Approximate Voronoi Diagrams

David M. Mount

University of Maryland

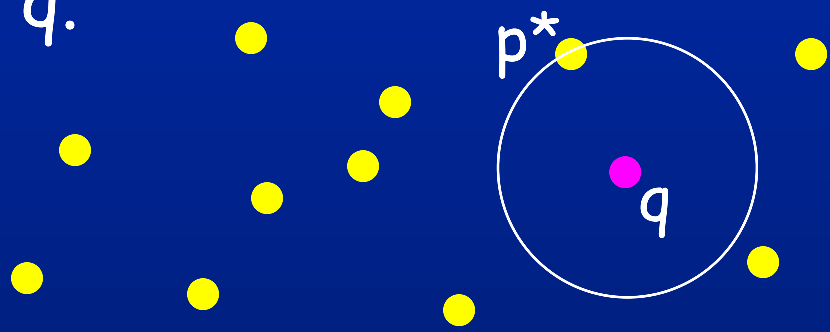
Joint work with:

Sunil Arya (Hong Kong U. of Sci. and Tech)

Charis Malamatos (Max Plank Inst.)

Nearest Neighbor Searching

Nearest Neighbor: Given a point set $S \subseteq R^d$ and $q \in R^d$, find the point $p^* \in S$ that is closest to q.



NN Queries: Preprocess S so that nearest neighbors can be computed efficiently.

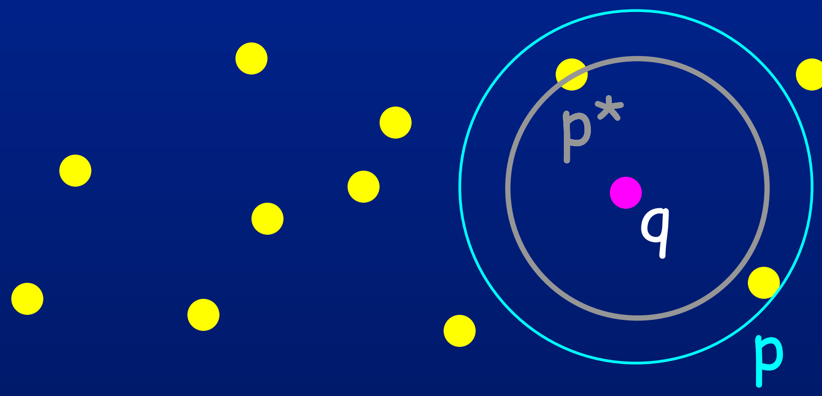
Curse of Dimensionality: Exp growth in d.

Approximation

ε-Nearest Neighbor: Given $\varepsilon > 0$ and $q \in R^d$, a point $p \in S$ is an ε-nearest neighbor of q if,

$$dist(q,p) \leq (1+\varepsilon) dist(q,p^*),$$

where $p^* \in S$ is the nearest neighbor of q.



Approximate NN Searching

Low dimensional Approaches: (n big, d=O(1)).
Near linear space, exponential in dimension.

	Query Time	Space
Arya, et al '98	$(1/\varepsilon)^d \log n$	n
Clarkson '97 Chan '98	$(1/\varepsilon)^{\frac{d-1}{2}} \log n$	$(1/\varepsilon)^{\frac{d-1}{2}} n\log n$

Why $(1/\varepsilon)^{\frac{d-1}{2}}$ **?** Any convex body in R^d can be ε–approximated by a polyhedron with $(1/\varepsilon)^{\frac{d-1}{2}}$ facets (Dudley).

Approximate NN Searching

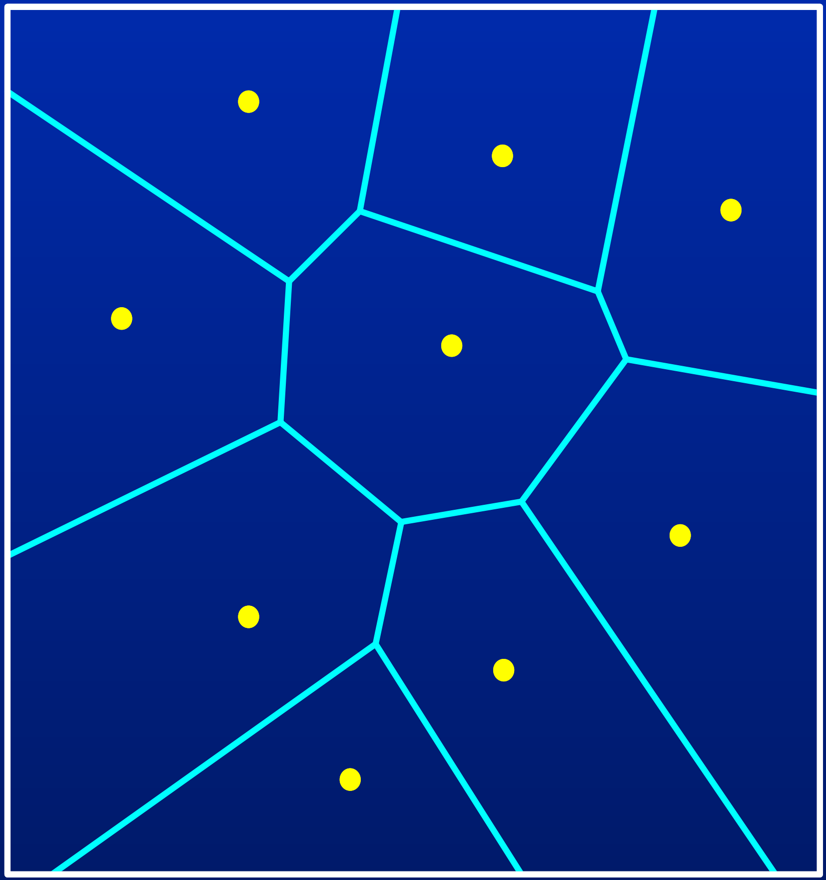
High dimensional: ($n \gg d \gg O(1)$) Polynomial size and polynomial dependence on dimension.

