CMSC 754 - Computational Geometry
Lecture 14: Well-Separated Pair Decompositions (I)

Geometric Approximations:

- Useful when exact computation is too costly
- Geometric inputs are measurements
  - Meaningless to compute a result more accurate than input uncertainty

Examples:

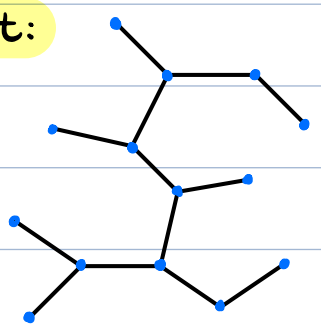
Euclidean Min. Spanning Tree (EMST)

- $\mathbb{R}^2$ - Exact solution in $O(n \log n)$ time
- $\mathbb{R}^d$ - Best exact solution $O(n^{2-4/d})$ (Brute force: $O(n^2 \log n)$)

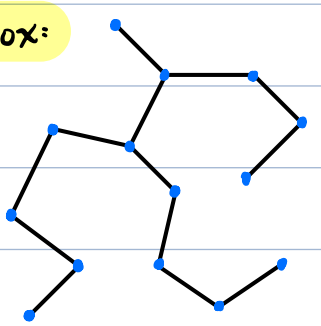
Not much better as d increases

Approximation: Given $\mathbb{P}$ + $\varepsilon > 0$, compute any spanning tree of weight $\leq (1+\varepsilon)$ EMST($\mathbb{P}$)

Exact:



Approx:

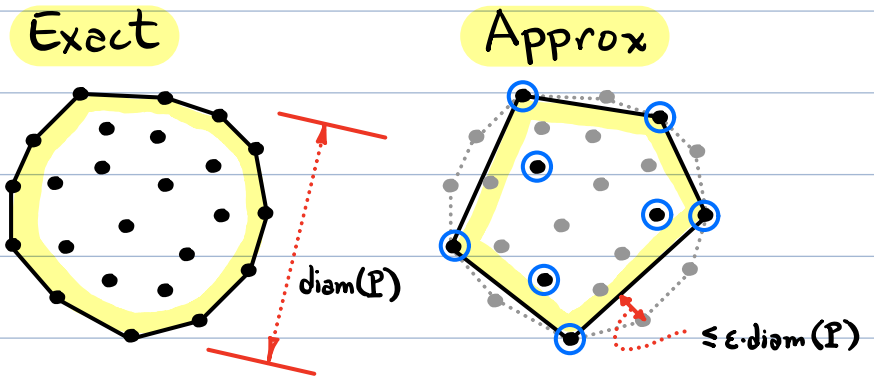


Convex Hull: Given $P = \{p_1, \ldots, p_n\}$ in $\mathbb{R}^d$

- $\mathbb{R}^2$: $O(n \log n)$ exact
- $\mathbb{R}^d$: $O(n^{d/2})$ exact

Approximation: Compute subset $P' \subseteq P$ s.t.

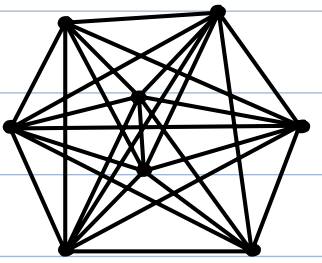
$\text{dist}(\text{conv}(P), \text{conv}(P')) \leq \varepsilon \cdot \text{diam}(P)$



Euclidean Graph: Given $P = \{p_1, \ldots, p_n\}$ in $\mathbb{R}^d$

this is a weighted complete graph ($\binom{n}{2}$ edges)

where $w(p_i, p_j) = \|p_i - p_j\|$



Approximation: Given P & $\varepsilon > 0$,

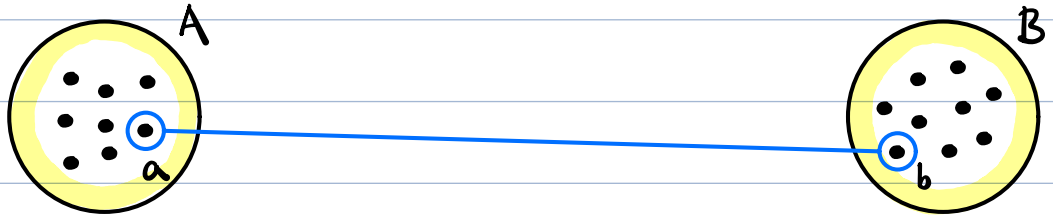
can we encode all edges with

$O(n)$ pairs, so distances preserved

within factor $(1 \pm \varepsilon)$?

