Rapport de ALGAV

Caculer rectangle minimum convexe et cercle minimum convexe

Author :
Qiwei XIAN

Professeur :
Prof.Bui-Xuan

17 décembre 2019

Table des matières

1 Algorithme Toussaint

1.1 Introduction

Donner une ensemble $\mathcal{P}$ de n points dans $\mathbb{R}$, le rectangle minimum convexe est le rectangle minimum qui entourne tous les points de $\mathcal{P}$.

Calculer le rectangle minimum convexe est un des problèmes classiques de la géométrie algorithmique. Pour résoudre ce problème, un majorant de $\mathcal{O}(n^2)$ est donné par l'algorithm recherche exhaustive. Il existe aussi les autre meilleurs algorithmes comme **l'algorithme Shamos**, le premier fois que Michael Shamos l'a utilisé afin de calculer la diamère dun polygon convexe en temps $\mathcal{O}(n)$ en 1978. En plus **l'algorithme Toussaint**, Godfried Toussaint a résolu beaucoup de problèmes de la géométrie algorithmique en utilisant la phase "**rotating caliper**". Dans mon projet, je cherche à calculer le rectangle minimum convexe en utilisant l'algorithme Toussaint.

Néanmoins il faut trouver l'enveloppe convexe de $\mathcal{P}$ avant d'utiliser ces algorithmes. Pour faire ce précalcul, il y a plusieurs choix possibles. Par exemple le **parcours de Graham**, il nous permet de calculer un enveloppe convexe en temps $\mathcal{O}(n \log n)$. Un autre algorithme la **marche de Jarvis**, il a aussi une excellente complexité en $\mathcal{O}(nh)$ où h représente le nombre de sommets de l'enveloppe convexe.

Dans ce projet, je choisis la marche de Jarvis afin de précalculer l'enveloppe convexe de $\mathcal{P}$. Dès que obtenir l'envelloppe, je vais utiliser rotating caliper pour chercher les qutre points de l'enveloppe qui se trouve aussi dans le rectangle. Après je peux calculer les sommets de rectangle en utilisant ces quatre points. La complexité théorique est $\mathcal{O}(n+r)$ où r est la complexité de la marche de Jarvis.

1.2 Définitions, notations, stuctures de donnés utilisée

1. $\overrightarrow{ab}$ est représente la vecteur depuis a vers b, $|\overrightarrow{ab}|$ est le scalaire de $\overrightarrow{ab}$.
2. $\overrightarrow{ab} \cdot \overrightarrow{cd}$ dénote le produit scalaire entre $\vec{ab}$ et $\vec{cd}$.
3. $\overrightarrow{ab} \times \overrightarrow{cd}$ dénote le produit vectoriel entre $\overrightarrow{ab}$ et $\overrightarrow{cd}$.

On dénote l'ensembles de points par $\mathcal{P}$, le nombre de points de $\mathcal{P}$ par n.
Une enveloppe convexe E de $\mathcal{P}$ est un sous-ensemble E de $\mathcal{P}$ en ordre du sens horaire tel que le polygon composé par tous les points de E peut entourner tous les points de $\mathcal{P}$.
Un rectangle convexe Rec est une liste de qutre points. $RecMin$ est le rectangle minimun convexe. AR est une ensemble de quatre arêtes de Rec.

Lemma 1 if Rec est $RecMin$, alors $\exists \mathcal{B}$, $\mathcal{B} \in$ E tel que $\forall b, b \in \mathcal{B}, \exists c, ar, c \in Rec, ar \in AR$, $\overrightarrow{bc}$ sont colinéaire avec ar.
Lemma 2 il existe une arête α de $RecMin$, α passe une arête du polygon minimum convexe.

1.3 Présentation de l'algorithm de Toussaint

Afin de calculer les sommets A, B, C, D de rectangle, on a besoin d'abord de trouver α qui est colinéaire avec une arête de rectangle et les points A', B', C', D' de $\mathcal{B}$ qui est sur les arêtes de $RecMin$.

