

Managing Massive Multiplayer Online Games  
 SS 2016

Übungsblatt 2: Raum

Besprechung: 28.04.2015

Betrachten Sie im Folgenden ein abstraktes Spiel, bei dem sich die Spieler in einer zweidimensionalen Welt bewegen. Ein Spieler kann in dieser Welt Objekte und andere Spieler innerhalb einer kreisförmigen Region mit Radius  $s$  (*Sichtradius*) wahrnehmen.

**Aufgabe 2-1**     *Sichtbarkeit*

- (a) Wie lässt sich effizient überprüfen, ob ein Spieler  $S$  mit Position  $(S.x, S.y)$  ein kreisförmiges Objekt  $o$  mit Mittelpunkt  $o_m$  und Radius  $o_r$  sehen kann?
- (b) Sei  $R$  eine rechteckige Approximation einer Menge  $M$  von Spielern. Wie lässt sich effizient überprüfen, ob die Möglichkeit besteht, dass ein Spieler  $S \notin M$  Spieler aus  $M$  sehen kann, ohne auf die exakten Positionen der Spieler in  $M$  zuzugreifen?
- (c) Seien  $R_1$  und  $R_2$  rechteckige Approximation disjunkter Mengen  $M_1$  und  $M_2$  von Spielern. Wie lässt sich effizient überprüfen, ob die Möglichkeit besteht, dass sich Paare von Spielern  $(s_1, s_2) \in M_1 \times M_2$  sehen können, ohne auf die exakten Positionen der Spieler in  $M_1$  und  $M_2$  zuzugreifen?

Hinweis: Folgende Funktionen können im Folgenden verwendet werden:

- Die euklidische Distanz zwischen zwei Punkten  $p_1$  und  $p_2$ :

$$Dist(p_1, p_2) = \sqrt{(p_1.x - p_2.x)^2 + (p_1.y - p_2.y)^2}$$

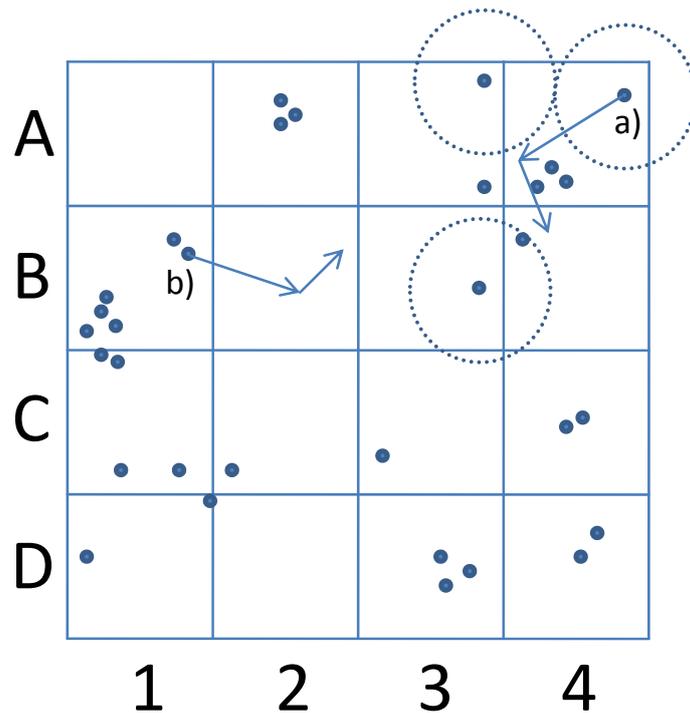
- Die minimale euklidische Distanz zwischen einem Punkt  $p$  und einem Rechteck  $R$ .

$$MinDist(p, R) = \sqrt{\sum_{i=1}^2 \begin{cases} |R_i^{min} - p_i|^2, & \text{if } R_i^{min} > p_i \\ |p_i - R_i^{max}|^2, & \text{if } p_i > R_i^{max} \\ 0, & \text{else} \end{cases}} \quad (1)$$

- Die minimale euklidische Distanz zwischen zwei Rechtecken  $A$  und  $B$ .

$$MinDist(A, B) = \sqrt{\sum_{i=1}^2 \begin{cases} |A_i^{min} - B_i^{max}|^2, & \text{if } A_i^{min} > B_i^{max} \\ |B_i^{min} - A_i^{max}|^2, & \text{if } B_i^{min} > A_i^{max} \\ 0, & \text{else} \end{cases}} \quad (2)$$

