

## Übungsblatt 10 – Quadtrees

**Ausgabe:** 19. Juni 2012

**Abgabe:** 26. Juni 2012

### 1 Delaunay Triangulierungen - Meshing

Der in der Vorlesung vorgestellte Meshing-Algorithmus erzeugt nur nicht-stumpfe Dreiecke, d.h. Dreiecke die keinen Winkel haben der größer als  $90^\circ$  ist. Sei  $\mathcal{T}$  eine Triangulierung einer endlichen Punktmenge  $P \subset \mathbb{R}^2$  welche nur nicht-stumpfe Dreiecke enthält. Zeige, dass  $\mathcal{T}$  eine Delaunay Triangulierung von  $P$  ist.

### 2 Komprimierte Quadtrees

In dieser Aufgabe geht es um die Verringerung des Speicheraufwands für Quadtrees. Seien dazu eine endlichen Menge  $P \subset \mathbb{R}^2$  von  $n$  Punkten sowie ein Quadtree  $\mathcal{Q}$  mit Tiefe  $d$  auf  $P$  gegeben. Ein gewöhnlicher Quadtree hat Speicheraufwand  $\mathcal{O}((d+1)n)$ . Es ist möglich diesen Aufwand auf  $\mathcal{O}(n)$  zu reduzieren in dem man jeden Knoten  $v$  entfernt der nur einen Kinds-Knoten hat in dem sich Knoten befinden. Dabei ändert man den Pointer vom Eltern-Knoten von  $v$  auf  $v$  so ab, dass er auf den einzig interessanten Kinds-Knoten von  $v$  zeigt.

- Beweise, dass das beschriebene Verfahren tatsächlich den Speicheraufwand auf  $\mathcal{O}(n)$  reduziert.
- Wie ist die Laufzeit des Verfahrens, wenn man es auf einen Quadtree anwendet?
- Ist es auch möglich den Aufwand von  $\mathcal{O}((d+1)n)$  für die Konstruktion eines solchen Quadtrees zu verringern?

*Bitte wenden*

### 3 Balancierung

In der Vorlesung wurde als Balancierungs-Bedingung für Quadrees gefordert, dass adjazente Rechtecke sich höchstens um einen Faktor von zwei in ihrer Größe unterscheiden dürfen. Wir verschärfen nun diese Bedingung und fordern, dass alle Rechtecke exakt gleich groß sind. Wandeln wir nun einen Quadtree mit  $m$  Knoten in einen 'balancierten' Quadtree  $\mathcal{Q}$ , der obiger Bedingung genügt, um. ist die Anzahl der Knoten in  $\mathcal{Q}$  dann immer noch linear in der Anzahl der Knoten des original Quadrees  $[O(m)]$ ? Wenn das nicht der Fall ist kann man eine obere Schranke für die Anzahl der Knoten angeben?

### 4 Range Queries mit Quadrees

Man kann Quadrees nutzen um Range Queries durchzuführen. Beschreibe einen möglichst effizienten Algorithmus der einen Quadtree  $\mathcal{Q}$  auf einer Punktmenge  $P$  nutzt um eine Query für eine Anfrage-Region  $R$  durchzuführen. Dabei sei der Quadtree  $\mathcal{Q}$  bereits gegeben.

- a) Analysiere die worst-case Laufzeit für den Fall, dass  $R$  ein Rechteck mit Kanten Achsenparallelen Kanten ist.
- b) Wie ändert sich die worst-case Laufzeit, wenn  $R$  eine Halbebene ist, die durch eine vertikale Gerade begrenzt ist.