1.3.1 Etape 1 calculer A', B', C', D'

Pour A', on peut énumérer chaque points i de E, $A' = E_i$, et $\alpha = \overrightarrow{E_{i+1}E_i}$

Pour B', il est le plus ouest point tel que $|\alpha \cdot \overrightarrow{B'A'}|$ est maximum, il exprime $\angle\alpha\overrightarrow{B'A'}$ est le plus grand. Comme E est en ordre du sens horaire, on énumère chaque point E_r de E, $|\alpha \cdot \overrightarrow{E_rA'}| \leqslant |\alpha \cdot \overrightarrow{E_{r+1}A'}|$, si une fois $|\alpha \cdot \overrightarrow{E_rA'}| > |\alpha \cdot \overrightarrow{E_{r+1}A'}|$, alors $B' = E_r$

Pour C', il est le point antipodal de A', la distance entre A' et C' est plus grande que A' et les autres points, donc $|\alpha \times \overrightarrow{E_uA'}|$ est maximum. Comme E est en ordre du sens horaire, si on énumère chaque point E_u de E, au début $|\alpha \times \overrightarrow{E_uA'}| \leqslant |\alpha \times \overrightarrow{E_{u+1}A'}|$, si une fois $|\alpha \times \overrightarrow{E_uA'}| > |\alpha \times \overrightarrow{E_{u+1}A'}|$, alors $C' = E_u$

Le dernier point D' est le plus est point, $\angle\alpha D'$ est le plus petit. On énumère chaque point E_l de E, comme E est en ordre du sens horaire, au début $|\alpha \cdot \overrightarrow{E_lA'}| \geq |\alpha \cdot \overrightarrow{E_{l+1}A'}|$, si une fois $|\alpha \cdot \overrightarrow{E_lA'}| < |\alpha \cdot \overrightarrow{E_{l+1}A'}|$, alors $D' = E_l$

1.3.2 Etape 2 calculer l'aire $\mathcal{S}$ de rectangle

Pour la hauteur de rectangle h, on suppose que le point C'_α est le projecteur de C' sur α, h $= |\overrightarrow{C'_\alpha C'}|$
$\mathcal{S}_{\text{parallélogramme}} = |\vec{\alpha} \times \overrightarrow{C'A'}|$.
h $= \frac{\mathcal{S}_{\text{parallélogramme}}}{|\vec{\alpha}|}$

Pour la largeur de rectangle w, on suppose que B'_α est le projecteur de B' sur α et D'_α est le projecteur de D' sur α, w $= |\overrightarrow{B'_\alpha D'_\alpha}|$
$\overrightarrow{B'_\alpha A'} = \frac{\alpha \cdot \overrightarrow{B'A'}}{|\alpha|}$
$\overrightarrow{D'_\alpha A'} = \frac{\alpha \cdot \overrightarrow{D'A'}}{|\alpha|}$
$|\overrightarrow{B'_\alpha D'_\alpha}| = |\overrightarrow{D'_\alpha A'} - \overrightarrow{B'_\alpha A'}|$
Donc, $\mathcal{S} = h \times w$. Si $\mathcal{S} < \mathcal{S}_{RecMin}$, on va continue à l'étape 3 afin de construire le nouveau $RecMin$ sinon on essaie le A' suivant.