Kushilevitz, et al. '98, Indyk, Motwani '98 (Har-Peled, Indyk, Motwani '02).

	Query Time	Space
HIM '02	$\frac{d \cdot \log n}{\min(\varepsilon^2, 1)}$	$n^{O(1/\varepsilon^2 + \log(1+\varepsilon)/(1+\varepsilon))}$
HIM '02	$dn^{1/(1+\varepsilon)}$	$n^{1+1/(1+\varepsilon)} + dn$

Near Neighbors and Pt Location

- Given a set S of n point **sites** in R^d.
- **Voronoi diagram** is a subdivision of space into regions according to which site is closest.
- Use **point location** to answer NN queries.



Voronoi Diagrams: Difficulties

High Complexity: In dimension d, it may be as high as

$$\Theta\left(n^{\lceil d/2 \rceil}\right).$$

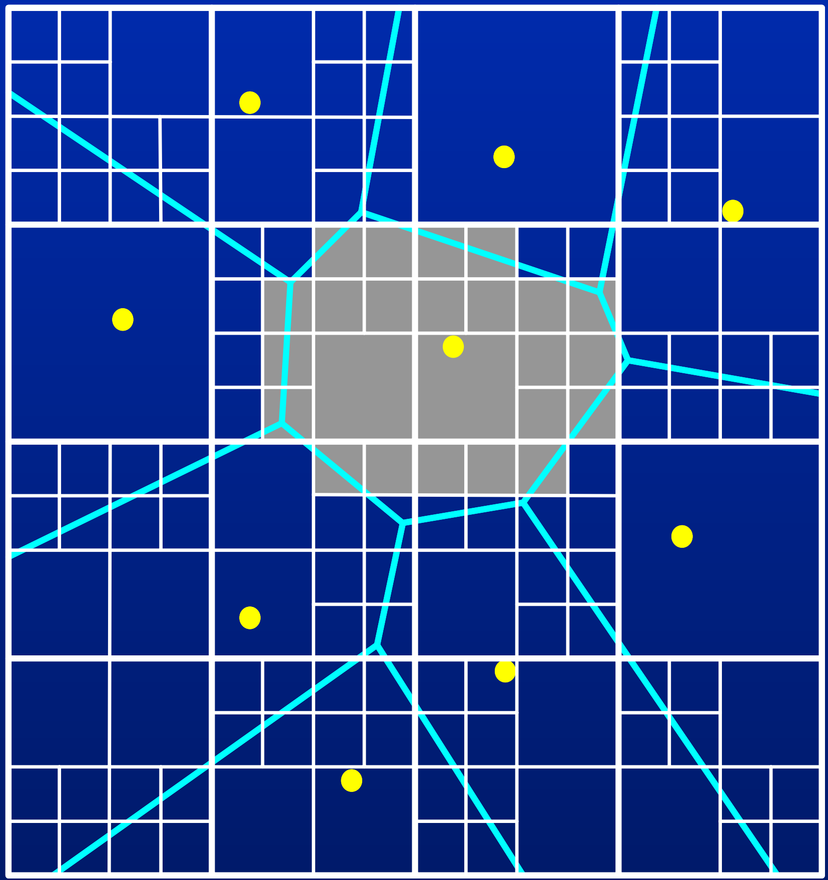
Computational Issues: Geometric degeneracies and topological consistency.

Point Location: Optimal solutions only in 2-d.

Question: Are there simpler/faster methods if we are willing to approximate?

Approx Voronoi Diagrams

- **ε-AVD:** (Har-Peled '01) Quadtree-like subdivision of space. Each cell stores a **representative site**, $r \in S$, such that r is an ε-NN of any point q in the cell.
- ε-NN → pt location



Approx Voronoi Diagrams

Har-Peled '01: Size:

$$O\left(\frac{n}{\varepsilon^d}(\log n)\left(\log\frac{n}{\varepsilon}\right)\right).$$

ε-NN Queries: Point location in a compressed quadtree in time

$$O\left(\log\frac{n}{\varepsilon}\right).$$

Variants and Extensions

Arya and Malamatos '02 explored variations/improvements to the AVD.

Well-separated pair construction: Eliminated log factors, to produce an AVD with $O(n/\varepsilon^d)$ cells.

Lower Bounds: Showed that $\Omega(n/\varepsilon^d)$ rectangular cells are needed.

