Maintaining Nets and Net Trees under Incremental Motion

Minkyoung Cho, David Mount, and Eunhui Park

Department of Computer Science
University of Maryland, College Park

December 18, 2009

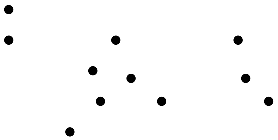
Nets

Net [KL04,HM06, CG06, GR08]

P is a finite set of points in a $\mathbb{R}^d$. Given $r > 0$, an r-net for P is a subset $X \subseteq P$ such that,

$$\max_{p \in M} dist(p, X) < r \quad \text{and}$$

$$\min_{\substack{x,x' \in X \\ x \neq x'}} \text{dist}(x, x') \geq r.$$



Features

- Intrinsic: Independent of coord. frame
- Stable: Relatively insensitive to small point motions

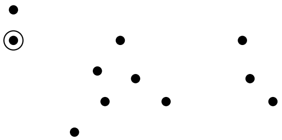
Nets

Net [KL04,HM06, CG06, GR08]

P is a finite set of points in a $\mathbb{R}^d$. Given $r > 0$, an r-net for P is a subset $X \subseteq P$ such that,

$$\max_{p \in M} dist(p, X) < r \quad \text{and}$$

$$\min_{\substack{x,x' \in X \\ x \neq x'}} \text{dist}(x, x') \geq r.$$



Features

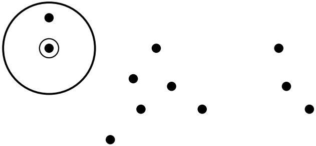
- Intrinsic: Independent of coord. frame
- Stable: Relatively insensitive to small point motions

Nets

Net [KL04,HM06, CG06, GR08]

P is a finite set of points in a $\mathbb{R}^d$. Given $r > 0$, an r-net for P is a subset $X \subseteq P$ such that,

$$\max_{p \in M} dist(p, X) < r \quad \text{and}$$
$$\min_{\substack{x,x' \in X \\ x \neq x'}} \mathrm{dist}(x, x') \geq r.$$



Features

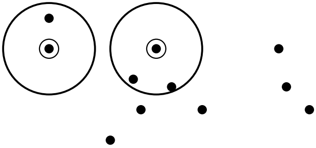
- Intrinsic: Independent of coord. frame
- Stable: Relatively insensitive to small point motions

Nets

Net [KL04,HM06, CG06, GR08]

P is a finite set of points in a $\mathbb{R}^d$. Given $r > 0$, an r-net for P is a subset $X \subseteq P$ such that,

$$\max_{p \in M} dist(p, X) < r \quad \text{and}$$
$$\min_{\substack{x, x' \in X \\ x \neq x'}} \text{dist}(x, x') \geq r.$$



Features

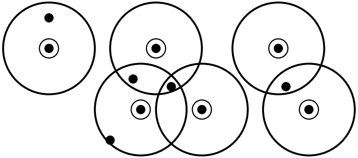
- Intrinsic: Independent of coord. frame
- Stable: Relatively insensitive to small point motions

Nets

Net [KL04,HM06, CG06, GR08]

P is a finite set of points in a $\mathbb{R}^d$. Given $r > 0$, an r-net for P is a subset $X \subseteq P$ such that,

$$\max_{p \in M} dist(p, X) < r \quad \text{and}$$
$$\min_{\substack{x,x' \in X \\ x \neq x'}} \mathrm{dist}(x, x') \geq r.$$



Features

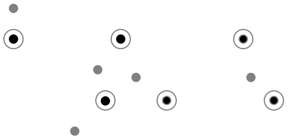
- Intrinsic: Independent of coord. frame
- Stable: Relatively insensitive to small point motions

Nets

Net [KL04,HM06, CG06, GR08]

P is a finite set of points in a $\mathbb{R}^d$. Given $r > 0$, an r-net for P is a subset $X \subseteq P$ such that,

$$\max_{p \in M} dist(p, X) < r \quad \text{and}$$
$$\min_{\substack{x,x' \in X \\ x \neq x'}} \text{dist}(x, x') \geq r.$$



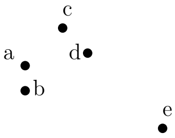
Features

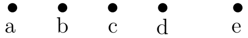
- Intrinsic: Independent of coord. frame
- Stable: Relatively insensitive to small point motions

Net Trees

Net Tree [KL04,HM06, CG06, GR08]

- The leaves of the tree consists of the points of P.
- The tree is based on a series of nets, $P^{(1)}, P^{(2)}, \ldots, P^{(h)}$, where $P^{(i)}$ is a (2^i)-net for $P^{(i-1)}$.
- Each node on level $i-1$ is associated with a parent, at level i, which lies lies within distance 2^i.

