

Avancement Snake

Olivier Y.

01/02/2024

Résumé

Dans ce document, nous étudierons l'Hamiltonicité de certains graphes, en particulier des graphes grilles carrés et nous y rechercherons des cycles Hamiltoniens. Nous verrons aussi différentes méthodes de résolution du jeu Snake que l'on comparera pour se rapprocher de la résolution la plus optimale possible (si elle existe).

Table des matières

I	Préliminaires	3
1	Présentation	3
2	Quelques définitions	3
2.1	Graphes	3
2.2	Indicages	4
2.3	Graphes isomorphes	4
2.4	Éléments de graphes	5
2.5	Opérations sur les graphes	5
2.6	Structures particulières	5
2.7	Graphes associés	6
2.8	Autres	6
II	Théorie générale	7
3	Introduction	7
4	Matrices	7
4.1	Généralités	7
4.2	Matrices stochastiques	7
4.3	Matrices d'adjacence	8
4.4	Matrice Laplacienne	10
5	SAT	10
6	Matroïdes	10

7 Faits supplémentaires	10
7.1 Correction de A^*	11
8 Autres	11
III Snake	12
9 Comment avoir la certitude de gagner ?	12
10 Génération de cycle Hamiltonien	12
10.1 Existence	12
10.2 Cycles murables	12
10.3 Génération d'un cycle Hamiltonien avec contrainte	13
11 Méthodes de résolution de Snake	14
11.1 Méthode optimale	15
11.2 Plus court chemin	15
11.2.1 Théorie	15
11.2.2 Résultats et analyse	16
11.3 Plus court chemin sans couper grille	16
11.3.1 Théorie	16
11.3.2 Résultats et analyse	16
11.4 Suivi de cycle Hamiltonien	16
11.4.1 Théorie	16
11.4.2 Résultats et analyse	17
11.5 Raccourcis naïf sur cycle Hamiltonien	17
11.5.1 Théorie	17
11.5.2 Résultats et analyse	19
11.6 Raccourcis sur cycle Hamiltonien (Version 2)	19
11.7 Raccourcis sur cycle Hamiltonien (Version 3)	19
11.7.1 Présentation	19
11.7.2 Résultats et analyse	19
11.8 Raccourcis sur cycle Hamiltonien (Version 3.5)	20
11.9 Monte Carlo tree search	20
11.10 Méthode "Pomme Moins Queue"	20
11.11 Méthode "Pomme Moins Queue" avec cycle Hamiltonien dynamique	20
11.12 Plus court chemin sur cycle Hamiltonien dynamique (Version 1)	20
11.13 Plus court chemin sur cycle Hamiltonien dynamique avec réparation	20
11.14 Plus court chemin sur cycle Hamiltonien dynamique (Version 2)	20
11.15 Plus court chemin dans graphe orienté	20
11.16 Parcours de cellules	20
11.17 Plus court chemin avec vérification post-déplacement	20
11.18 Mixage des méthodes	21
12 Résultats et commentaires	21
citer et vérifier que toutes les citations dans le .bib sont utilisées	

Première partie

Préliminaires

1 Présentation

Snake est un jeu vidéo joué seul sur une grille carrée de dimensions (n, n) où $n \in \mathbb{N}^*$. Le temps est discrétisé de manière régulière. Le joueur contrôle le mouvement de la tête du serpent. Il peut la déplacer dans trois directions orthogonales à chaque pas de temps. Au démarrage du jeu, la position de la tête du serpent est choisie arbitrairement et le serpent n'occupe qu'une case. A chaque instant, une unique pomme est présente sur le terrain, elle occupe une case. Lorsqu'une pomme est mangée par la tête du serpent, une nouvelle apparaît de manière aléatoire et uniforme sur la grille. Une pomme ne peut apparaître sur le corps du serpent. Le corps du serpent suit exactement la tête du serpent et s'allonge d'une unité lorsque la tête du serpent mange une pomme. La tête du serpent ne peut sortir de la grille ni intersecter avec son corps, auquel cas la partie est perdue. La partie est gagnée lorsque le serpent en entier occupe toute la grille, ou de manière équivalente, lorsqu'une nouvelle pomme ne peut apparaître sur la grille.

L'environnement de jeu est alors un environnement stochastique dû à l'aléatoire des positions d'apparition des pommes.

Terminologie On distingue deux parties disjointes du serpent, on appelle tête du serpent la case la plus récente qu'il vient d'atteindre, et sa queue (ou son corps) le reste des cases qu'il occupe à un moment donné.

Conventions Lorsque le serpent mange une pomme, il s'allonge à l'image suivante.

Le serpent apparaît initialement sur la case la plus en haut à gauche de la grille avec un score de 0 et une longueur de 1 (c'est-à-dire qu'il n'est constitué que d'une tête). Le jeu est alors gagné lorsque le serpent a mangé n^2 pommes i.e. lorsque son score est de n^2 .

2 Quelques définitions

Tous les graphes G considérés seront supposés non orientés, sans boucle ni arête multiple sauf si précision.

2.1 Graphes

Définition 2.1 On appelle **graphe non orienté** tout couple $G = (V, E)$ pour V un ensemble et E une partie de $\{\{v_1, v_2\} \mid v_1, v_2 \in V \wedge v_1 \neq v_2\}$.

On appelle **sommets** les éléments de V et V l'**ensemble des sommets** de G .

On appelle **arêtes** les éléments de E et E l'**ensemble des arêtes** de G .

On note Ω l'ensemble des graphes non orientés.

Définition 2.2 On appelle **graphe orienté** tout couple $G = (V, E)$ pour V un ensemble et E une partie de $\{(v_1, v_2) \mid v_1, v_2 \in V \wedge v_1 \neq v_2\}$.

On appelle **sommets** les éléments de V et V l'**ensemble des sommets** de G .

On appelle **arc** les éléments de E et E l'**ensemble des arcs** de G .

Définition 2.3 Soit $G = (V, E)$ un graphe orienté ou non. On appelle **ordre** de G l'entier $\#V$.

Définition 2.4 Soit $G = (V, E)$ un graphe. Pour $(v_1, v_2) \in V^2$, si $\{v_1, v_2\} \in E$ alors on note $v_1 \sim_G v_2$ pour désigner que v_1 et v_2 sont **reliés** (ou **voisins** ou encore **adjacents**) dans G . S'il n'y a pas d'ambiguïté, on notera simplement $v_1 \sim v_2$.

Définition 2.5 Soit $G = (V, E)$ un graphe. Pour $v \in V$, on appelle **degré** de v , noté $\deg(v)$, l'entier $\deg(v) = \#\{v' \in V \mid \{v, v'\} \in E\}$.

Définition 2.6 Soit $G = (V, E)$ un graphe. On dit qu'un graphe $H = (V', E')$ est un **sous-graphe** de G si $V' \subseteq V$.

Définition 2.7 Soit $G = (V, E)$ un graphe. On dit qu'un sous graphe $H = (V', E')$ de G est un **sous-graphe couvrant** de G si $V' = V$.

Définition 2.8 Soit $G = (V, E)$ un graphe. On dit qu'un sous graphe $H = (V', E')$ de G est un **sous-graphe induit** de G si pour tout $(v_1, v_2) \in V'^2$, $v_1 \sim_H v_2$ si et seulement si $v_1 \sim_G v_2$. Dans ce cas E' peut être obtenu en partant de E et en y enlevant les arêtes qui ont une extrémité dans $V \setminus V'$.