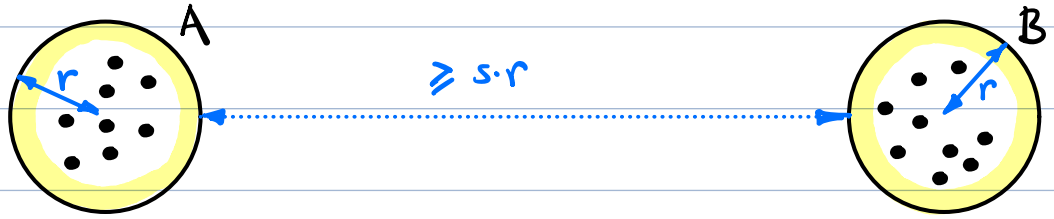
→ Well-separated pair decomposition (WSPD)

Intuition: If two sets A + B are well separated from each other, the distances of $A \times B$ can be well approximated by any pair $a \in A$, $b \in B$.



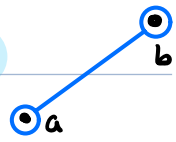
Idea: Represent all $\binom{n}{2} = O(n^2)$ pairs with just $O(n)$ well-separated pairs

Def: Given $P = \{p_1, \ldots, p_n\}$ in $\mathbb{R}^d$ + $s > 0$
Sets $A, B \subseteq P$ are s-well separated if A + B can be enclosed in balls of some radius r, s.t. these balls are separated by distance $\geq s \cdot r$



Obs:

- A, B are s-well sep. $\Rightarrow$ they are s'-well sep. for $s' \leq s$
- Individual points are s-well sep. for any $s > 0$



Def: Given sets A & B, set of unordered pairs:

$$A \otimes B = \{\{a,b\} : a \in A,\ b \in B,\ a \neq b\}$$

$P \otimes P$ = set of $\binom{n}{2}$ 2-element subsets of P

Def: Given $P \subseteq \mathbb{R}^d$ & $s > 0$, an **s-well separated pair decomposition** (s-WSPD) is collection of pairs:

$$\{\{A_1, B_1\}, \ldots, \{A_m, B_m\}\} \quad \text{where } A_i, B_i \subseteq P$$

such that:

(i) A_i & B_i are s-well separated, $1 \leq i \leq m$

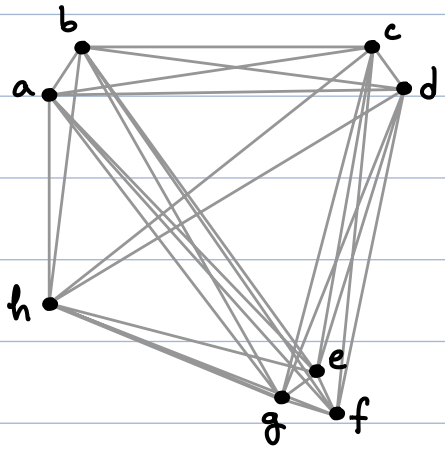
[$\Rightarrow$ A_i & B_i are disjoint]

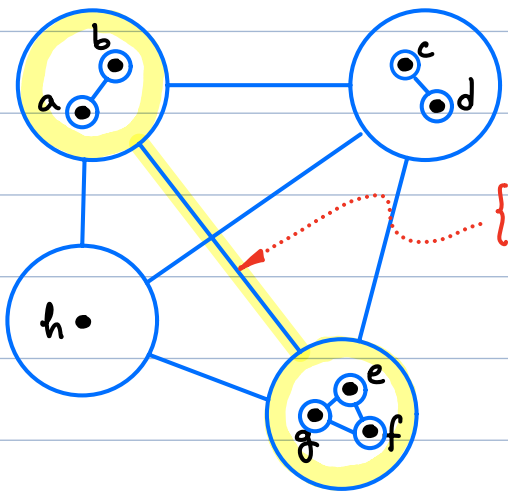
(ii) $\bigcup_{i=1}^{m} A_i \otimes B_i = P \otimes P$

[$\Rightarrow$ all pairs are covered]

Our construction will have the property that every pair is covered just once

28 pairs





11 well-separated pairs

$\{\{a,b\},\{e,f,g\}\}$

Does such a decomposition exist?

