche à la taupe d'être sur la troisième branche. Si ce n'est pas le cas, lorsque le chasseur va essayer d'éliminer les possibilités sur celle-ci, la taupe pourra en profiter pour retourner sur les pas du chasseur et ainsi lui échapper. Puisqu'elle connaît la stratégie du chasseur, elle peut s'arranger initialement pour choisir de ne pas être sur cette branche.

Finalement, il faut retenir que c'est la possibilité de revenir sur des sommets vérifiés et éliminés par le chasseur qui permet à la taupe de s'enfuir. □

Théorème 7. *Un graphe G connexe est hunter-win si et seulement si il est un homard.*

Démonstration. Montrons le sens direct en cherchant les graphes connexes hunter-win.

Maintenant, si le graphe n'est pas un arbre, il contient un cycle et il est donc mole-win d'après le théorème 5. Ainsi, seuls des arbres peuvent être hunter-win.

Si c'est un arbre qui n'est pas un homard, alors il contient $S_{3,3}$. En effet, il existe un plus long chemin noté $\mathcal{P}$ car le nombre de noeuds de G est borné (par n) et dans $\mathbb{N}$. Il existe un noeud u à une distance d'au moins 3 de $\mathcal{P}$, sinon notre graphe serait un homard. En notant c le sommet de $\mathcal{P}$ le plus proche de u, il existe deux sommets v_1, v_2 à distance 3 de c sur $\mathcal{P}$, sinon $\mathcal{P}$ passerait par u pour être plus long. En utilisant c comme noeud central et les noeuds entre c et u, v_1, v_2 pour les branches, on a trouvé un sous-graphe de la forme de $S_{3,3}$. D'après le théorème 6, ce graphe est mole-win. Finalement, seuls les arbres qui sont des homards peuvent être hunter-win.

Finalement, d'après le théorème 4, les homards sont hunter-win, et ce sont les seuls graphes connexes qui le sont. □

Implémentation d'une application pour visualiser Cops & Robbers

Tout le code est sur mon GitHub.

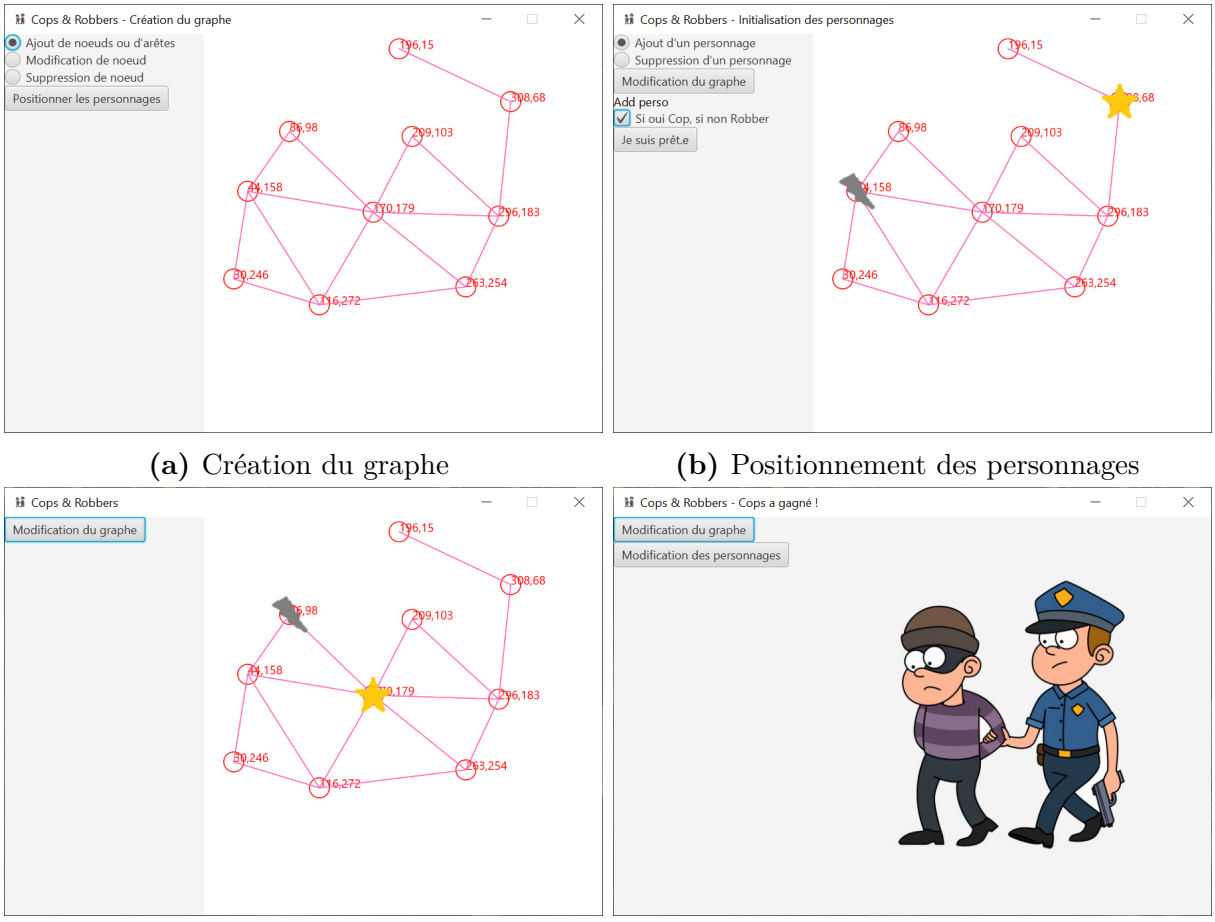
J'ai développé une interface visuelle pour dessiner des graphes et pouvoir jouer aux Cops & Robbers. J'ai opté pour une Programmation Orientée Objet et une logique Modèle-Vue-Contrôleur. J'ai donc préféré utiliser Java (et JavaFX).