2.2 Indicages

Définition 2.9 Soit $G = (V, E)$ un graphe. On appelle **indicage** de G toute bijection $\nu : V \rightarrow \llbracket 1, \#V \rrbracket$.

Définition 2.10 Soient $G = (V, E)$ un graphe. Soit $\nu : V \rightarrow \llbracket 1, \#V \rrbracket$ un indicage de G . On appelle **graphe indicé** par ν de G le graphe, noté $G_\nu = (V', E')$, vérifiant :

$$\begin{cases} V' = \llbracket 1, \#V \rrbracket \\ E' = \{\{\nu(v_1), \nu(v_2)\} \mid \{v_1, v_2\} \in E\} \end{cases}$$

2.3 Graphes isomorphes

Définition 2.11 Soient $G = (V, E)$ et $G' = (V', E')$ deux graphes. On dit que G et G' sont **isomorphes** s'il existe une bijection $\varphi : V \rightarrow V'$ vérifiant :

$$\forall (v_1, v_2) \in V^2 \quad \{v_1, v_2\} \in E \iff \{\varphi(v_1), \varphi(v_2)\} \in E'$$

Dans ce cas, on note $G \cong G'$.

Propriété 2.1 La relation binaire \cong est une relation d'équivalence.

Remarque 2.1 On traite alors tous les graphes de la même classe d'équivalence pour la relation \cong comme égaux à l'exception des graphes indicés.

Propriété 2.2 Soit G un graphe. Soient ν un indicage de G . Alors $G \cong G_\nu$.

Propriété 2.3 Soit $G = (V, E)$ un graphe. Soient ν_1, ν_2 deux indicages de G . Alors $G_{\nu_1} \cong G_{\nu_2}$.

2.4 Éléments de graphes

Définition 2.12 Soit $G = (V, E)$ un graphe orienté ou non. Pour $k \in \mathbb{N}^*$, on appelle **chemin** de G de longueur k tout k -uplets $(p_1, \dots, p_k) \in V^k$ vérifiant $\forall (i, j) \in \llbracket 1, k \rrbracket^2 \quad i \neq j \implies p_i \neq p_j$ et $\forall i \in \llbracket 1, k-1 \rrbracket \quad \{p_i, p_{i+1}\} \in E$.

Remarque 2.2 Un sommet quelconque est un chemin.

Définition 2.13 Soit $G = (V, E)$ un graphe orienté ou non. Pour $k \in \mathbb{N}^*$, on appelle **cycle** de G de longueur k tout chemin $(p_1, \dots, p_k) \in V^k$ vérifiant $\{p_1, p_k\} \in E$.

On définit la notion d'inclusion de tuples :

Définition 2.14 Soit V un ensemble. Soit $(k, l) \in (\mathbb{N}^*)^2$. Soient $A = (a_1, \dots, a_k) \in V^k$ et $B = (b_1, \dots, b_l) \in V^l$. On dit que B est **inclu** dans A , noté $B \subseteq A$, s'il existe $i \in \llbracket 1, k \rrbracket$ vérifiant $B = (a_i, \dots, a_{i+l-1})$ avec la convention $\forall j \in \mathbb{N}^* \quad a_{k+j} = a_j$.

Les cycles et chemins d'un graphe orienté ou non étant des tuples, on définit alors la notion d'inclusion de chemin dans un cycle, de chemin dans un chemin et de cycle dans un cycle.

Définition 2.15 Soit $G = (V, E)$ un graphe. Soient $v_1, v_2 \in V$. On dit que v_1 et v_2 sont **connectés** dans G s'il existe $k \in \mathbb{N}^*$ et un chemin $(p_1, \dots, p_k) \in V^k$ vérifiant $p_1 = v_1$ et $p_k = v_2$.

Définition 2.16 Soit G un graphe. On dit que G est **connexe** (respectivement **fortement connexe**) si G est non orienté (respectivement non orienté) et pour tout couple de sommets $(v_1, v_2) \in V^2$, v_1 et v_2 sont connectés dans G .

Définition 2.17 Soit $G = (V, E)$ un graphe. Soient $(v_1, v_2) \in V^2$. On appelle **chemin Hamiltonien** dans G de v_1 vers v_2 tout chemin $P_H = (v_1, p_2, \dots, p_k, v_2)$ de G de longueur $\#V$.

Définition 2.18 Soit G un graphe. On appelle **cycle Hamiltonien** dans G tout cycle C_H de G passant une unique fois par chaque sommet de G .

Définition 2.19 Soit G un graphe. On dit que G est un **graphe Hamiltonien** s'il existe un cycle Hamiltonien dans G .

Définition 2.20 Soit G un graphe. Soit H un sous-graphe induit de G . On dit que H est un **cycle Hamiltonien induit** par G s'il est 2-régulier, connexe et non orienté.

2.5 Opérations sur les graphes

Définition 2.21 Soient $G_1 = (V_1, E_1), G_2 = (V_2, E_2)$ deux graphes. On définit le **produit cartésien** de G_1 et G_2 , noté $G_1 \square G_2 = (V_1 \times V_2, E_{G_1 \square G_2})$ vérifiant que deux sommets $(v_1, v_2), (v'_1, v'_2) \in V_1 \times V_2$ sont reliés si $(v_1 = v'_1) \wedge \{v_2, v'_2\} \in E_2$ ou $(v_2 = v'_2) \wedge \{v_1, v'_1\} \in E_1$.

2.6 Structures particulières

Définition 2.22 Soit $n \in \mathbb{N}^*$. On appelle **graphe chemin** (d'ordre n) tout graphe noté $P_n = (\{v_1, \dots, v_n\}, E)$ vérifiant $E = \{\{v_k, v_{k+1}\} \mid k \in \llbracket 1, n-1 \rrbracket\}$.

Définition 2.23 Soient $n_1, n_2 \in \mathbb{N}^*$. On appelle **graphe grille rectangulaire** (de dimensions) (n_1, n_2) le graphe noté $\mathcal{G}_{n_1, n_2} = P_{n_1} \square P_{n_2}$.

Dans le cas où $n_1 = n_2 = n$, on notera $\mathcal{G}_n = (\mathcal{V}_n, \mathcal{E}_n)$ et on dira que c'est un **graphe grille carré** (de dimensions) (n, n) .

Définition 2.24 Soit G un graphe. Pour $k \in \mathbb{N}^*$, on dit que G est **k -régulier** si tous ses sommets sont de degré k .

Remarque 2.3 Un graphe cycle est un graphe 2-régulier.

Définition 2.25 Soit G un graphe. On dit que G est **planaire** s'il peut être dessiné sans qu'aucune arêtes ne s'intersecte.

Définition 2.26 Soit $G = (V, E)$ un graphe. On dit que G est **biparti** s'il existe une partition $V = V_1 \sqcup V_2$ vérifiant que chaque arête de G possède une extrémité dans V_1 et l'autre dans V_2 .

Définition 2.27 Soit G un graphe. Pour $k \in \mathbb{N}$, on dit qu'un graphe F est un **k -facteur** de G si c'est un sous-graphe couvrant k -régulier de G .

