

Tache 8 Partie 1

Courbe_hilbert

Courbe_hilbert_7

Le test se termine sans aucun souci avec un temps user

=====

./test_simplification-courbe_hilbert_7 39.09s user 24.90s system 89% cpu 1:11.48 total

Courbe_hilbert_8

Le test compte le nombre des contours mais la simplification ne se termine pas (killed)

=====

zsh: killed ./test_simplification-courbe_hilbert_8

./test_simplification-courbe_hilbert_8 44.85s user 28.77s system 82% cpu 1:28.89 total

Si on prend en compte la taille de l'image qui est 511x511, avec un nombre de segment total de 262144, pour d=0 et vu qu'il y a un seul contour, la fonction récursive de simplification est trop surchargée. En fait, la taille des listes chaines continue d'augmenter jusque qu'on arrive un moment pic de mémoire utilisé – définie par le system d'exploitation.

L'ordinateur prendre la décision d'arrêter le processus pour protéger son système central et les autres applications qui ont besoin de la RAM, par rapport au stack existant.

En architecture Mac, il n'y a pas de limite par rapport aux temps accumulé en CPU. Pourtant, il y a un mécanisme sur la machine qui dit que si une seule opération dépasse la limite de 80GB de mémoire sur une seule opération, on va faire un kill de l'opération cela pourrait être une opération qui voudrait endommager le système d'exploitation. On a observé que dans une période de 10 seconds, le programme a consommé 20GB de RAM qui était sauvegardé dans une partie du disque dure pour donner de la place dans la mémoire RAM pour le reste des opérations.

On peut conclure que le fait que le programme est arrêté, ne dépend pas par la taille de l'image, mais il dépend de nombre des segments à simplifier, qui est analogue au mémoire utilisé (par les listes chaines et les listes chaines des contours simplifiés).

Courbe_hilbert_9

On a observé le même comportement que pour Courbe_hilbert_8. Ici on a :

zsh: killed ./test_simplification-courbe_hilbert_9

./test_simplification-courbe_hilbert_9 46.02s user 29.62s system 90% cpu 1:23.93 total

Courbe_hilbert_10

On a observé le même comportement que pour Courbe_hilbert_8. Ici on a :

zsh: killed ./test_simplification-courbe_hilbert_10

./test_simplification-courbe_hilbert_10 48.87s user 29.84s system 92% cpu 1:25.13 total

Zebres

Zebres-2000x1500

Le test se termine sans aucun souci avec un temps user

```
=====
./test_simplification-zebres-2000x1500 36.70s user 20.66s system 86% cpu 1:06.20 total
```

Zebres-1000x0750

Le test se termine sans aucun souci avec un temps user

```
=====
./test_simplification-zebres-1000x0750 6.25s user 3.72s system 87% cpu 11.434 total
```

Zebres-3000x2250

Le test compte le nombre des contours mais la simplification ne se termine pas (killed)

```
=====
zsh: killed ./test_simplification-zebres-3000x2250
./test_simplification-zebres-3000x2250 57.92s user 28.64s system 89% cpu 1:36.78 total
```

On a observe que la mémoire qu'on a besoin pour tous les operations de simplification de cette image est un peu pres 63GB. Mais meme si on n'a pas arrive au pic de 80GB de protection d'ordinateur, comme explique pour l'image Courbe_hilbert_8.

Dans le cas de cette image, le stress mémoire a arrivé à 100% et c'est pourquoi le programme était arrete. En fait, plus il y a de "stress", moins la mémoire traite efficacement les données.

Zebres-4000x3000

Le test compte le nombre des contours mais la simplification ne se termine pas (killed)

On a observé le même comportement que pour Courbe_hilbert_8. Ici on a :

```
=====
zsh: killed ./test_simplification-zebres-4000x3000
./test_simplification-zebres-4000x3000 66.76s user
30.00s system 84% cpu 1:54.34 total
```