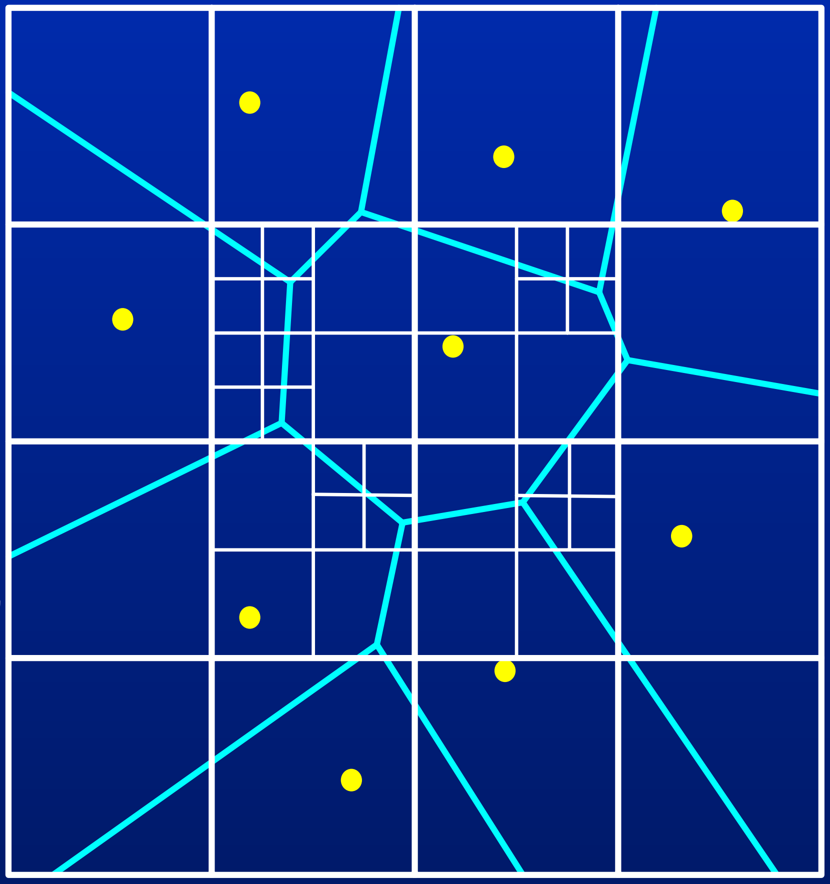
Multiple Representatives

Multi-representatives: Each cell is allowed up to $t \geq 1$ representatives.

Tradeoff: cells vs. representatives.

NN-Query: Point loc. and distance comp.

t=2



Cell/Rep Tradeoff

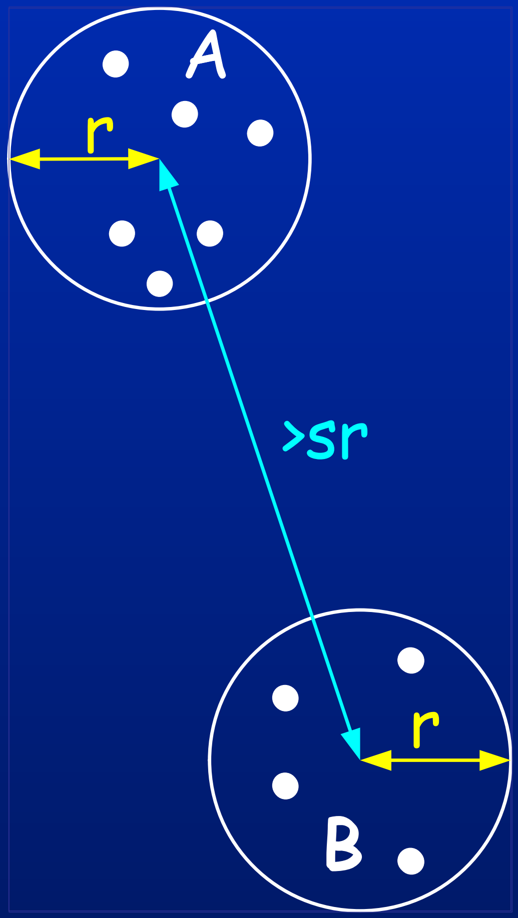
Theorem: (AM '02) Given an n-element point set S in R^d and $2 < \gamma < 1/\varepsilon$, there is an ε-AVD with $O(n\gamma^d)$ cells and $O(1/(\varepsilon\gamma)^{(d-1)/2})$ reps per cell, which can answer ε-NN queries in $O(\log (n\gamma) + 1/(\varepsilon\gamma)^{(d-1)/2})$ time.

γ	Rep/Cell	No. Cells	Query Time
$1/\varepsilon$	1	$O(n/\varepsilon^d)$	$O(\log (n/\varepsilon))$
2	$O(1/\varepsilon^{(d-1)/2})$	$O(n)$	$O(\log n + 1/\varepsilon^{(d-1)/2})$

Basic Tools: WSPDs

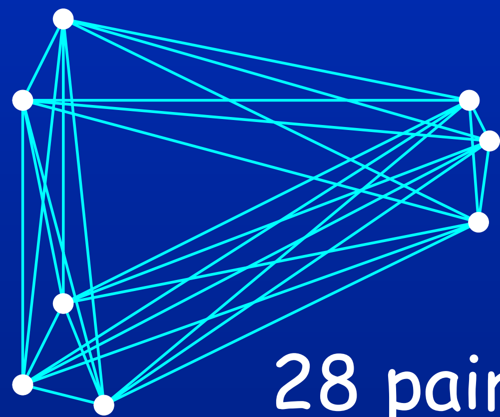
Separation factor: $s > 2$.

Two sets A and B are **well-separated** if they can be enclosed in spheres of radius r, whose centers are at distance least sr.