Trivial: For any $s > 0$, there exists an s-WSPD consisting of $\binom{n}{2}$ singleton pairs.

But can we do this with just $O(n)$ pairs?

Yes! $O(s^d \cdot n)$ pairs in $\mathbb{R}^d$

Computable in $O(n \log n + s^d n)$ time

Two steps:

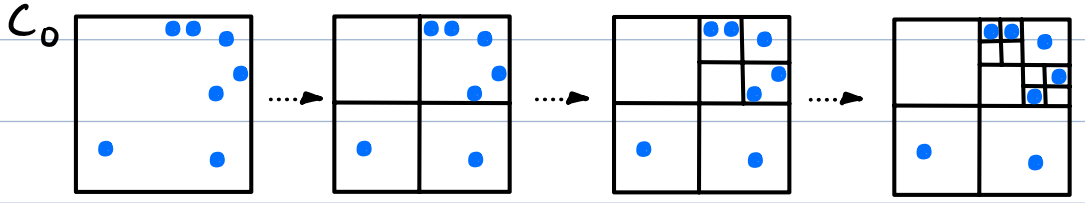
1. Compute a (compressed) quadtree for P in $O(n \log n)$ time
2. Generate WSPs by recursive refinement in $O(s^d n)$ time

Quadtrees: A hierarchical spatial partition of an n-element point set P in $\mathbb{R}^d$

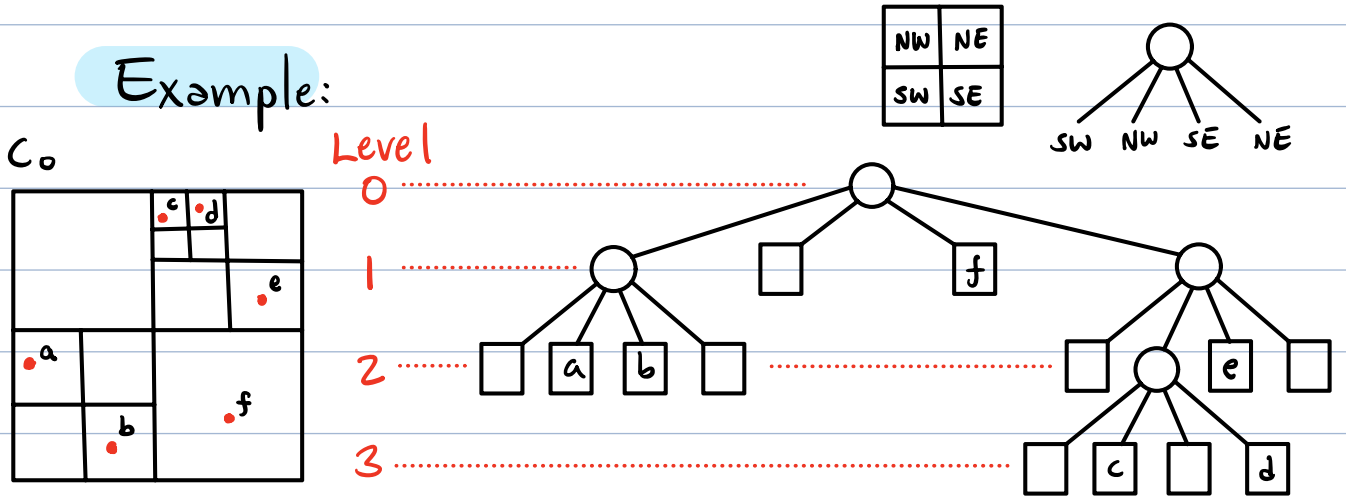
- Subdivides space into cubical cells

Simple Construction:

- Let C_0 be a cube containing P
- While a cell contains 2 or more points, split into 2^d cubes of half side length → children

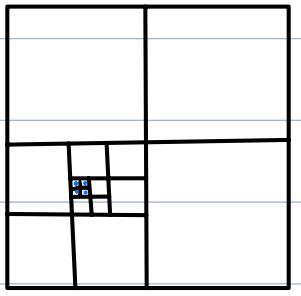


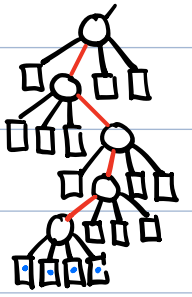
Example:



Problem with simple construction:

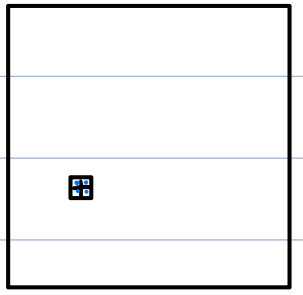
If points are densely clustered there may be many fruitless splits





Fix: Path compression

- Remove sequences of fruitless splits
- Compress tree path to single edge

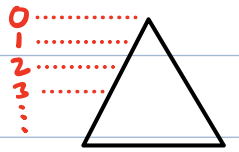




$\Rightarrow$ Compressed Quadtree

Theorem: Given a set $P = \{p_1, \ldots, p_n\}$ in $\mathbb{R}^d$, can construct a compressed quadtree with space $O(n)$ in $O(n \log n)$ time.

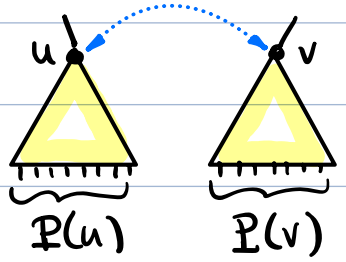
Notation: For each node:

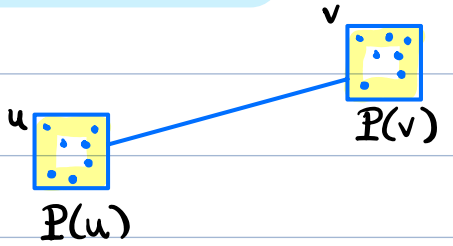


- level(u) = distance of u from root
- P(u) = points in u's subtree
- rep(u) = representative point from P(u) (chosen arbitrarily)

> **Lemma:** levels + representatives can be computed in $O(n)$ time.

WSPs - Each well-separated pair is represented by a pair of nodes

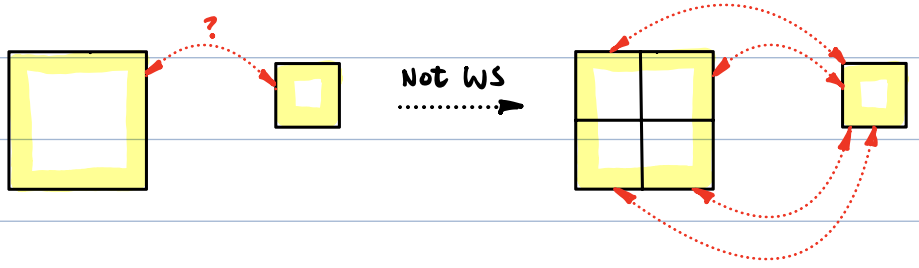




WSP: $P(u) \otimes P(v)$

Recursive Construction:

- Starting at root we try all pairs of nodes
- If current pair is not well separated, split larger node (i.e., lower level num.) + recurse on children