1.3.3 Etape 3 calculer A, B, C, D

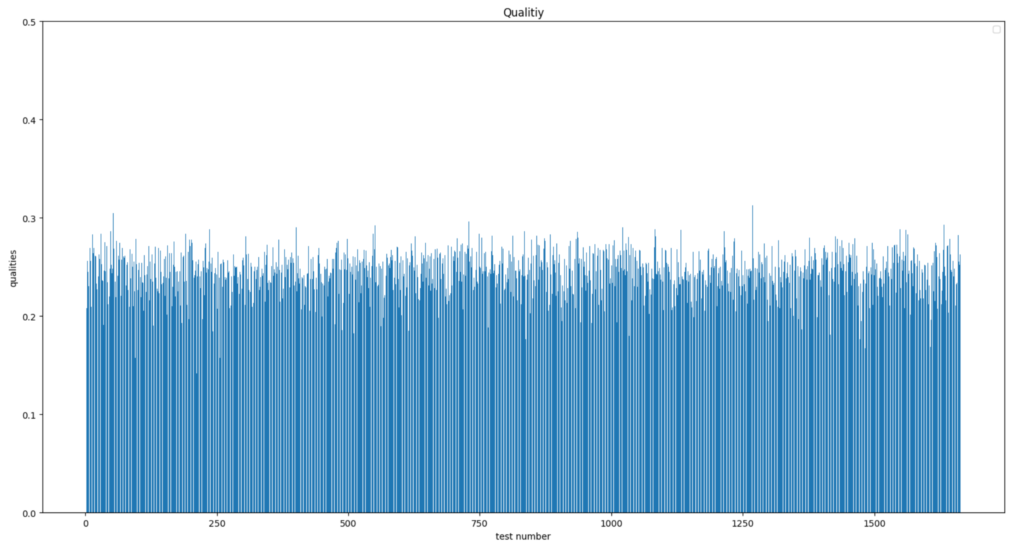
Dès que'on obtient A', B', C', D', α, $\overrightarrow{B'_\alpha A'}$, $\overrightarrow{B'_\alpha D'_\alpha}$, $\overrightarrow{C'_\alpha C'}$, on peut facilement calculer A, B, C, D.
A est le point B'_α, $B'_\alpha = A' - \alpha \cdot \frac{\overrightarrow{B'_\alpha A'}}{|\alpha|}$
B est le point suivant de A au sens de horaire, $B = B'_\alpha + \overrightarrow{B'_\alpha B'} \cdot \frac{\overrightarrow{C'_\alpha C'}}{|\overrightarrow{B'_\alpha B'}|}$
C est le point suivant de B au sens de horaire, $C = B + \overrightarrow{B'_\alpha A'} \cdot \frac{\overrightarrow{B'_\alpha D'_\alpha}}{|B'_\alpha A\prime|}$
Le dernier point $D = A - B = C + \overrightarrow{BB'_\alpha}$

Algorithm 1 : pseudocode de l'algorithm de Toussaint

```
𝒫 ← Ensemble de points ;
E ← Jarvis(𝒫) ;
u ← 1 ;
l ← 1 ;
r ← 1 ;
minAire ← ∞ ;
for each arête →(E_{i+1}E_i), 0 ⩽ i < |E| do
    A' ← E_i ;
    α ← →(E_{i+1}E_i) ;
    while |α × →(E_uE_i)| ⩽ |α × →(E_{u+1}E_i)| do
        u ← (u + 1) mod |E| ;
    end while
    C' ← E_u ;
    while |α · →(E_rE_i)| ⩽ |α · →(E_{r+1}E_i)| do
        r ← (r + 1) mod |E| ;
    end while
    B' ← E_r ;
    if i == 0 then
        l = r ;
    end if
    while |α · →(E_lE_i)| ⩽ |α · →(E_{l+1}E_i)| do
        l ← (l + 1) mod |E| ;
    end while
    D' ← E_l ;
    →(B'_α A') ← (α · →(B'A')) / |α| ;
    →(D'_α) ← (α · →(D'A')) / |α| ;
    height ← |C'_α C'| ← (α × →(C'A')) / |α| ;
    width ← →(B'_α A') − →(D'_α A') ;
    aire ← width × height ;
    if aire < minAire then
        minAire ← aire ;
        A ← A' − α · →(B'_α A') / |α| ;
        B ← B'_α + →(B'_α B') · →(C'_α C') / |→(B'_α B')| ;
        C ← B + →(B'_α A') · →(B'_α D'_α) / |B'_α A'| ;
        D ← C + (A − B) ;
    end if
end for
return A, B, C, D ;
```