Basic Tools: WSPDs

Well-Separated Pair Decomposition (WSPD): Given a set of n points and separation factor s, it is possible to represent all $O(n^2)$ pairs as $O(s^d n)$ well-separated pairs. (Callahan, Kosaraju '95)

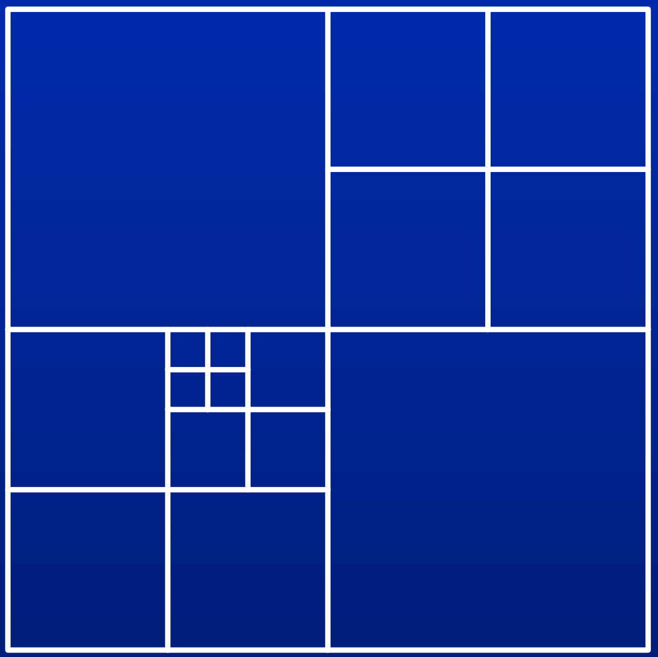


28 pairs

9 WSPs

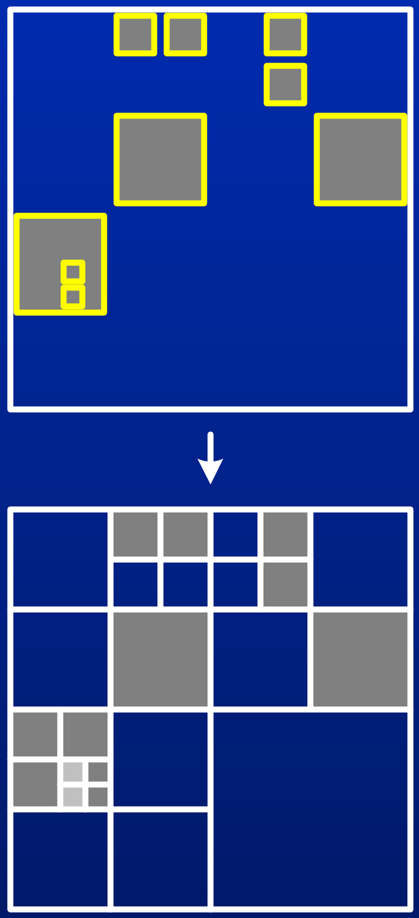
Basic Tools: BBD Trees

Quadtree Box: A box that can be obtained by repeatedly splitting the unit hypercube into 2^d identical boxes.



Basic Tools: BBD Trees

BBD Tree: Given a set of m quadtree boxes, we can build a BBD-tree of size O(m) and height O(log m) whose induced subdivision is a refinement of the box subdivision. (AMN+98)



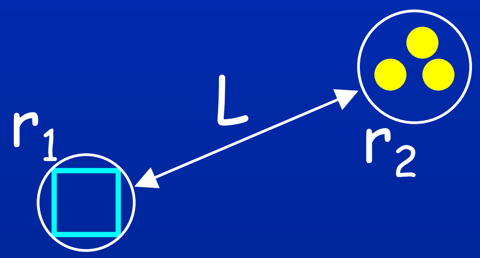
Separation & Representatives

The greater the separation from a set of points, the fewer representatives are needed to guarantee that one is an ε-NN.

Disjoint & Concentric Balls

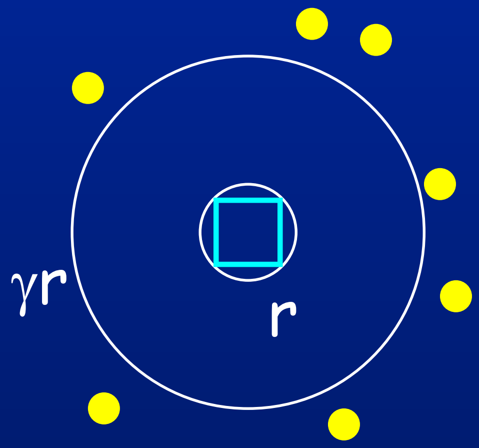
Disjoint Ball Lemma: Given disjoint balls of radii r_1 and r_2 separated by L, the number of representatives needed is

