Structure de données II

Projet: listes des heuristiques et attribution par groupes

Université de Mons – Année académique 2021 – 2022

Professeur V. Bruyère – Assistant G. Devillez

1 Introduction

Ce document fait suite au document "Enoncé du projet" dans lequel il est précisé que, par rapport à la construction des arbres BSP, chaque groupe d'étudiants fournira:

- une heuristique construisant un arbre BSP aléatoirement (tel que décrit dans le chapitre 12 de [1]);
- une heuristique construisant un arbre BSP en choisissant de partitionner le plan en prenant la droite qui contient le premier segment de la liste, puis le second, ...
- une autre heuristique spécifique pour créer un arbre BSP et qui sera attribuée à ce groupe d'étudiants.

Ce document contient la liste des heuristiques spécifiques et l'attribution de ces heuristiques aux groupes.

Veuillez noter que vous êtes *libre d'implémenter plus que les 3 heuristiques* demandées ci-dessus – dans le but d'étoffer les résultats de vos tests :

- soit en ajoutant une heuristique décrite dans ce document mais qui ne vous a pas été attribuée;
- soit en imaginant vous-même une autre façon de construire de "bons" arbres BSP.

2 Notations et définitions

Nous supposons que nous travaillons dans le plan $\mathbb{R}^2$.

Plans et régions

Dans $\mathbb{R}^2$, une droite d peut être représentée par une équation $ax + by + c = 0$. Chaque droite d divise le plan en trois parties : deux demi-plans que l'on note d^+ et d^-, et la droite d elle-même. Par exemple, la droite $d \equiv 2x + 3y - 6 = 0$ divise le plan en (illustration en figure 1) :

- un demi-plan ouvert *positif* d^+, qui représente la partie du plan constituée de tous les points dont les coordonnées (x, y) respectent l'inégalité $2x + 3y - 6 > 0$;
- un demi-plan ouvert *négatif* d^- qui est quant à lui défini par l'inégalité $2x + 3y - 6 < 0$;
- la droite d.

L'intersection d'un ensemble fini d'inégalités (représentant des demi-plans) définit donc une *région du plan*. Par exemple, les 4 inégalités suivantes représentent une région de la forme d'un carré :

$$
\begin{aligned}
x - 1 &< 0 && (1)\\
x &> 0 && (2)\\
y - 1 &< 0 && (3)\\
y &> 0 && (4)
\end{aligned}
$$

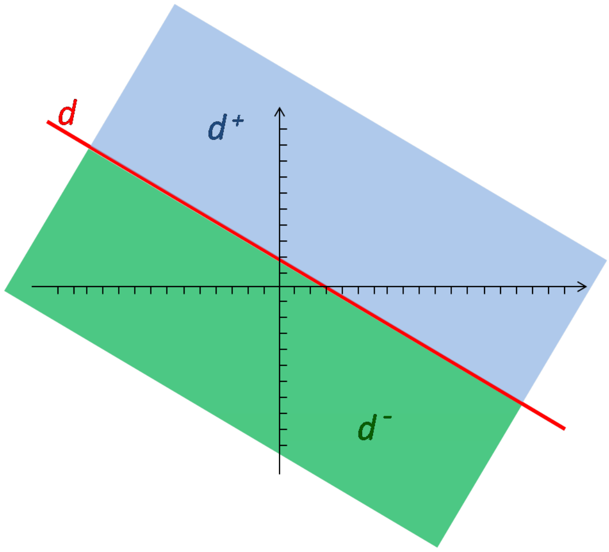

Figure 1: Division du plan ($\mathbb{R}^2$) en 3 parties par la droite $d \equiv 2x + 3y - 6 = 0$.

Arbres BSP

Soit S un ensemble de segments dans $\mathbb{R}^2$. Soit $\mathcal{T}$ un arbre BSP associé à l'ensemble des segments S. Rappelons que chaque nœud interne v de $\mathcal{T}$ contient l'équation d'une droite d_v. Le sous-arbre gauche de v est un arbre BSP pour le demi-plan d_v^- et le sous-arbre droit pour d_v^+. A chaque nœud v de l'arbre, on peut donc associer :

- une région $\mathcal{R}_v$ du plan (déterminée par les droites définies dans les nœuds allant de la racine jusqu'au nœud v);
- un ensemble I_v des segments se trouvant dans $\mathcal{R}_v$: soit un segment de S, soit un fragment d'un segment de S.

Soient un nœud v donné et un segment $s \in I_v$. On note d la droite supportant le segment s. Nous pouvons alors calculer, parmi l'ensemble de segments I_v, les valeurs suivantes :

- le nombre f_d de segments de I_v intersectés par d;
- le nombre f_d^+ de segments de I_v (entièrement) contenus dans d^+;
- le nombre f_d^- de segments de I_v (entièrement) contenus dans d^-;
- la proportion σ_d de segments intersectés par d définie par

$$\sigma_d = \frac{f_d}{|I_v|}.$$

- le nombre g_s de droites d' supportant un segment $s' \neq s$ de I_v qui intersectent le segment s.