Remarque 2.4 A222202 (OEIS) pour le nombre de 2-facteurs dans un graphe grille $(2n, 2n)$, ($n \in \mathbb{N}^*$). Pour des dimensions (produit des dimensions) impairs, [2] a prouvé que ça valait 0 (équivalence).

Définition 2.28 Soit G un graphe. On dit que G est **acyclique** s'il ne contient pas de cycle.

Définition 2.29 Soit T un graphe. On dit que T est un **arbre** si c'est un graphe acyclique connexe.

Définition 2.30 Soit G un graphe. Soit T un sous-graphe G . On dit que T est un **arbre couvrant** de G si T est un arbre et que T est un sous-graphe couvrant de G .

2.7 Graphes associés

Définition 2.31 Soit G un graphe planaire. On appelle **graphe dual** de G le graphe $D_G = (V', E')$ vérifiant qu'à chaque sommet $v \in V'$ est associée une face de G et que pour tout couple $(v_1, v_2) \in V'^2$, $\{v_1, v_2\} \in E'$ si les faces correspondantes à v_1 et v_2 ont une arête en commun.

Remarque 2.5 Dans cette définition, on ne considère pas la "face extérieure" au graphe.

Remarque 2.6 Tout graphe grille non nécessairement carré est planaire.

Remarque 2.7 Soit $n \in \mathbb{N}^*$. Le dual D_{G_n} de G_n est isomorphe à G_{n-1} .

Définition 2.32 Soit G un graphe. On appelle **graphe adjoint** de G le graphe $L_G = (V', E')$ vérifiant que $V' = E$ et pour tout couple $(v_1, v_2) \in V'^2$, $\{v_1, v_2\} \in E'$ si les arêtes associées dans G ont une extrémité commune.

Remarque 2.8 Un graphe grille rectangulaire est en particulier le produit cartésien de deux graphes chemins.

2.8 Autres

Définition 2.33 Soient $i, j \in \mathbb{N}$. On définit le **symbole de Kronecker** δ_{ij} par :

$$\delta_{ij} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{sinon} \end{cases}$$

Deuxième partie

Théorie générale

3 Introduction

Définition 3.1 On note **HCP** (**H**amiltonian **C**ycle **P**roblem) la problématique de savoir s'il existe un cycle Hamiltonien dans un graphe donné.

On note **HPP** (**H**amiltonian **P**ath **P**roblem) la problématique de savoir s'il existe un chemin Hamiltonien entre deux points dans un graphe donné.

En 1982, Alon Itai, Christos H. Papadimitriou, et Jayme Luiz Szwarcfiter ([10]) ont prouvé que le HCP dans un graphe grille quelconque est \mathcal{NP} -complet et que le HPP (donc aussi le HCP) étaient \mathcal{P} dans un graphe grille rectangulaire.

C. Umans ([12]) a montré que le HCP était dans la classe \mathcal{P} pour les graphes grilles sans trous.

4 Matrices

Soit $n \in \mathbb{N}^*$. Dans toute cette section, les graphes G considérés seront supposés d'ordre n , non orientés, sans boucle ni arête multiple sauf si précision.

4.1 Généralités

Définition 4.1 Soient $M = (m_{ij})_{1 \leq i, j \leq n}, N = (n_{ij})_{1 \leq i, j \leq n} \in \mathcal{M}_n(\mathbb{R})$. On définit le **produit de Kronecker** entre M et N par :

$$M \otimes N = \begin{pmatrix} m_{11}N & \dots & m_{1n}N \\ \vdots & \ddots & \vdots \\ m_{n1}N & \dots & m_{nn}N \end{pmatrix}$$

Remarque 4.1 Pour $M \in \mathcal{M}_{n_1}(\mathbb{R})$ et $N \in \mathcal{M}_{n_2}(\mathbb{R})$, $M \otimes N \in \mathcal{M}_{n_1 n_2}(\mathbb{R})$

Définition 4.2 Soient $M, N \in \mathcal{M}_n(\mathbb{R})$. On définit la **somme de Kronecker** entre M et N par :

$$M \oplus N = M \otimes I_n + I_n \otimes N$$

Propriété 4.1 Soient $M, N \in \mathcal{M}_n(\mathbb{R})$. On a :

$$Sp(M \oplus N) = \{\lambda + \mu \mid \lambda \in Sp M, \mu \in Sp N\}$$

4.2 Matrices stochastiques

Définition 4.3 Soit $G = (V, E)$ un graphe. Soit $P = (p_{ij})_{1 \leq i, j \leq n} \in \mathcal{M}_n(\mathbb{R})$.

On dit que P est **doublement stochastique** associée à G si elle vérifie :

$$— \forall i \in \llbracket 1, n \rrbracket \quad \sum_{j=1}^n p_{ij} = 1$$

- $\forall j \in \llbracket 1, n \rrbracket \quad \sum_{i=1}^n p_{ij} = 1$
- $\forall (i, j) \in \llbracket 1, n \rrbracket^2 \quad p_{ij} \geq 0$
- $\forall (i, j) \in \llbracket 1, n \rrbracket^2 \quad \{i, j\} \notin V \implies p_{ij} = 0$

On note \mathcal{DS} l'ensemble des matrices doublement stochastiques induites par G .

On peut résoudre le HCP dans un graphe G quelconque via problème d'optimisation ([8, 5], un peu [6]) :

$$\min_{P \in \mathcal{DS}} -\det(I_n - P + \frac{1}{n}J) \quad \text{où } J \in \mathcal{M}_n(\mathbb{R}) \text{ est constituée uniquement de } 1$$

Les chaines de Markov peuvent être utiles pour le HCP : [9], [6], [7]

4.3 Matrices d'adjacence

Définition 4.4 On définit la **matrice d'adjacence** (selon un indiciage fixé) $M_G = (m_{ij})_{1 \leq i, j \leq n}$ de G par :

$$\forall (i, j) \in \llbracket 1, n \rrbracket^2 \quad m_{ij} = \begin{cases} 1 & \text{si } \{i, j\} \in E \\ 0 & \text{sinon} \end{cases}$$

Remarque 4.2 La matrice d'adjacence d'un graphe dépend de l'indiciage choisi. Deux indiciages différents d'un même graphe induisent deux graphes isomorphes.

Propriété 4.2 Soient G et G' deux graphes isomorphes. Alors il existe une matrice de permutation P vérifiant $M_{G'} = PM_G P^T$.

Remarque 4.3 En particulier, les matrices d'adjacence de graphes venant d'indiciages différents d'un même graphe sont orthogonalement semblables.