$$\left(r_1 r_2 / \left(\varepsilon L^2\right)\right)^{\frac{d-1}{2}}$$



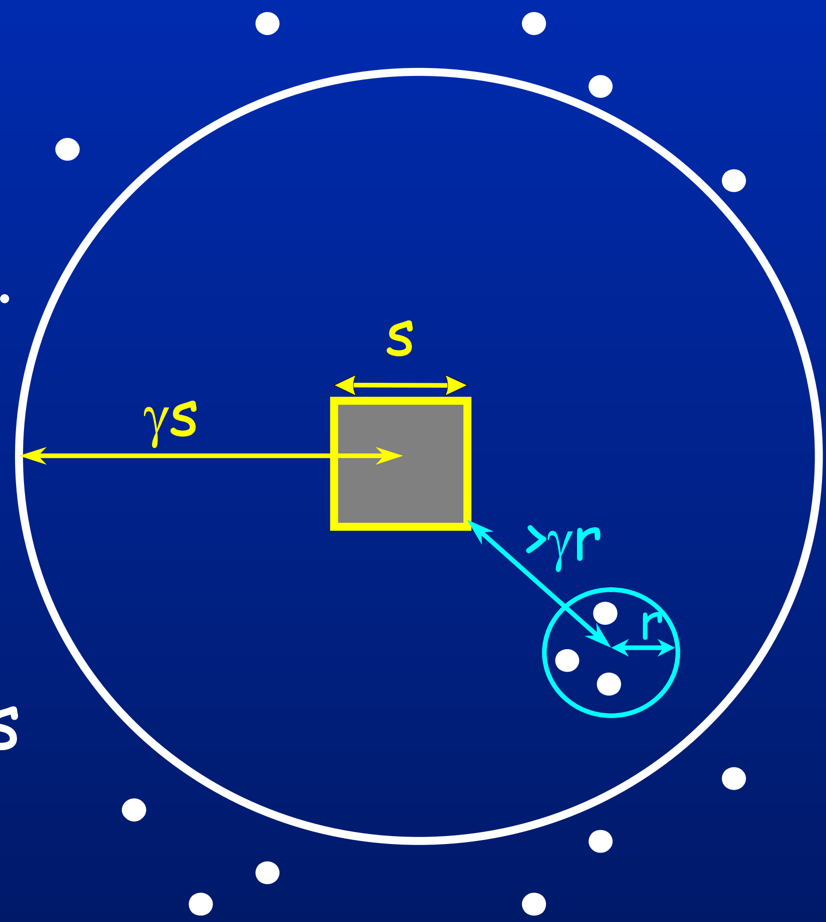
Concentric Ball Lemma: Given concentric balls of radii r and γr, the number of representatives needed is

$$1/(\varepsilon\gamma)^{\frac{d-1}{2}}$$



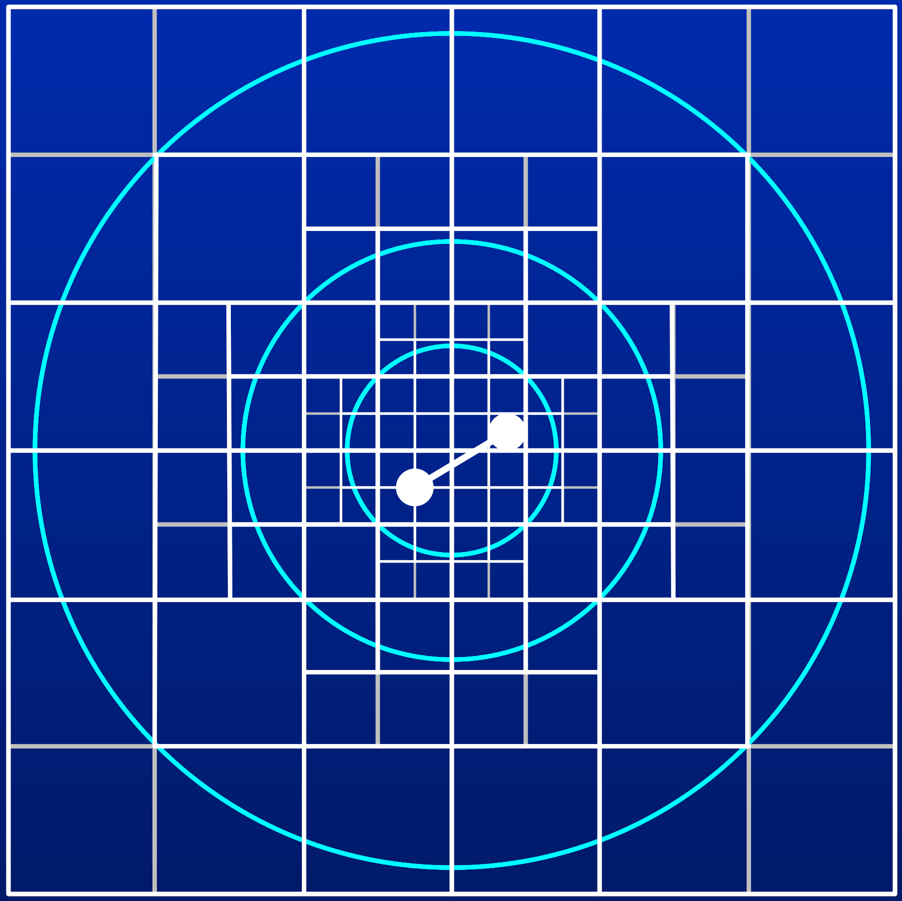
Separation Lemma (Simplified)

Lemma: Given $\gamma > 2$, there exists a subdivision with $O(n\gamma^d)$ cells. For each cell u of size s, all sites within distance γs can be enclosed within a ball whose γ expansion does not intersect u.