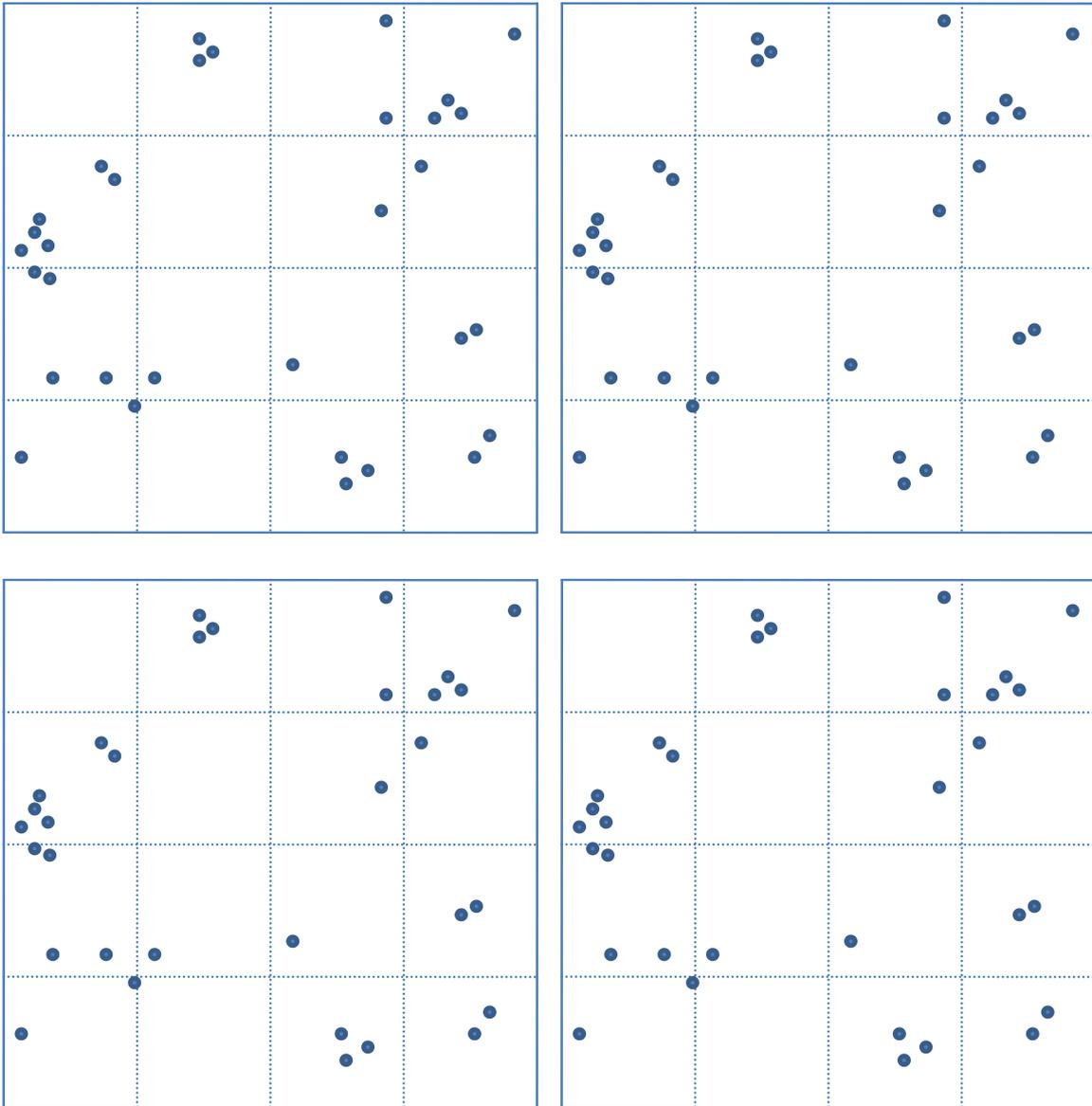
wobei  $A_i$  die Projektion von  $A$  auf Dimension  $i$  (etwa der  $i$ -te Eintrag eines Vektors oder das Intervall, das die Ausdehnung eines Rechtecks in der  $i$ -ten Dimension beschreibt) und  $X^{min}$  ( $X^{max}$ ) das Minimum (Maximum) eines Intervalls  $X$  bezeichnet.



**Aufgabe 2-2** *Spatial-Publish-Subscribe*

Im Folgenden sei das Spielfeld gleichmäßig in ein  $4 \times 4$  Grid von quadratischen Micro-Zones aufgeteilt. Auf diesem Spielfeld bewegen sich 32 Objekte mit wie oben gezeigten initialen Positionen. Die Area of Interest (AoI) eines Objekts soll kreisförmig sein, wobei der Radius genau der halben Seitenlänge einer Gridzelle entspricht. Obige Abbildung zeigt das Spielfeld, die AoI ist für einige Objekte exemplarisch eingezeichnet

- Betrachten Sie das mit *a*) markierte Objekt in Zelle A4. Bei welchen Micro-Zones ist dieses Objekt zunächst subscribed?
- Von welchen Objekten bekommt *a*) zunächst Positionsinformationen gesendet? Welche Objekte erhalten zunächst Positionsinformationen von *a*) gesendet?
- Bei welchen Micro-Zones muss sich *a*) subscriben (unsubscribe), wenn es sich wie im Bild gezeigt bewegt?
- Bei welchen Micro-Zones muss sich das mit *b*) markierte Objekt in Zelle B1 subscriben (unsubscribe), wenn es sich wie im Bild gezeigt bewegt?



**Aufgabe 2-3** *Räumliche Indexstrukturen*

Im Folgenden soll der oben gegebene Datensatz indexiert werden. Verwenden Sie dazu:

- (a) Einen Quadtree mit zwei Objekten Seitenkapazität. Für die ersten Splits sind Hilfslinien vorgegeben.
- (b) Einen kD-Trees mit vier Objekten Seitenkapazität. Beginnen Sie mit einem Split der x-Achse.
- (c) Einen R-Baum mit einer Seitenkapazität von zwei Objekten. Erstellen Sie den R-Baum mit Hilfe des Sort-Tile Recursive Algorithmus.