Propriété 4.3 Pour $\sigma \in S_n$ une permutation, en notant $P_\sigma = (\delta_{i, \sigma(j)})_{(i, j) \in \llbracket 1, n \rrbracket^2}$ sa matrice associée, on a :

$$\chi_{P_\sigma} = \prod_{k=1}^n (X^k - 1)^{C_k} \quad \text{où les } C_i \text{ (} i \in \llbracket 1, n \rrbracket \text{) sont le nombre de } i\text{-cycles dans } \sigma$$

Corollaire 4.1 Une matrice de permutation $P \in \mathcal{M}_n(\{0, 1\})$ est la matrice d'adjacence d'un cycle Hamiltonien si et seulement si $\chi_P = X^n - 1$ ([4])

Définition 4.5 On définit (avec abus) la **matrice d'adjacence symbolique** X_G de G par :

$$\forall (i, j) \in \llbracket 1, n \rrbracket^2 \quad [X_G]_{ij} = \begin{cases} x_{ij} & \text{si } \{i, j\} \in V \\ 0 & \text{sinon} \end{cases} \quad \text{où les } x_{ij} \in [0, 1] \text{ sont quelconques}$$

Le HCP est équivalent à chercher les $x_{ij} \in \mathbb{R}$, $(i, j) \in \llbracket 1, n \rrbracket^2$, qui vérifient ([4]) :

- $\forall (i, j) \in \llbracket 1, n \rrbracket^2 \quad x_{ij}(1 - x_{ij}) = 0 \quad (\text{pour qu'ils soient dans } \{0, 1\})$
- $\forall i \in \llbracket 1, n \rrbracket \quad \sum_{j=1}^n x_{ij} = 1 \quad (\text{la somme sur chaque ligne est égale à } 1)$

- $\forall j \in \llbracket 1, n \rrbracket \sum_{i=1}^n x_{ij} = 1$ (la somme sur chaque colonne est égale à 1)
- $\chi_{X_G} - X^n + 1 = 0$ (pour que X_G soit associé à un cycle Hamiltonien)

Remarque 4.4 Les 3 premières conditions traduisent que X_G est une matrice de permutation et la dernière assure que la permutation associée soit un n -cycle.

Les différents facteurs du déterminant symbolique ($\det(X_G)$) encodent toutes les partitions de G en cycles orientés à sommets disjointes (2-facteurs orientés) ([4]) (et donc potentiellement un cycle Hamiltonien s'il existe, dans ce cas il y en aura "deux fois trop").

Propriété 4.4 On suppose n pair. Soit G un graphe biparti à n sommets. Alors le spectre de M_G est symétrique au sens suivant :

$$\forall \lambda \in \text{Sp } M_G \quad -\lambda \in \text{Sp } M_G$$

Preuve 4.1 Utiliser que M_G est semblable à $\begin{pmatrix} 0 & B^T \\ B & 0 \end{pmatrix}$

où $B \in M_{\frac{n}{2}}(\mathbb{R})$

Propriété 4.5 Soit P_n un graphe chemin à n sommets. Le spectre de M_{P_n} est :

$$\text{Sp } M_{P_n} = \{2\cos(\frac{i\pi}{n+1}) \mid i \in \llbracket 1, n \rrbracket\}$$

Preuve 4.2 à avoir, voir C_{2n+2} , relation de récurrence

Corollaire 4.2 On a :

$$\text{Sp } M_{G_n} = \{2\cos(\frac{i\pi}{n+1}) + 2\cos(\frac{j\pi}{n+1}) \mid (i, j) \in \llbracket 1, n \rrbracket^2\}$$

Propriété 4.6 Soient G_1, G_2 deux graphes à n sommets. La matrice d'adjacence $M_{G_1 \square G_2}$ du produit cartésien de G_1 et G_2 vérifie :

$$M_{G_1 \square G_2} = M_{G_1} \oplus M_{G_2}$$

Corollaire 4.3 On a :

$$M_{G_n} = M_{P_n} \oplus M_{P_n}$$

Propriété 4.7 Soient G un graphe à n sommets. Soit $k \in \mathbb{N}$. G est k -régulier si et seulement

si $\begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$ est vecteur propre de M_G .

Dans ce cas la valeur propre associée est k et pour tout vecteur propre $\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \in \mathcal{M}_n(\mathbb{R})$, on

$$a \sum_{k=1}^n x_k = 1.$$

Preuve 4.3 A ECRIRE

Pour l'équivalence il suffit de regarder la première diagonale de M_G .

Pour la valeur propre associée, calculer.

Pour la somme qui vaut 1, théorème spectral.

4.4 Matrice Laplacienne

Dans toute cette section, on fixe un graphe G non orienté, sans boucle ni arête multiple. On note $n = \#V$ le nombre de sommets de G .

Définition 4.6 On définit la **matrice Laplacienne** L_G de G par $L_G = D_G - M_G$ où $D_G = \text{Diag}((\deg(i))_{i \in \llbracket 1, n \rrbracket})$ et M_G est la matrice d'adjacence de G .

Propriété 4.8 (Théorème de Kirschhoff) Le nombre d'arbres couvrants de G est égal, au signe près, à n'importe quel cofacteur de L_G

Corollaire 4.4 Le nombre d'arbres couvrants $a(n)$ de G vérifie :

$$a(n) = \frac{1}{n} \prod_{\substack{\lambda \in \text{Sp } M_G \\ \lambda \neq 0}} \lambda$$

Propriété 4.9 Soient G_1, G_2 deux graphes à n sommets. On a $([1])$:

$$L_{G_1 \square G_2} = L_{G_1} \oplus L_{G_2}$$

Propriété 4.10 On a :

$$\text{Sp } L_{P_n} = \{2 - 2\cos(\frac{i\pi}{n}) \mid i \in \llbracket 1, n \rrbracket\}$$

Corollaire 4.5 On a :

$$\text{Sp } L_{G_n} = \{4 - 2\cos(\frac{i\pi}{n}) - 2\cos(\frac{j\pi}{n}) \mid (i, j) \in \llbracket 1, n \rrbracket^2\}$$

5 SAT

Le HCP dans un graphe quelconque est équivalent à un problème SAT ([11]) qui est alors \mathcal{NP} -complet

6 Matroïdes

j'arrive

7 Faits supplémentaires

Soit G est un graphe grille (n_1, n_2) sans trou, alors $\#V = n_1 n_2$ et $\#E = n_1(n_2 - 1) + n_2(n_1 - 1)$

Définition 7.1 Soit $G = (V, E)$ un graphe. On note Γ l'ensemble des sous-graphes de G . On définit la loi de composition interne $\oplus : \Gamma \times \Gamma \rightarrow \Gamma$ appelée différence symétrique par : ([12])

$$\forall H_1 = (V_1, E_1), H_2 = (V_2, E_2) \in \Gamma \quad H_1 \oplus H_2 = (V_1 \cup V_2, (E_1 \cup E_2) \setminus (E_1 \cap E_2))$$

Propriété 7.1 (Γ, \oplus) est un groupe abélien d'inverse le graphe vide d'ordre $\#V$.

Propriété 7.2 $\#\Gamma$ est une puissance de 2.

Preuve 7.1 On a pour $H \in \Gamma$, H est son propre inverse.

7.1 Correction de A*

8 Autres

<https://kychin.netlify.app/snake-blog/hamiltonian-cycle/> <https://sites.flinders.edu.au/flinders-hamiltonian-cycle-project/>

A FAIRE

Troisième partie

Snake

Soit $n \in \mathbb{N}^*$. On encodera le plateau de jeu du Snake par un graphe grille $\mathcal{G}_n = (V, E)$ où chaque sommet représente une position atteignable et chaque arête représente un déplacement possible. (Remarquons $\#V = n^2$)

9 Comment avoir la certitude de gagner ?

Sécurité : configuration permettant de suivre une stratégie gagnante si voulu. On prend comme sécurité le fait d'avoir le corps et la tête du serpent sur un cycle Hamiltonien. Une autre sécurité est d'avoir un chemin de la tête au bout de la queue tel qu'il est possible d'éviter un sommet de ce chemin (en en prenant un autre localement).