Construction

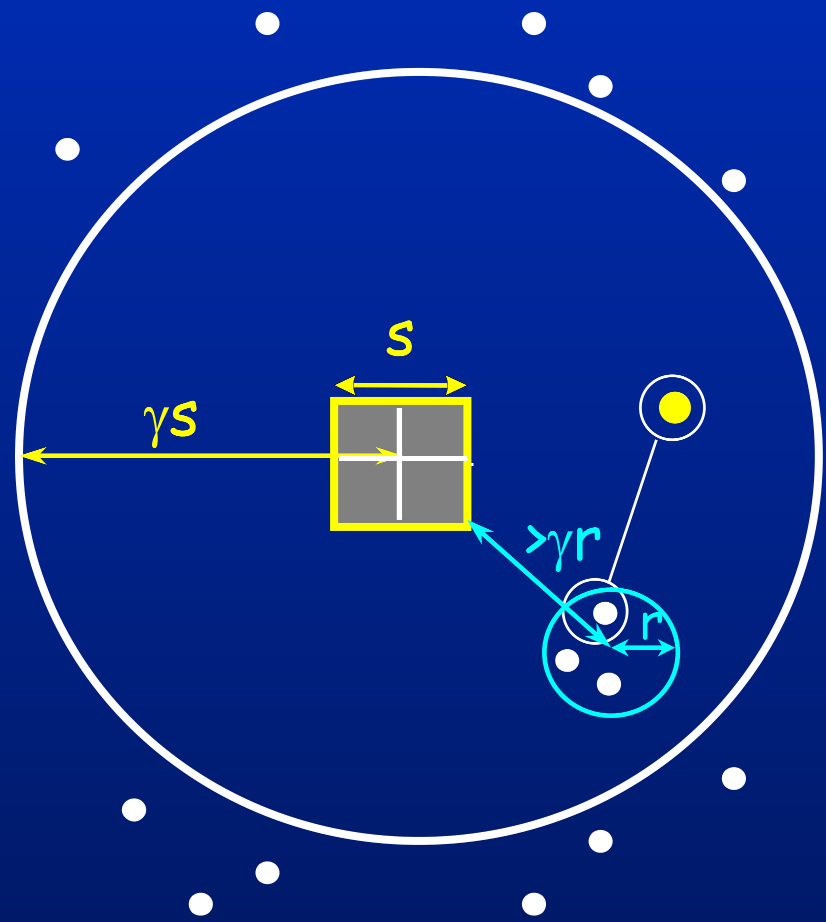
- Create a **WSPD** with separation 4.
- For each WSP, create a set of **quadtree boxes** whose sizes depend on the dist from this WSP.
- Build a **BBD tree** for these boxes.



Achieving Separation

Why does it work?

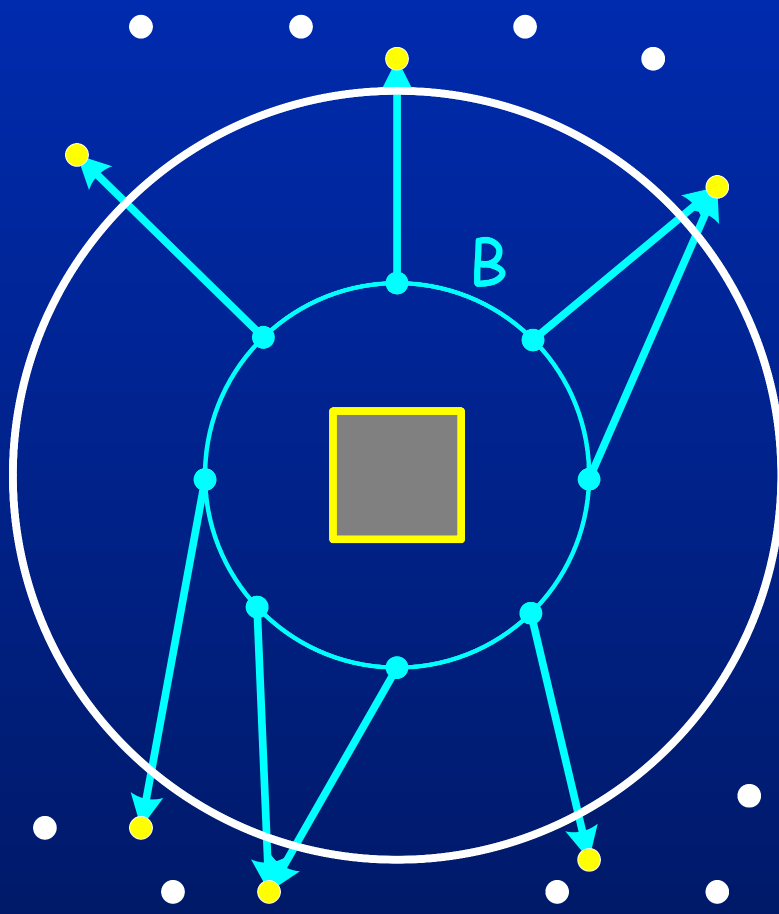
Suppose that the points within the γs expansion are not contained within a separated ball. Then there would be a well-separated pair that would force the cell to be split.



Selecting Representatives

Two-Step Approach:

- Construct a set of $1/(\varepsilon\gamma)^{(d-1)/2}$ points uniform on an intermediate sphere B.
- Reps are the nearest neighbors of these points.



Preprocessing Time

Construction time is dominated by the time to compute approximate nearest neighbors of the intermediate points.

Using Chan's algorithm, this can be done in time

$$(\text{Total Reps}) \cdot \left(\frac{1}{\varepsilon}\right)^{\frac{d-1}{2}} \log n$$

Space Efficient AVDs

Total space: For $\gamma=2$, $O(n/\varepsilon^{(d-1)/2})$ space. Can we eliminate the dependence on ε in the space?

Theorem: Given a set of n sites in R^d, $0 < \varepsilon < \frac{1}{2}$, we can build an ε-AVD with $O(1/\varepsilon^{(d-1)/2})$ reps per cell consisting of $O(n\ \varepsilon^{(d-1)/2})$ cells.