L'application s'ouvre sur une fenêtre (figure 5a) où on peut créer des graphes. Grâce à un menu, on peut choisir de créer des noeuds (par clic) et des arêtes (par drag) ; on peut modifier les noeuds (par drag) ; et on peut supprimer des noeuds (par clic).

On peut ensuite positionner les personnages, représenter par un éclair gris pour le voleur et un étoile jaune pour les policiers (figure 5b).

Finalement, on peut jouer aux Cops & Robbers en déplaçant (par drag) les personnages sur des noeuds voisins (figure 5c).

Une fois le voleur attrapé, on a un menu (figure 5d) permettant de ré-initialiser le graphe, avec une option pour modifier le graphe (5a) ou réinitialiser les personnages (5b).



(a) Création du graphe **(b)** Positionnement des personnages

(c) Fenêtre de jeu **(d)** Fenêtre de ré-initialisation

FIGURE 5 Captures d'écran de mon application

Discussion

Limitations

Mon application est fonctionnelle, mais n'a pas encore un visuel attirant.

Un dernier détail à ajouter est la possibilité de faire vraiment des tours, permettant à l'utilisateur de savoir quel personnage a déjà bougé sur ce tour ci, limittant ainsi les déplacements à une distance de 0 ou 1 pour les policiers et les voleurs alternativement.

J'ai eu un problème de santé qui m'a empêché de faire tout ce que je voulais, notamment l'aspect attractif visuellement de mon application.

Bilan et ouverture

Ce rapport est un résumé de mes recherches dans le domaine des théories des graphes et des jeux sur différentes variantes du problème Cops & Robbers en particulier sur Hunter & Mole.

J'ai concentré beaucoup de rigueur à ce rapport, afin qu'il se rapproche d'un article de papier. C'était là le but de mon initiation à la recherche initialement prévue avec Gena Hahn.

Quant à mon application, il est évident qu'elle peut toujours être améliorée. J'ai privilégié la fonctionnalité et la maintenabilité à l'aspect visuel. En effet, le problème de Cops & Robbers peut

facilement être joué et il est facile de modifier le code pour ajouter des fonctionnalités propres aux variantes. Par exemple, il est très facile de permettre aux policiers de ne pas se déplacer uniquement selon les arêtes et ainsi jouer au problème de Hunter & Mole.

Références

[1] N. Komarov, "Hunter vs. mole," (St Lawrence University), 2015.

[2] R. Nowakowski and P. Winkler, "Vertex-to-vertex pursuit in a graph," *Discrete Mathematics*, vol. 43, no. 2, pp. 235–239, 1983.

[3] N. Komarov and P. Winkler, "Hunter vs. mole," 2013.

[4] S. Tayu and S. Ueno, "On evasion games on graphs," in *Discrete and Computational Geometry and Graphs*, Lecture Notes in Computer Science, (Tokyo, Japan), pp. 253–264, Springer, Cham, 2016.

[5] T. V. Abramovskaya, F. V. Fominb, P. A. Golovach, and M. Pilipczuk, "How to hunt an invisible rabbit on a graph," *European Journal of Combinatorics*, vol. 52, no. A, pp. 12–26, 2016.

[6] J. R. Britnell and M. Wildon, "Finding a princess in a palace : A pursuit-evasion problem," 2012.

[7] H. Guggiari, A. Roberts, and A. Scott, "Approximating the position of a hidden agent in a graph," 2018.