La première difficulté est de choisir la (ou les) sécurité qui minimise les contraintes de déplacements.

10 Génération de cycle Hamiltonien

A003763 (OEIS.org) pour le nombre de cycles Hamiltoniens dans \mathcal{G}_n . A priori pas encore d'équivalent trouvé.

10.1 Existence

Avant de vouloir générer des cycles Hamiltoniens dans \mathcal{G}_n , assurons nous déjà d'une condition nécessaire et suffisante à leur existence dans \mathcal{G}_n .

Propriété 10.1 \mathcal{G}_n est un graphe Hamiltonien si et seulement si n est pair.

Preuve 10.1 (\Leftarrow) Supposons n pair.

Alors le cycle Hamiltonien canonique (DÉFINIR) convient.

(\Rightarrow) Raisonnons par contraposée et supposons n impair.

Alors $\#V = n^2$ est impair. Or tout cycle de \mathcal{G}_n est de longueur pair. En particulier tout cycle Hamiltonien, s'il en existe, est de longueur pair. C'est impossible. \mathcal{G}_n n'est alors pas Hamiltonien.

Propriété 10.2 Tout graphe planaire 4-régulier est Hamiltonien.

Preuve 10.2 Par Tutte

10.2 Cycles murables

Définition 10.1 On peut générer un cycle Hamiltonien dans \mathcal{G}_n en générant un arbre couvrant dans le graphe grille $\mathcal{G}_{n,1/2}$ de dimensions $(\frac{n}{2}, \frac{n}{2})$ qu'on lui associe. On en fait ensuite le contour dans \mathcal{G}_n . On dit que le cycle Hamiltonien obtenu est **murable** c'est à dire que c'est le contour dans \mathcal{G}_n d'un arbre couvrant de $\mathcal{G}_{n,1/2}$.

Remarque 10.1 Avec cette méthode on peut générer autant de cycle qu'il y a d'arbres couvrants dans $\mathcal{G}_{n,1/2}$.

COMBIEN PAR RAPPORT AU NOMBRE DE CYCLES HAMILTONIENS EXISTANTS ?

Propriété 10.3 *Le nombre $a(n)$ d'arbres couvrants dans \mathcal{G}_n est :*

$$a(n) = \frac{1}{n^2} \prod_{\substack{(k,l) \in \llbracket 0, n-1 \rrbracket^2 \\ (k,l) \neq (0,0)}} (4 - 2\cos(\frac{k\pi}{n}) - 2\cos(\frac{l\pi}{n})) = \frac{1}{n^2} \prod_{\substack{(k,l) \in \llbracket 0, n-1 \rrbracket^2 \\ (k,l) \neq (0,0)}} (4\sin^2(\frac{k\pi}{2n}) + 4\sin^2(\frac{l\pi}{2n}))$$

Preuve 10.3 *Corollaire du théorème de Kirschhoff et du spectre du graphe chemin P_n*

Propriété 10.4 *On a :*

$$a(n) \sim \frac{2^{\frac{1}{4}} \Gamma(\frac{1}{4}) \exp(\frac{4Gn^2}{\pi})}{\pi^{\frac{3}{4}} \sqrt{n} (1 + \sqrt{2})^{2n}}$$

où $G = \sum_{n=0}^{+\infty} \frac{(-1)^n}{(2n+1)^2}$ est la constante de Catalan

Preuve 10.4 *Se fait par somme de Riemann (convergence vers la double intégrale)*

On a déjà $\int_0^1 \int_0^1 \ln(4 - 2\cos(\pi x) - 2\cos(\pi y)) dx dy = \frac{4G}{\pi}$. Et regarder A007341 (OEIS.org)

En utilisant un changement de variable $u=(x+y)/2$ et $v=(x-y)/2$, un développement asymptotique de $\ln(1-x)$ et en s'inspirant de Wallis on peut montrer : (ECRIRE)

$$\int_0^1 \int_0^1 \ln(4 - 2\cos(\pi x) - 2\cos(\pi y)) dx dy = \ln 4 - \frac{1}{2} \sum_{n=1}^{+\infty} \frac{1}{n} \frac{(2n)!^2}{4^{2n} n!^4} = \ln 4 - \frac{1}{2} \sum_{n=1}^{+\infty} \frac{1}{n} \left(\frac{\binom{2n}{n}}{4^n}\right)^2$$

D'où en notant $K = \ln 4 - \frac{1}{2} \sum_{n=1}^{+\infty} \frac{1}{n} \frac{(2n)!^2}{4^{2n} n!^4} > 0$, on a :

$$a(n) = \exp(Kn^2 + o(n^2))$$

10.3 Génération d'un cycle Hamiltonien avec contrainte

Soit $p \in \mathbb{N}^*$. Soit $S = (s_1, \dots, s_p) \in \mathcal{V}_n^p$ un chemin de \mathcal{G}_n . On s'intéresse ici à la problématique de trouver un cycle Hamiltonien H dans \mathcal{G}_n vérifiant $S \subseteq H$ ou de savoir qu'il n'en existe pas.

La méthode de prendre un plus court chemin de la tête au bout de la queue puis "d'allonger" successivement chaque arête ne donne pas forcément un cycle Hami. C'est une méthode gloutonne.

La suite de cette sous-section est basée sur les travaux de C. Umans ([12]).

Définition 10.2 *Soit $G = (V, E)$ un graphe. On appelle **graphe transformé** de G le graphe $G^* = (V^*, E^*)$ construit en partant de G comme suit :*

- 1. *On remplace toutes les arêtes $\{x, y\} \in E$ par les sommets $x', y' \in V^*$ et les arêtes $\{x, x'\}, \{y, y'\}, \{x', y'\} \in E^*$.*
- 2. *Pour tout $x \in V$, on ajoute le sommet $x'' \in V^*$ et les arêtes $\{x'', y\}$ pour y parcourant l'ensemble des voisins de x après l'étape 1.*

Propriété 10.5 *Soit G un graphe. Supposons G biparti, alors G^* est biparti*

Propriété 10.6 *Soit G un graphe. On a qu'il existe un couplage parfait (DEFINIR) dans G^* si et seulement s'il existe un 2-facteur dans G .*

Remarque 10.2 (Passage d'un couplage parfait à un 2-facteur) *Soit $G = (V, E)$ un graphe. Soit (V, M) un couplage parfait. En posant $F = \{\{x', y'\} \mid x, y \in G, \{x', y'\} \notin M\}$, on obtient un 2-facteur (V, F) .*

Comme \mathcal{G}_n est Hamiltonien et qu'un cycle Hamiltonien est un 2-facteur particulier, on a l'existence d'un couplage parfait dans G^* .