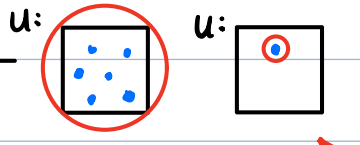


Assume quadtree constructed

- Let u_0 be root
- Invoke recursive procedure

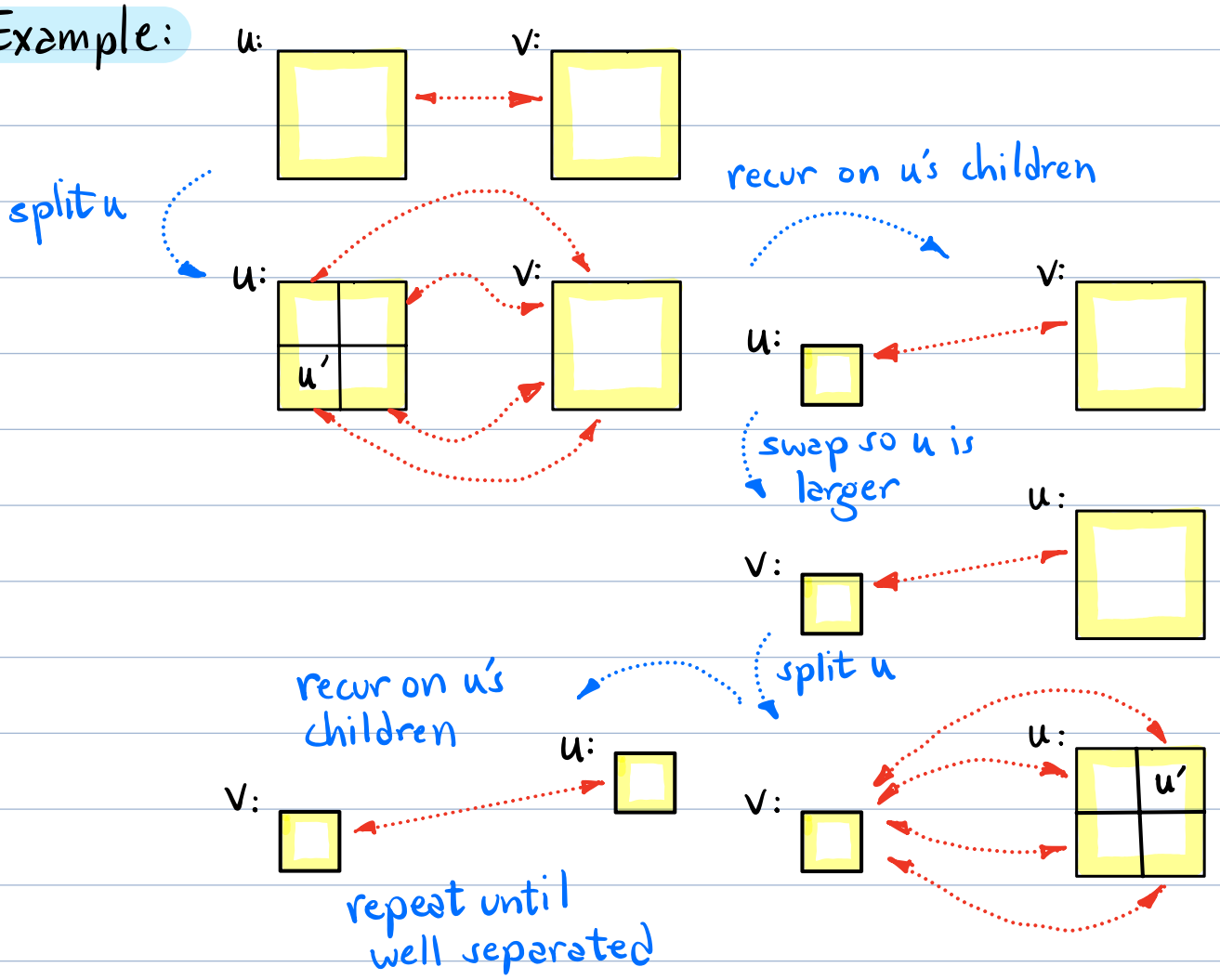
ws-pairs(u_0, u_0, s)

```
ws-pairs ( Node u, Node v, Scalar s )
    if ( u + v are both leaves + u = v ) return ∅
    if ( either u or v is empty leaf ) return ∅
    if ( well-separated (u,v,s) ) // (see below)
        return {u,v}  // found a pair
    if ( level(u) > level(v) )
        swap u ↔ v  // swap so u is larger
    wsp ← ∅
    for each child u' of u  // recurse on children
        wsp ← wsp ∪ ws-pairs (u', v, s)
    return wsp  // return union of results
```




```
well-separated (u,v,s)
    ball_u ← if (u is internal) enclosing ball for u's cell
             else ball of radius 0 centered at rep(u)
    ball_v ← (same)
    return ball_u + ball_v are s-well separated
```

Example:



Analysis:

> **Lemma:** Given n points in $\mathbb{R}^d$, the construction produces $O(s^d \cdot n)$ pairs, for $s \geq 1, d = O(1)$.