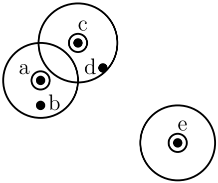


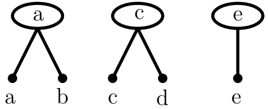


Net Trees

Net Tree [KL04,HM06, CG06, GR08]

- The leaves of the tree consists of the points of P.
- The tree is based on a series of nets, $P^{(1)}, P^{(2)}, \ldots, P^{(h)}$, where $P^{(i)}$ is a (2^i)-net for $P^{(i-1)}$.
- Each node on level $i-1$ is associated with a parent, at level i, which lies lies within distance 2^i.

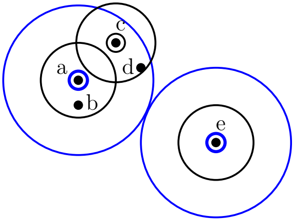


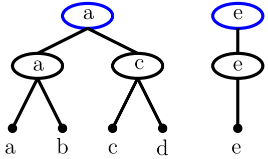


Net Trees

Net Tree [KL04,HM06, CG06, GR08]

- The leaves of the tree consists of the points of P.
- The tree is based on a series of nets, $P^{(1)}, P^{(2)}, \ldots, P^{(h)}$, where $P^{(i)}$ is a (2^i)-net for $P^{(i-1)}$.
- Each node on level $i-1$ is associated with a parent, at level i, which lies lies within distance 2^i.

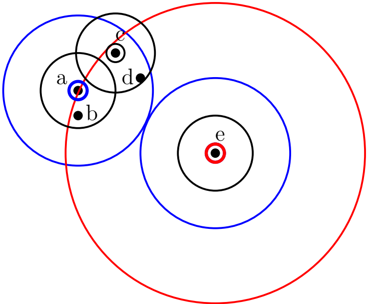


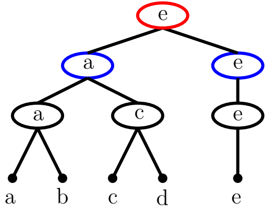


Net Trees

Net Tree [KL04,HM06, CG06, GR08]

- The leaves of the tree consists of the points of P.
- The tree is based on a series of nets, $P^{(1)}, P^{(2)}, \ldots, P^{(h)}$, where $P^{(i)}$ is a (2^i)-net for $P^{(i-1)}$.
- Each node on level $i-1$ is associated with a parent, at level i, which lies lies within distance 2^i.





Incremental Motion

Individual Updates

Insertion/deletion of a single point

KDS(Kinetic Data Structures):([Guibas98], [GGN04])

- future motion known.
- e.g., flight plans.

Incremental (Black-Box) Motion:([YZ09], [MNP+04], [Kahan91])

- Motion occurs in discrete time steps
- All points may move
- No constraints on motion, but processing is most efficient when motion is small or predictable

Competitive Ratio

Competitive Ratio

- We establish the efficiency through a competitive analysis
- Given an incremental algorithm A and motion sequence $\mathcal{P}$, define

$$\begin{aligned} C_A(P) &= \text{Total cost of running } A \text{ on } \mathcal{P} \\ C_{OPT}(P) &= \text{Total cost of optimal algorithm on } \mathcal{P} \end{aligned}$$

  The optimal algorithm may have full knowledge of future motion

- Competitive Ratio:

$$\max_{\mathcal{P}} \frac{C_A(P)}{C_{OPT}(P)}$$

Observer-Builder Model

Observer-Builder Model

- Two agents cooperate to maintain data structure [MNP+04,YiZ09]
  - Observer: Observes points motions
  - Builder: Maintains the data structure
- Certificates: Boolean conditions, which prove structure's correctness

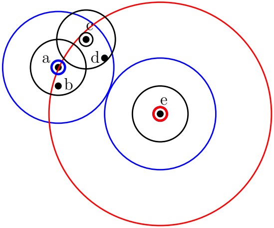
Observer-Builder Model

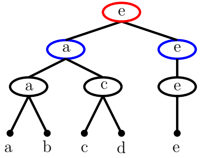
Communication Protocol

- Builder maintains structure and issues certificates
- Observer notifies builder of any certificate violations
- Builder then fixes the structure and updates certificates

Observer

Builder



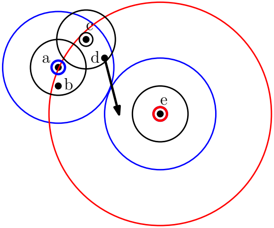


Observer-Builder Model

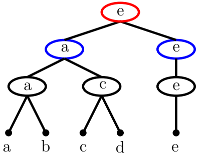
Communication Protocol

- Builder maintains structure and issues certificates
- Observer notifies builder of any certificate violations
- Builder then fixes the structure and updates certificates