(was $O(n)$)

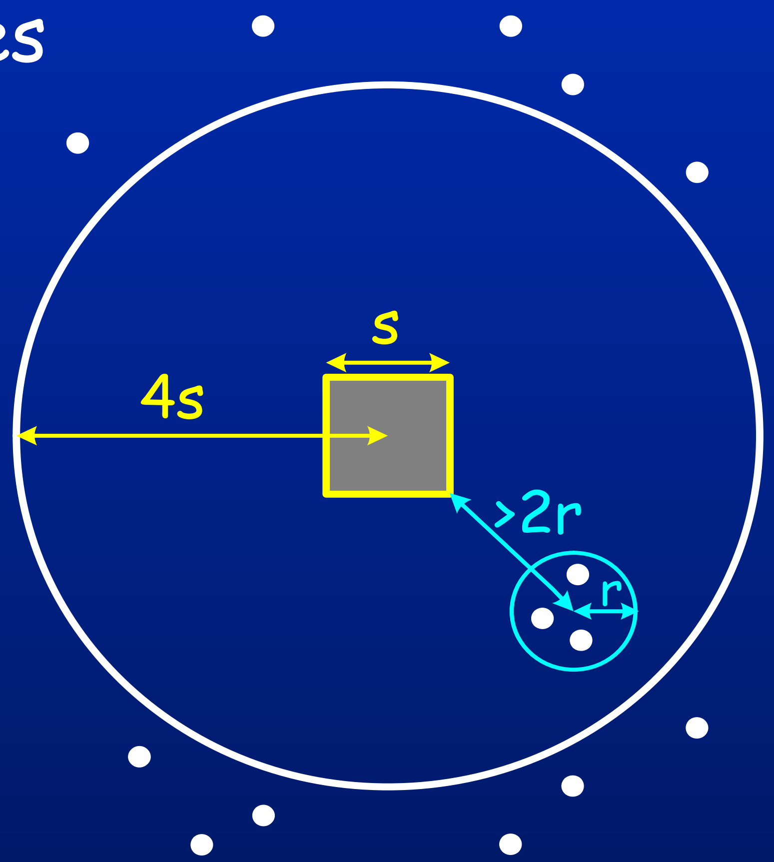
Corollary: ε-NN queries can be answered in $O(\log n + 1/\varepsilon^{(d-1)/2})$ time and $O(n)$ space.

Space Reduction: Sampling

Recall that representatives come from two sources:

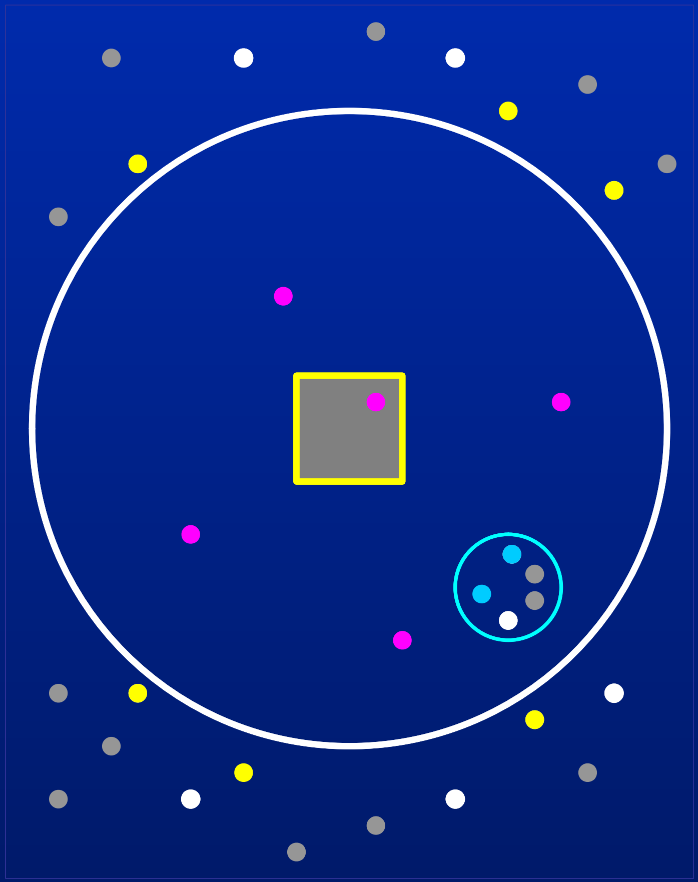
- From outside large ball
- From inner cluster
- No points exist in the remaining "no-man's land"

Idea: Allow more points into no-man's land, and make them all as reps.



Space Reduction: Sampling

- **Intuition:** Use a sample S' of $n\varepsilon^{(d-1)/2}$ points in the basic AVD construction. We expect $O(1/\varepsilon^{(d-1)/2})$ points of S to lie in no-man's land.
- **Representatives:** From outer, inner cluster, and no-man's land.



Space Reduction: Sampling

- **Deterministic Sampling:** To avoid log n factors, we sample deterministically from each node of the BBD-tree that has at most k points, but whose parent has more than k. ($k = O(1/\varepsilon^{(d-1)/2})$)
- **Stronger Separation:** Build the AVD for the sample, using twice the separation parameter value.

This guarantees O(k) reps per cell.

Extensions

- Approx **farthest-point** Voronoi diagram
- Approx **k-th order** Voronoi diagram
- Approx **spherical range counting** queries:
  - Points lying within a $(1+\varepsilon)$ expansion of the sphere may be counted.
  - (AM'95) $O(n)$ space and $O(\log n + 1/\varepsilon^{d-1})$ time.