Propriété 10.7 *Soit G un graphe. Soit H un sous-graphe de G . On suppose G biparti. Alors H est biparti.*

Preuve 10.5 *Comme G est biparti, il est deux coloriable. Alors H l'est aussi et est alors biparti.*

La contrainte $S \subseteq H$ est équivalente à $E(G_H) \subseteq \mathcal{E}_n \setminus \{\{s_i, v\} \mid i \in \llbracket 1, p \rrbracket, v \in \mathcal{V}_n\}$. DEFINIR E(GN)

Alors (EXPLICITER) pour trouver un deux facteur F de \mathcal{G}_n satisfaisant $S \subseteq F$, on peut supprimer de \mathcal{G}_n^* les arêtes $\{s_i, v\}$ pour (i, v) parcourant $\llbracket 1, p \rrbracket \times \mathcal{V}_n$ puis y chercher un couplage parfait.

Puis en faisant $F \oplus c_1 \oplus \dots \oplus c_k$ pour (c_1, \dots, c_k) une suite de cellules alternées, on réduit le nombre de composantes de F par 1 à chaque différence symétrique, pour finalement obtenir un cycle Hamiltonien (on a l'existence d'une telle suite de cellules alternées). On veille de plus à ce qu'aucune des arêtes des c_i ne coïncident avec le serpent. Mais n'appliquer que des cellules de Type III (on ne sait pas comment trouver les c_i) à un 2-facteur ne suffit pas pour trouver un cycle Hamiltonien H vérifiant $S \subseteq H$ (contre-exemple facile). Il faut utiliser les suites de bandes alternées.

Propriété 10.8 *Soit F un 2-facteur de \mathcal{G}_n . Soit c une cellule de Type III dans $\mathcal{G}_{n,F}$ DEFINIR, alors $F \oplus c$ possède une composante de moins que F .*

Conjecture 10.1 *Soit F un 2-facteur de \mathcal{G}_n . Il existe une cellule de Type III dans $\mathcal{G}_{n,F}$.*

Preuve 10.6 *Supposons par l'absurde que $\mathcal{G}_{n,F}$ ne contient aucune cellule de Type III.*

Si $\mathcal{G}_{n,F}$ contient une cellule de Type IV, alors $\mathcal{G}_{n,F}$ contient une cellule de Type III. Donc $\mathcal{G}_{n,F}$ ne contient pas de cellule de Type IV.

Comme \mathcal{G} est Hamiltonien, $\mathcal{G}_{n,F}$ ne peut pas posséder que des cellules de Type II.

Donc ou bien $\mathcal{G}_{n,F}$ ne contient que des cellules de type I ou bien $\mathcal{G}_{n,F}$ contient au moins une cellule de Type I et une cellule de Type II.

A FINIR

11 Méthodes de résolution de Snake

Lorsque l'on gagne le jeu on est sur un cycle Hamiltonien (ou sur un chemin Hamiltonien). Soit $n \in \mathbb{N}^*$ pair pour que \mathcal{G}_n soit Hamiltonien, cela simplifiera les méthodes envisagées.

REGARDER SI N IMPAIR CE QU'ON FERAIT

11.1 Méthode optimale

On suppose l'unicité d'une stratégie (S) optimale pour gagner au Snake.

Trouver (S) revient à trouver, à chaque pomme mangée le chemin optimal (associé à (S)) jusqu'à la nouvelle pomme.

Il faudrait par exemple trouver un plus court chemin vérifiant que le serpent soit sur un cycle Hamiltonien lorsqu'il mange la pomme (condition (HP)). On a accès aux plus courts chemins (PPC) via A* (plus rapide qu'un parcours en largeur). Ensuite il faut trouver les différents PCC (plus courts chemins) vérifiant (HP), mais le nombre de plus courts chemins semble (A ETRE SUR) croître exponentiellement en fonction de la distance à parcourir.

On peut aussi voir ça d'un point de vue plus dynamique en 2 phases : La première est la considération du mouvement du corps du serpent, si on parcourt beaucoup de sommets avant de se rapprocher du corps du serpent, on n'aura pas besoin de le considérer puisqu'il ne sera déjà plus là. La deuxième est de fixer le graphe obtenu (en retirant des arêtes) lors de la première phase puis de chercher classiquement un cycle Hamiltonien dessus.

On fixe la tête t du serpent à un instant quelconque. Le déplacement du serpent peut être découpé en deux phases : La première est lorsque t est dans le serpent. Le mouvement du bout de la queue peut être prévu. La deuxième lorsque t n'est plus dans le serpent.

Une fois qu'on a trouvé tous les cycles Hamiltoniens vérifiant (HP), il y en aura un "meilleur" qui prévoira le mieux "la suite". Il ne sera pas le meilleur d'un point de vu probabiliste (car prendra en compte des cas qui ne sont pas du tout probable d'arriver). On peut le voir d'un point de vue probabiliste en fonction de là où peut apparaître la pomme pour avoir une méthode non forcément gagnante mais bien plus efficace. Il faut trouver un équilibre entre risques et performances.

On ne peut pas se limiter aux cycles Hamiltoniens murables puisqu'il semble en avoir (PROUVER) exponentiellement moins que des quelconques.

Une question qu'on peut se poser est la nécessité de la condition (HP). En effet, pour une configuration donnée, il y a "toute la première phase de recherche" de "marge" avec le cycle dynamique qui va pouvoir nous donner un cycle Hamiltonien. La seule condition est alors d'éviter la pomme, ça n'est pas une condition à négliger

On se rend compte que la première phase de la recherche (première phase du déplacement du serpent), n'existe plus à partir de la moitié du jeu. C'est pourquoi la condition (HP) est utile (on peut faire sans, mais c'est compliqué de trouver un cycle Hamiltonien dedans en prenant en compte la pomme. Si on mange la pomme en plein milieu, on aura déjà prévu ça donc c'est ok, mais si on en mange une deuxième après cette marge etc.. on peut perdre) à partir de la moitié du jeu.

La méthode la plus rapide sera alors non forcément gagnante (car on va imaginer que les cas peu probables n'arriveront jamais). Il y aura une corrélation entre le taux de victoire et le temps de jeu. On remarque qu'on l'a déjà avec la méthode A* seul (gros risques pour peu de performances), mais pour cette méthode, c'est très peu probable que les pommes soient toutes devant la tête du serpent et forment un cycle Hamiltonien.

11.2 Plus court chemin

11.2.1 Théorie

Fort risque donc bonnes performances peu probables.

Énoncé de la méthode Lorsqu'une pomme est mangée, on trouve un plus court chemin (s'il existe) de la tête du serpent jusqu'à la pomme.

On utilise l'algorithme A* pour trouver un plus court chemin, on prendra la distance Manhattan (déplacements en diagonales non autorisés) lorsqu'il s'agira d'en calculer. De plus, l'ordre de priorité des sommets considérés est : Ouest < Nord < Sud < Est.

On prend en compte les déplacements de la queue lors de la recherche du plus court chemin (seul la première phase est à prendre en compte car si le plus court chemin trouvé s'intersecte, alors il n'est pas de longueur minimale).

11.2.2 Résultats et analyse

Le plus rapide, mais la victoire n'est pas (du tout) assurée.

11.3 Plus court chemin sans couper grille

11.3.1 Théorie

Un peu mieux sûrement.

Énoncé de la méthode A* mais si la grille est coupée en deux, on évite.

11.3.2 Résultats et analyse

jsp

11.4 Suivi de cycle Hamiltonien

11.4.1 Théorie

Énoncé de la méthode Soit H un cycle Hamiltonien. La méthode consiste à simplement suivre le cycle Hamiltonien jusqu'à que la partie soit gagnée.