3 Heuristiques de construction des arbres BSP

Idée de base commune à toutes les heuristiques

Pour construire un nœud v d'un arbre BSP, on va déterminer quelle est la "meilleure" droite d^* qui va lui être attribuée en fonction de l'heuristique imposée :

1. pour tout segment $s \in I_v$,
   (a) déterminer la droite d qui supporte le segment s;
   (b) évaluer la "qualité" de d selon les critères de l'heuristique.
2. la droite d^* attribuée au nœud v est la "meilleure" selon ces critères.

La spécificité des heuristiques décrites ci-après dépend de la manière dont on définit la "qualité" d'une droite.

3.1 Heuristique H_1

Auteurs: Krishnaswamy, Alijani, Su [2]
Détermination de la partition: on choisit un segment s qui *maximise*

$$g_s.$$

Intuition: En maximisant g_s, espérer que par la suite, le nombre f_d de fragments sera petit.

3.2 Heuristique H_2

Auteurs: Krishnaswamy, Alijani, Su [2]
Détermination de la partition: on choisit un segment s et la droite d qui le supporte, qui

- *minimisent* f_d;
- s'il y a plusieurs tels segments s, on en choisit un qui *maximise* g_s.

Intuition: Minimiser le nombre des segments interceptés par d.

3.3 Heuristique H_3

Auteurs: Thibault, Naylor [4]
Détermination de la partition: on choisit une droite d qui *maximise*

$$f_d^+ \, f_d^- - w f_d,$$

où w est un poids à ajuster[1]. Vous êtes invités à faire des expérimentations sur la valeur de w.
Intuition: Equilibrer le nombre de segments de chaque côté de d, et minimiser le nombre de segments intersectés par d, avec priorité donnée à l'équilibre.

3.4 Heuristique H_4

Auteur: Teller [3]
Détermination de la partition: on choisit une droite d qui

- *maximise* σ_d si $\sigma_d \geq \tau$,
- *minimise* f_d sinon,

où τ est un seuil fixé[2]. Vous êtes invités à faire des expérimentations sur la valeur de τ.
Intuition: S'il y a beaucoup de segments colinéaires, cette heuristique va couper peu de segments.

3.5 Heuristique H_5

Implémentez l'amélioration *free splits* de l'heuristique aléatoire, telle que décrite à la page 265 du Chapitre 12 du livre de référence [1] (donné sur moodle).

4 Attribution des heuristiques spécifiques à chaque groupe

Les attributions suivantes ont été générées aléatoirement.

[1]Les auteurs proposent d'utiliser $w = 8$ dans le cas d'un ensemble de rectangles S dans l'espace
[2]Les auteurs proposent d'utiliser $\tau = 0.5$

N°	Étudiant 1	Étudiant 2	Heuristique
1	Kouamé Brou	Valentin Marchand	H_4
2	Simon MICHEL	Virgil SURIN	H_2
3	Roméo IBRAIMOVSKI	Maxime NABLI	H_3
4	Alvaro GOMILA OTERO	Amira MASTOURI	H_5
5	Yorick ESTIEVENART	Hugo VENTUROSO	H_1
6	Ingrid FONDJA TCHOUMBA	Emmanuel Pondi Djon	H_4
8	Clément SAMAIN	Nicolas Sournac	H_3
9	Noa Foucoux	Adrien Vandekerckhove	H_1
10	Nathan AMORISON	Emilien LEMAIRE	H_5
11	Lorentz BOIVIN	Amanzio DI NICOLO	H_2
12	Moustapha AIT SALAH	Ali Tazribine	H_4
13	Louis DASCOTTE	Nicolas Gerard	H_2
14	Jean Koagne Takam	Christian Menkoncho	H_3
15	Maksym IAKOVENKO	Alessandro SPINOSI	H_1
16	Valentin Gailliez	Adjany Linda TATCHOU KEMADJOU	H_5
17	Matilde SUBTIL GALVAO	Nicolas VALLOIS	H_2
19	Philippe Etene	Gerard Pongo Elonge	H_3
20	Alexandre-Alain Delcour	Lionnel MBIANDA DAHGANG	H_4
21	Loris Danese	Clément MASSE	H_5
23	Ogün ARK	Nicolas MOREAU	H_1

Références

[1] de Berg, M., van Kreveld, M., Overmars, M., and Schwarzkopf, O. *Computational Geometry – Algorithms and Applications.* Second Ed., Springer, 2000.

[2] Krishnaswamy, R., Alijani, G.S., and Su, S.-C. On constructing binary space partitioning trees. In *CSC '90: Proceedings of the 1990 ACM annual conference on Cooperation* (New York, 1990), ACM Press, pp. 230–235.

[3] Teller, S.J. *Visibility Computations in Densely Occluded Polyhedral Environments.* PhD thesis, University of California, 1992.

[4] Thibault, W.C., and Naylor, B.F. Set operations on polyhedra using binary space partitioning trees. In *Proc. SIGGRAPH 87, Comput. Graph.* (1987), vol. 21, ACM SIGGRAPH, pp. 153 – 167.