

Structure de données II

Rapport de projet

HUYLENBROECK Florent
DACHY Corentin
BA3 Info

15 avril 2019

Contents

1	Introduction	3
1.1	Notions théoriques	3
1.2	Enoncé du problème	3
2	Calcul de la complexité de l'algorithme du peintre	4
3	Comparaison des heuristiques	5
4	Mode d'emploi du logiciel de test	6
5	Conclusion	7

1 Introduction

1.1 Notions théoriques

- Une **scène** est un ensemble de segments colorés dans un repère fini. Dans ce rapport, nous nommerons une scène S .
- Un **BSP** (*Binary Space Partition* ou *partition binaire de l'espace*) est un arbre binaire permettant de représenter une scène en fragmentant celle-ci en sous-espaces convexes, lesquels contiennent au plus un segment. Son fonctionnement est le suivant :

Soit v la racine de l'arbre τ .

- Si v est une feuille, alors $S(v)$ (sous-ensemble de segments de la scène contenus dans v) contient 0 ou 1 segment.
- Sinon v est un noeud interne du BSP. Dans ce cas, v contient une équation de droite de la forme $\ell \equiv ax + by + c = 0$ et un ensemble $S(v)$
Le fils gauche v_{gauche} de v sera la racine du sous-BSP τ^- décrivant la partie de la scène située sous la droite ℓ , et le fils droit v_{droit} de v sera la racine du sous-BSP τ^+ décrivant la partie de la scène située au-dessus de la droite ℓ .
 $S(v)$ contient les segments totalement inclus à la droite ℓ . Leur nombre n'est pas limité comme dans les feuilles.

Si une droite de séparation ℓ_v d'un noeud interne v intersecte un segment en un point autre qu'une extrémité, ce segment est coupé en deux parties et celles-ci sont ajoutées dans τ^- et τ^+ en conséquence.

- La **taille** d'un BSP est le nombre total de segments contenus dans chacun des noeuds de celui-ci.
- Une **heuristique** de construction d'un BSP intervient au moment de choisir le segment de la scène par lequel passera la prochaine droite de séparation ℓ .

1.2 Enoncé du problème

Pour ce projet, il nous a été demandé à partir de fichier `.txt` au format Scene2D de :

- Construire un BSP de la scène décrite dans le fichier.
- Appliquer l'algorithme du peintre sur ce BSP.
- Fournir une application ayant deux modes, console et graphique, afin de tester différentes heuristiques de construction pour les BSP. L'application console doit aussi mesurer la hauteur/taille du BSP, le temps CPU pris par la construction et par l'exécution de l'algorithme du peintre.
- Trois heuristiques nous ont été imposées pour ce projet :
 - RANDOM - Le segment est choisi au hasard parmi $S(v)$.
 - ORDERED - Le segment choisi est le premier dans $S(v)$.
 - FREE_SPLIT - Le segment choisi est le premier dans $S(v)$ dont les deux extrémités sont chacune sur une droite de séparation ℓ . Si aucun segment satisfaisant ces conditions n'est trouvé dans $S(v)$, alors le segment est choisi au hasard dans $S(v)$.

2 Calcul de la complexité de l'algorithme du peintre

3 Comparaison des heuristiques

4 Mode d'emploi du logiciel de test

Mode graphique

Ouvrir un terminal, se déplacer dans le dossier *code* du logiciel et exécuter la commande :

```
$ ant testgui
```

Le logiciel consiste en une fenêtre séparée en deux parties. La partie du haut permet de visualiser le BSP et celle du bas ce que voit l'oeil. Afin d'ouvrir un BSP, il faut se rendre dans le menu *Actions* → *Open* et choisir un fichier *Scene2D* valide. Il est possible de changer l'heuristique de construction via le menu *Heuristics*.

Mode console

Ouvrir un terminal, se déplacer dans le dossier *code* du logiciel et exécuter la commande :

```
$ ant test
```

Il est possible de spécifier les conditions de départ des test grâce aux paramètres suivants :

- *path* - Le chemin jusqu'au fichier *Scene2D*. La valeur par défaut est "octogone.txt".
- *n* - Le nombre de répétitions du test. La valeur par défaut est 100.
- *ex* - L'abscisse de l'oeil. La valeur par défaut est 0.
- *ey* - L'ordonnée de l'oeil. La valeur par défaut est 0.

Pour modifier ces paramètres, il faut utiliser le modificateur *-D*.

```
$ ant test -Dpath=somepath.txt
```

Pour modifier plusieurs paramètres, plusieurs *-D* sont nécessaires.

```
$ ant test -Dpath=somepath.txt -Dn=1000 -Dex=10.05 -Dey=3.141
```

Javadoc

Ouvrir un terminal, se déplacer dans le dossier *code* du logiciel et exécuter la commande :

```
$ ant doc
```

5 Conclusion