Propriété 11.1 *La méthode de suivi de cycle Hamiltonien est gagnante est gagnée en $\mathcal{O}(n^4)$ pas en moyenne.*

Preuve 11.1 *La distribution du nombre de pas faits en une partie est $T = n^2 - 1 + \sum_{i=1}^{n^2-1} X_i$ avec $X_i \sim \mathcal{U}([1, i])$ ($[3]$).*

Alors on a :

$$\begin{aligned}
\mathbb{E}(T) &= n^2 - 1 + \sum_{i=1}^{n^2-1} \mathbb{E}(X_i) && \text{Par linéarité de l'espérance} \\
&= n^2 - 1 + \sum_{i=1}^{n^2-1} \frac{i+1}{2} \\
&= n^2 - 1 + \frac{1}{2} \sum_{i=1}^{n^2-1} i + \frac{1}{2} \sum_{i=1}^{n^2-1} 1 \\
&= n^2 - 1 + \frac{n^2(n^2-1)}{4} + \frac{n^2-1}{2} \\
&= \frac{n^2(n^2-1)}{4} + \frac{3(n^2-1)}{2} \\
\mathbb{E}(T) &= \frac{1}{4}(n^2-1)(n^2+6) \\
\mathbb{E}(T) &= \mathcal{O}(n^4)
\end{aligned}$$

Comme voulu.

11.4.2 Résultats et analyse

11.5 Raccourcis naïf sur cycle Hamiltonien

11.5.1 Théorie

Soit $G = (V, E) \in \mathcal{G}_n$. Soit H un cycle Hamiltonien. On se place dans le dual D_G de G . H partitionne les sommets de D_G en deux ensembles $Int_H(D_G)$ et $Ext_H(D_G)$ qui correspondent aux sommets de D_G à l'intérieur et à l'extérieur de H respectivement. (PROUVER)

On choisit un sommet d'origine $v_0 \in V$ (on peut prendre le sommet le plus "en haut à gauche" par exemple). On oriente H de sorte à ce que le sommet entrant à v_0 est "à sa droite". On appelle cette orientation une orientation trigonométrique (regarder le sommet en haut à gauche, dans l'autre cas on appellerait ça une orientation horaire). On numérote les sommets de G par un entier entre 0 et $n^2 - 1$ en commençant par v_0 et en suivant H . Cela nous établit une bijection $\nu_{v_0, H} : V \rightarrow \llbracket 0, n^2 - 1 \rrbracket$ que l'on appelle **numérotation relative** à H depuis v_0 .

Définition 11.1 Pour $v_1, v_2 \in V$, on dit que (v_1, v_2) est un **raccourci** si $\{v_1, v_2\} \in V$ et $1 < \nu_{v_0, H}(v_2) - \nu_{v_0, H}(v_1) < n^2 - 1$.

Définition 11.2 On note S_H l'ensemble des raccourcis relativement à H . On note $S_{Ext, H}$ (resp. $S_{Int, H}$) l'ensemble des raccourcis extérieurs (resp. raccourcis intérieurs) relativement à H .

Définition 11.3 On appelle **arbre associé** à H , noté T_H , le sous-graphe de D_G où deux sommets $v_1, v_2 \in V_{D_G}$ sont reliés si $v_1, v_2 \in Int_H(D_G)$

Propriété 11.2 (Reconfiguration des arbres couvrants) Soient $G \in \mathcal{G}_n$. Soient T, T' deux arbres couvrants de G . Alors il existe $k \in \mathbb{N}^*$ et T_1, \dots, T_k des arbres couvrants de G vérifiant :

$T_1 = T$, $T_k = T'$ et pour $i \in \llbracket 1, k - 1 \rrbracket$, T_i peut être transformé en T_{i+1} via un unique déplacement licite d'une arête.

Conjecture 11.1 *Tout déplacement licite d'une arête de l'arbre associé à H conserve $\#S_{Int,H}$, $\#S_{Ext,H}$, $\#Int_H(D_G)$, $\#Ext_H(D_G)$.*

Preuve 11.2 *à faire*

Propriété 11.3 *On a :*

$$\#Int_H(D_G) = \frac{n^2}{2} - 1$$

$$\#Ext_H(D_G) = (n-1)^2 - \#Int_H(D_G) = \frac{n^2}{2} - 2n + 2$$

$$\#S_{Ext,H} = \frac{n^2}{2} - 2n + 2 = \#Ext_H(D_G)$$

$$\#S_{Int,H} = \frac{n^2}{2} - 2 = \#Int_H(D_G) - 1$$

$$\#S_{Ext,H} + \#S_{Int,H} = \#S_H$$

Preuve 11.3 *Il suffit de se ramener au cycle Hamiltonien canonique H_{can} via une reconfiguration (qui préserve ces quantités) de T_H vers $T_{H_{can}}$ ("peigne")*

Définition 11.4 *Soit $s = (v_1, v_2)$ un raccourci. Soit $v_0 \in V$. On appelle **taux** de s l'entier $\nu_{v_0,H}(v_2) - \nu_{v_0,H}(v_1)$ que l'on notera $\tau_{v_0,H}(s)$.*

Définition 11.5 *Pour un sommet $v \in V$, on appelle **taux de raccourci** de v la somme des taux de tous les raccourcis partant de v , noté $\tau_{v_0,H}(v)$.*

Propriété 11.4 *On a $\#s = n(n-2)$*

Preuve 11.4 *à faire*

Définition 11.6 *On appelle **taux de raccourci** de G relativement à H , noté $\tau_{v_0,H}(G)$, la somme des taux de tous les raccourcis de G dans H*

Remarque 11.1 *On a que $\tau_{v_0,H}(G)$ dépend de l'origine choisie.*

Dans le cadre d'un cycle murable, on peut formaliser légèrement le fait de "faire le tour" de T_H :

Propriété 11.5 *L'ordre des sommets de T_H parcourus (au sens d'être dans la même cellule (à formaliser)) suit un parcours infixe de T_H .*

Alors pour un raccourci intérieur $s \in S_{Int}$, $\tau_{v_0,H}(s)$ augmente linéairement en fonction de la longueur du parcours infixe sur T_H depuis le sommet v_s de D_G que traverse s (à coté). Ceci est aussi vrai lorsqu'on regarde le taux d'un raccourci extérieur.

Le parcours est infixe au vu du choix de l'orientation de H , si H serait orienté de manière horaire, on regarderait plutôt un parcours postfixe.

Dans les deux cas, $\tau_{v_0,H}(s)$ va dépendre linéairement de la taille du sous arbre partant de v_s . Pour minimiser le taux de raccourci intérieur, il faut alors minimiser la taille de tous les sous-arbres de T_H .

On oriente le graphe G pour obtenir un graphe $G_O = (V_O, E_O)$ vérifiant que deux sommets $v_1, v_2 \in V_O$ sont reliés si $\{v_1, v_2\} \in E$ ou (v_1, v_2) est un raccourci dans G .

Le nombre d'orientations possibles d'un graphe est A001174 dans oeis.org

Conjecture/Idée : Avoir "beaucoup" de cycles dans G_O implique que les taux de raccourci des sommets de G sont faibles. Alors on peut conjecturer que le nombre de raccourcis de G est d'autant plus grand que le nombre de cycles dans G_O est faible.