Approximate Range Counting

Theorem: Given a point set S in R^d, and $2 < \gamma < 1/\varepsilon$, can answer ε-approx range queries with $O((n\gamma^d \log \gamma)/\varepsilon)$ space and query time $O(\log (n\gamma) + 1/(\varepsilon\gamma)^d)$.

γ	Space	Query Time
$1/\varepsilon$	$O((n \log (1/\varepsilon))/\varepsilon^{d+1})$	$O(\log (n/\varepsilon))$
2	$O(n)$*	$O(\log n + 1/\varepsilon^d)$

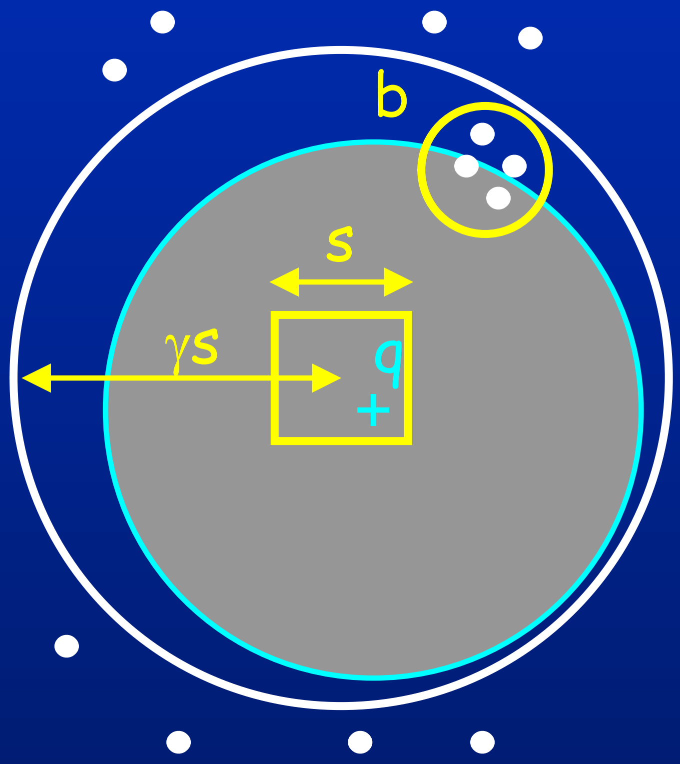
*$(1/\varepsilon)$ factor can be eliminated for $\gamma=2$

Range Searching

Cell construction: is the same as in the space-efficient AVD. We store information of size $O(1/(\varepsilon\gamma)^d)$ in both leaves and internal nodes.

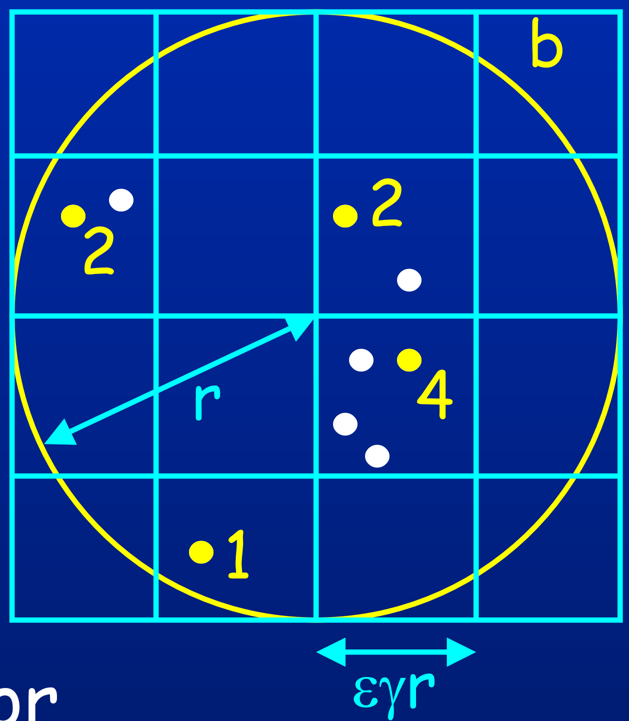
Each cell is responsible for answering queries contained within its γ-expansion.

Key: Handling points lying within cluster b.



Example: Small Range Case

Fragments: Let b be the ball of radius r containing the small cluster. Subdivide into $O(1/(\varepsilon\gamma)^d)$ fragments of side length $\varepsilon\gamma r$. For each store a **weighted representative point**.



Query: Test fragment reps for membership by brute force and sum weights.

Approx k-Near Neighbors

Approximate range query structure can be applied to answer approx k-NN queries. (k provided at query time.)

Time and space are the same:

$O((n\gamma^d \log \gamma)/\varepsilon)$ space

$O(\log (n\gamma) + 1/(\varepsilon\gamma)^d)$ query time

Conclusions

ε-AVD: A spatial subdivision in which ε-NN queries reduce to point location.

(t,ε)-AVD: Allow t representatives per cell, select the closest.

Space Efficiency: Through deterministic sampling and bisector sensitivity.

- $O(\log n + 1/\varepsilon^{(d-1)/2})$ time
- $O(n)$ space

Approx Range Queries and k-NN queries

Open Problems

- **Better bounds for range queries?** Range queries cannot used monotonicity properties that are used in nearest neighbor queries.
- **Approximating Voronoi Cells:** Some initial results by Arya and Vigneron.
- **Dependence on ε:** Must construction depend on ε?