(Simplified: Assume no path compression needed)

Terminal / Non-Terminal:

- A call ws-pair(u, v, s) is terminal if it makes no recursive calls

- ≤ 1 WSP generated by each terminal cell
- Since each non-terminal call makes $\leq 2^d$ recursive calls:

$\Rightarrow$ num. of WSPs $\leq$ num. terminal calls
$\leq 2^d \cdot$ num. non-term calls
$= O(\text{num. non-term calls})$
since $d = O(1)$ $2^d = O(1)$

$\Rightarrow$ It suffices to show $\leq s^d \cdot n$ non-term calls

Charging Argument:

- We'll charge each non-term call ws-pair(u, v, s) to the node with smaller cell (i.e. larger level)
- We'll show each node receives $O(s^d)$ charges
- Quadtree has $O(n)$ nodes

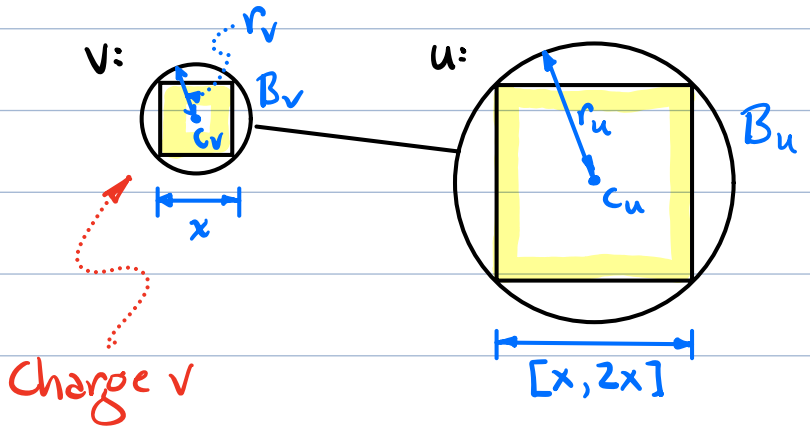
$\Rightarrow O(s^d \cdot n)$ charges $\Rightarrow O(s^d \cdot n)$ total WSPs.

Consider any non-term call ws-pair(u, v, s)

Non-term $\Rightarrow$ not s-well separated

Let: B_u, B_v - enclosing balls of cells
c_u, c_v - centers of enclosing balls
r_u, r_v - radii of enclosing balls
x - side length of v's cell
$\Rightarrow$ side length of u's cell $\in [x, 2x]$

Cell sizes are similar
Always split larger

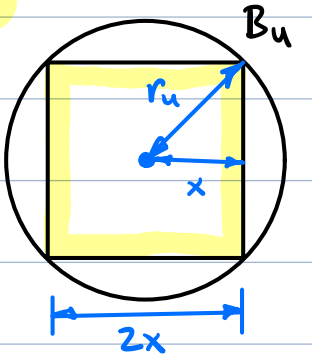


Because $u \not\sim v$ not s-well separated,

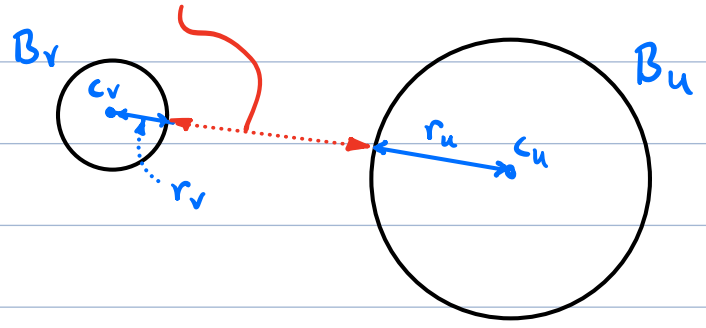
$$\text{dist}(B_u, B_v) < s \cdot \max(r_u, r_v)$$