Notons $\#c_{ij}$ le nombre de cycle du sommet indicé i au sommet indicé j dans G_O . Notons pour $v \in V$, $v_+ = \{w \in V \mid \{v, w\} \in V_O\}$. On a :

$$\#c_{0n} = \sum_{v \in V} \sum_{v' \in v_+} \#c_{v'n}$$

$$\tau_H(G) = \sum_{v \in V} \tau_{v_0, H}(v) = \sum_{v \in V} \sum_{v' \in v_+} (\nu_{v_0, H}(v') - \nu_{v_0, H}(v))$$

$\tau_H(G)$ dépend de l'origine choisie (donc théorie difficile dessus) et est moins efficace que réindicer en fonction du sommet que l'on considère et de faire des raccourcis dessus. Dans ce cas, le taux de raccourci est indépendant du cycle Hamiltonien choisit et sera toujours supérieur aux anciens taux.

Énoncé de la méthode On fixe un cycle Hamiltonien H , un sommet d'origine v_0 et une numérotation $\nu_{v_0, H}$ relative à H en partant de v_0 .

Le serpent suit H par défaut. Lorsque le serpent est sur un sommet de taux non nul, il emprunte le raccourci de taux de plus élevé en respectant la condition (HP). Si cette condition n'est pas respectée, le serpent continue de suivre H .

11.5.2 Résultats et analyse

Converge trop vite vers un simple suivit de cycle Hamiltonien

11.6 Raccourcis sur cycle Hamiltonien (Version 2)

Pour un sommet $v \in V$, on fait un changement de numérotation, en soustrayant (modulo $n^2 - 1$) à chaque sommet $\nu_{v_0, H}(v)$. Ainsi le taux est alors indépendant de l'origine, il est même commun à tout les cycles Hamiltoniens. Il est d'ailleurs bien supérieur à l'ancien taux maximal (à "prouver". Alors il est plus efficace de prendre les raccourcis selon une numérotation dynamique.

11.7 Raccourcis sur cycle Hamiltonien (Version 3)

11.7.1 Présentation

On essaie de monter le plus vite de numéro en restant inférieur au numéro de la pomme et du bout de la queue. (Par John Tapsell : <https://johnflux.com/tag/snake/>)

faire qu'on prévienne les prochains mouvements et que la queue peut partir entre temps donc pas besoin de prendre tant de marge (v3.5)

11.7.2 Résultats et analyse

Trous, on regarde alors score en fonction du moment où on ne fait plus que suivre un cycle Hamiltonien. A FAIRE

11.8 Raccourcis sur cycle Hamiltonien (Version 3.5)

Comme la Version 3 mais on prend en compte les futurs déplacements de la queue pour ne pas prendre trop de marge

11.9 Monte Carlo tree search

Ne peut pas marcher car trop de possibilités très différentes.

11.10 Méthode “Pomme Moins Queue”

Converge trop vite vers suivit cycle Hamiltonien

11.11 Méthode “Pomme Moins Queue” avec cycle Hamiltonien dynamique

On remarque que la longueur du plus court chemin de la tête jusqu’à la position désirée dépend du cycle Hamiltonien choisit, on peut donc le modifier en conséquent.

11.12 Plus court chemin sur cycle Hamiltonien dynamique (Version 1)

Victoire non assurée. Souvent bloqué dans des boucles virtuelles.

11.13 Plus court chemin sur cycle Hamiltonien dynamique avec réparation

Faire de la reconfiguration pour tendre vers le cycle Hami canonique (à rotation près) car ça doit être un cycle très modifiable.

SCORER LES CYCLES MODIFIABLE PAR TECHNIQUE DES BOUCLES (par Alpha-Pheonix)

11.14 Plus court chemin sur cycle Hamiltonien dynamique (Version 2)

On avance d’un pas dans la direction d’un PPC s’il existe un cycle Hamiltonien verifiant (HP) (on utilise l’algorithme de Umans et Lenhart)

11.15 Plus court chemin dans graphe orienté

Victoire assurée si on fait des détours si la grille serait coupée en deux.

11.16 Parcours de cellules

Par (??), plus efficace??

11.17 Plus court chemin avec vérification post-déplacement

Victoire non assurée. Si les pommes apparaissent devant la tête du serpent, on perd. Cela n’est pas un sécurité.

Pour la 1er moitié du jeu (recherche dans un graphe dynamique), cela semble efficace.

Ce qu'il fait perdre le serpent c'est si la pomme apparaît 2 fois consécutivement devant la tête du serpent alors qu'elle est à deux cases (sur le chemin) du bout de la queue. C'est peu probable toute la partie mais probable sur la fin de partie.

On voit ici que gagner à tous les coups implique une baisse de performances et que se permettre quelques défaites peut augmenter drastiquement les performances.

Il faut aussi regarder tous (ou au moins plus) les plus courts chemins (voire les "sous-plus courts chemins") pour prendre le meilleur.

11.18 Mixage des méthodes

Faire schéma de compatibilité des méthodes

12 Résultats et commentaires

Références

- [1] Sasmita Barik, Ravindra B. Bapat, and Sukanta Pati. On the laplacian spectra of product graphs. *Applicable Analysis and Discrete Mathematics*, 9 :39–58, 2015.
- [2] Olga Bodroža-Pantić and Ratko Tošić. On the number of 2-factors in rectangular lattice graphs. *Publications de l’Institut Mathématique. Nouvelle Série*, 56 :23–33, 01 1994.
- [3] Kevin Du, Ian M. Gemp, Yi Wu, and Yingying Wu. Alphasnake : Policy iteration on a nondeterministic np-hard markov decision process. In *AAAI Conference on Artificial Intelligence*, 2022.
- [4] Vladimir Ejov, Jerzy A. Filar, Stephen K. Lucas, and Joel L Nelson. Solving the hamiltonian cycle problem using symbolic determinants. *Taiwanese Journal of Mathematics*, 10 :327–338, 2006.
- [5] Vladimir Ejov, Jerzy A. Filar, Walter Murray, and Giang T. Nguyen. Determinants and longest cycles of graphs. *SIAM J. Discret. Math.*, 22 :1215–1225, 2008.
- [6] Vladimir Ejov and Giang T. Nguyen. Consistent behavior of certain perturbed determinants induced by graphs. *Linear Algebra and its Applications*, 431 :543–552, 2009.
- [7] Jerzy A. Filar. Controlled markov chains, graphs, and hamiltonicity. *Found. Trends Stoch. Syst.*, 1 :77–162, 2007.
- [8] Michael Haythorpe. Finding hamiltonian cycles using an interior point method. 2010.
- [9] Michael Haythorpe. Markov chain based algorithms for the hamiltonian cycle problem. 2013.
- [10] Alon Itai, Christos Papadimitriou, and Jayme Szwarcfiter. Hamilton paths in grid graphs. *SIAM J. Comput.*, 11 :676–686, 11 1982.
- [11] Anatoly D. Plotnikov. A logical model of hcp. *International Journal of Mathematics and Mathematical Sciences*, 26 :679–684, 2001.
- [12] Christopher Umans. An algorithm for finding hamiltonian cycles in grid graphs without holes. 1996.