1.4 Résultat du test

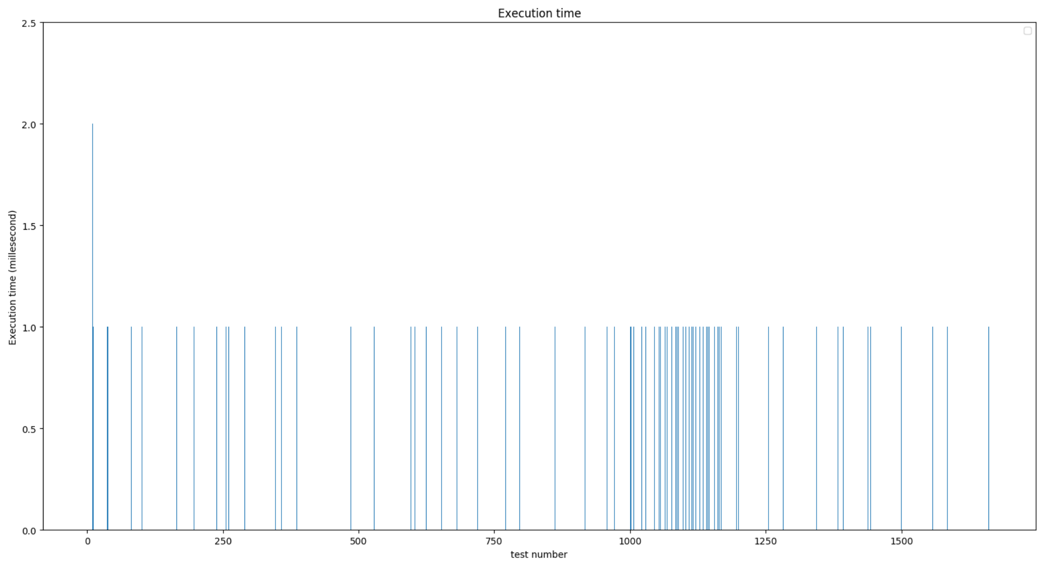
Le graphe dessous est le résultat de qualité associé à chaque teste de VAROUMAS.

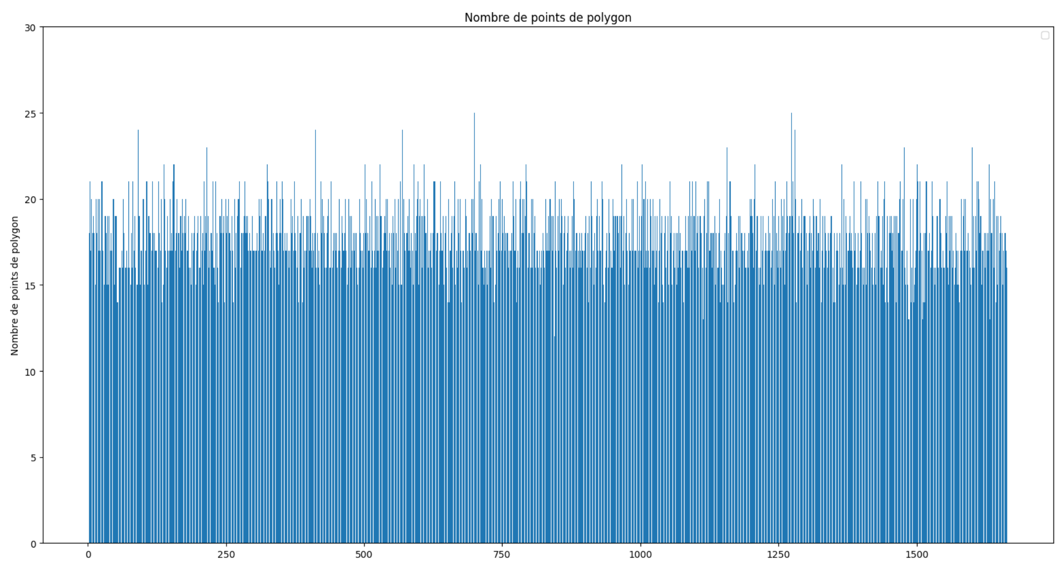


$\text{Qualité}_{rectangle} = \frac{\text{aire rectangle}}{\text{aire polygone}} - 100\%$
L'abscisse est représente le numéro de test et l'ordonnée est le qualité de rectangle minimum convexe. On peut voir que l'aire du rectangle minimum convexe est toujours plus grand que l'aire de polygone convexe.

1.5 Performance de l'algorithme

Le graphe dessous est le temps d'exécution pour chaque teste de VAROUMAS.





Pour chaque cas de test, il y a 255 points aléatoire, après calculer l'enveloppe convexe en utilisant Jarvis, il reste seulement h points de l'enveloppe convexe $12 \leqslant h \leqslant 25$ dans tous les tests. En fonction de la complexité théorique de l'algorithme $\mathcal{C} = \mathcal{O}(nh) + \mathcal{O}(h)$. Donc le temps d'exécution n'est pas très évident. Pour le plupart de tests, le temps d'exécution t, $0 < t <= 1ms$.

2 Algorithme Ritter