$$= s \cdot r_u$$

$$\leq s \cdot x \cdot \sqrt{d}$$



Obs: $\text{dist}(B_u, B_v) = \|c_u - c_v\| - r_u - r_v$



$$\Rightarrow \|c_u - c_v\| \leq r_u + r_v + s \times \sqrt{d}$$

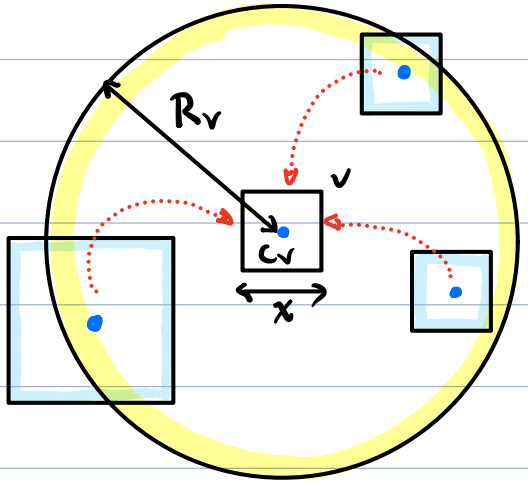
$$= x\sqrt{d} + \frac{x}{2}\sqrt{d} + s x \sqrt{d} = \left(1 + \frac{1}{2} + s\right) x \sqrt{d}$$

$$\leq 3s\, x \sqrt{d} \quad (\text{since } s \geq 1)$$

Summary: A node (v) of side length x is charged only by nodes (u) of side lengths in $[x, 2x]$ whose centers lie within a ball of radius $R_v = 3s x \sqrt{d}$.

How many such nodes can there be?

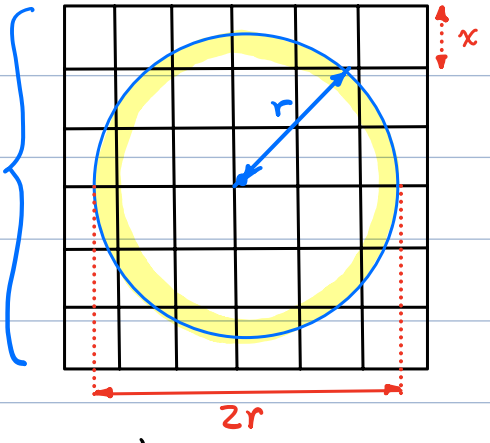




Packing Lemma: The number of disjoint quadtree cells of side length $\geq x$ whose centers are contained in ball of radius r is

$$\leq \left(1+\left\lceil \frac{2r}{x} \right\rceil\right)^d \leq O\left(\max\left(2, r/x\right)^d\right)$$

Proof (by picture)



So, total num. of charges per node is

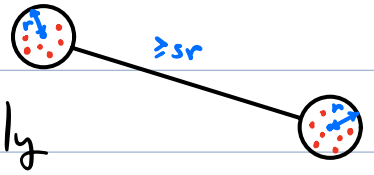
$$\leq \max\left(2, (R_v/x)^d\right)$$
$$\leq \max\left(2, (3s \times \sqrt{d}/x)^d\right)$$
$$\leq (3s\sqrt{d})^d = O(s^d) \quad [d \text{ is } O(1)]$$

Num. WSPs $\leq$ (num. nodes) $\cdot O(s^d) = O(s^d \cdot n)$ □

Theorem: Given an n-element point set P in $\mathbb{R}^d$ ($d = O(1)$) and $s \geq 1$, we can construct an s-WSPD for P of size $O(s^d n)$ in time $O(n \log n + s^d n)$.

Summary:

- Geometric approximations
- Well-separated pairs
  - Encoding many pairs concisely



- WSPD - Encodes all $\binom{n}{2}$ pairs using $O(n)$ well-separated pairs
- Can construct an s-WSPD for n points in $\mathbb{R}^d$ of size $O(s^d \cdot n)$ in time $O(n \log n + s^d \cdot n)$
  - Compressed quadtree - $O(n \log n)$
  - Recursive construction - $O(s^d \cdot n)$

Coming up?

- Applications