Examen DEA Images et Vision 2003-2004 : géométrie algorithmique
3h — 9 mars 2004

1 Recherche de plus proche voisin

Étant donné un ensemble $\mathcal{S}$ de n points dans le plan et un point du plan p on cherche à déterminer $NN_{\mathcal{S}}(p)$ le point de $\mathcal{S}$ le plus proche de p. On note $DT(\mathcal{S})$ la triangulation de Delaunay de $\mathcal{S}$.

w est initialisée avec un sommet quelconque de $DT(\mathcal{S})$.

```
N = un sommet de DT(S);
dmin = |pN|;
Faire {
        v = N;
        Pour chaque voisin w de v dans DT(S) {
                si ( dmin > |pw| ) {
                dmin = |pw|;
                N = w;
                }
        }
} tant que (N ≠ v);
return N;
```

1.1

Démontrer que ce programme calcule le plus proche voisin de v.

1.2

Quelle est la complexité dans le cas le pire de cet algorithme ? Donner un exemple réalisant cette pire complexité ($\forall n$).

1.3

Si on modifie l'algorithme comme suit:

Variante :

```
N = un sommet de DT(S);
dmin = |pN|;
Faire {
        v = N;
        Pour chaque voisin w de v dans DT(S) {
                si ( dmin > |pw| ) {
                dmin = |pw|;
                N = w;
                break; // Sortir de la boucle "Pour"
                }
        }
} tant que (N ≠ v);
return N;
```

Qu'apportent la variante?

Le nombre d'exécutions de l'instruction $N = w$ est-il modifié?

Dans quel sens?

Est-ce que l'on calcule bien toujours le plus proche voisin?

Commentaires?

2 Triangulation optimisant les cercles circonscrits

Étant donné un ensemble de n points du plan $\mathcal{S}$ et une triangulation $\mathcal{T}(\mathcal{S})$ de cet ensemble de points, on définit $\mu(\mathcal{T}(\mathcal{S}))$ le maximum des rayons des cercles circonscrits aux triangles de $\mathcal{T}(\mathcal{S})$.

2.1

Trouver une triangulation qui minimise $\mu(\mathcal{T}(\mathcal{S}))$. Algorithme, complexité...

2.2

Trouver une triangulation qui maximise $\mu(\mathcal{T}(\mathcal{S}))$ **dans le cas particulier** où les points de $\mathcal{S}$ forment un polygone convexe. Algorithme, complexité...

3 Plus petit carré englobant.

Étant donné un ensemble de n points du plan $\mathcal{S}$ on cherche le plus petit carré contenant ces n points. On supposera que l'on n'a pas trois points alignés.

3.1

Pour commencer on cherche un carré dont les cotés sont parallèles à une certaines direction (repéré par son angle θ avec l'axe des x). Donner un algorithme et sa complexité. En général, combien de points de $\mathcal{S}$ sont-ils sur le bord de ce carré.

3.2

A quel moments les points définissant le plus petit carré peuvent-ils changer lorsque θ augmente ?

3.3

On appelle ces valeurs de θ valeurs critiques.

Entre deux valeurs critiques comment varie la taille du carré ?

3.4

Lorsque l'on se situe à un certain θ comment déterminer la prochaine valeur critique.

3.5

Déduire un algorithme, donner sa complexité.

3.6

Et si on voulait trouver le rectangle englobant d'aire minimale ?