Observer



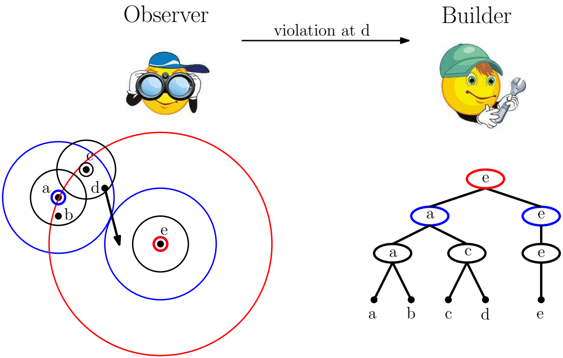
Builder



Observer-Builder Model

Communication Protocol

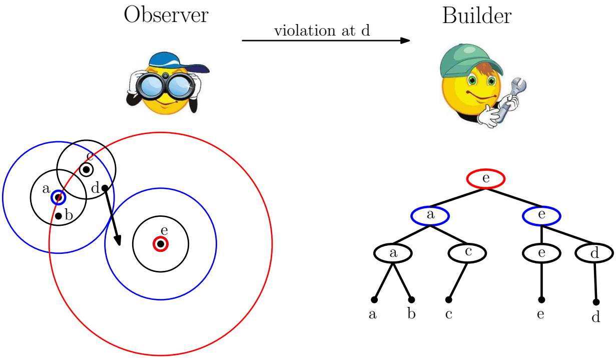
- Builder maintains structure and issues certificates
- Observer notifies builder of any certificate violations
- Builder then fixes the structure and updates certificates



Observer-Builder Model

Communication Protocol

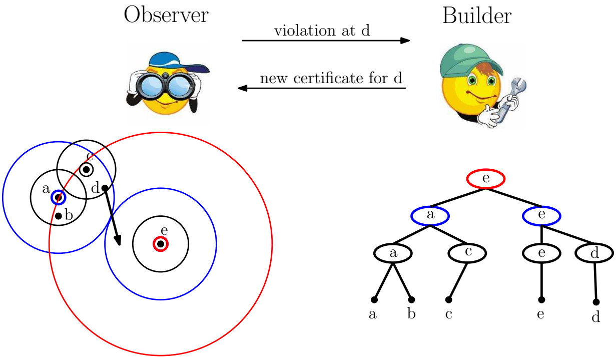
- Builder maintains structure and issues certificates
- Observer notifies builder of any certificate violations
- Builder then fixes the structure and updates certificates



Observer-Builder Model

Communication Protocol

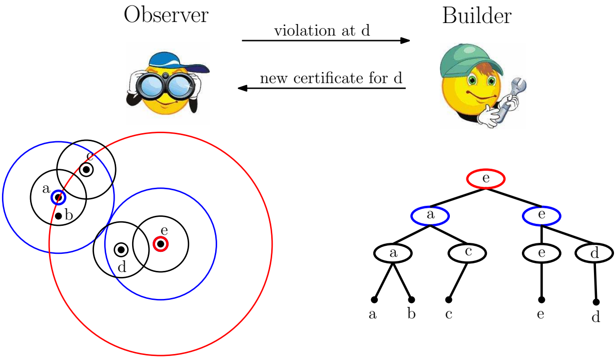
- Builder maintains structure and issues certificates
- Observer notifies builder of any certificate violations
- Builder then fixes the structure and updates certificates



Observer-Builder Model

Communication Protocol

- Builder maintains structure and issues certificates
- Observer notifies builder of any certificate violations
- Builder then fixes the structure and updates certificates



Observer-Builder — Cost Model

Cost Model

- Computational cost is the total communication complexity (e.g., number of bits) between the observer and builder.
- Builder's goal: Issue certificates that will be stable against future motion.
- Builder's and observer's overheads are not counted:
  - Builder's overhead: Is small.
  - Observer's overhead: Observer can exploit knowledge about point motions to avoid re-evaluating certificates.

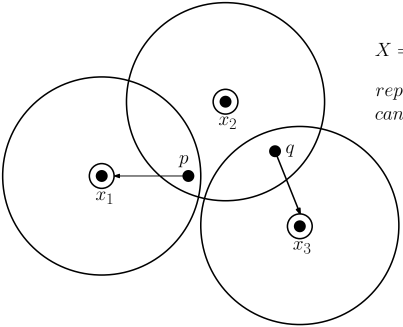
Talk Overview

- Nets, net trees, and incremental motion
- Observer-Builder model
- Maintaining nets for moving points
- Conclusion

Incremental Online Algorithm for Maintaining an r-Net

What the Builder Maintains

- The point set, P
- The r-net, X
- For each $p \in P$:
  - A representative $\text{rep}(p) \in X$, where $\text{dist}(p, x) \leq r$
  - A candidate list $\text{cand}(p) \subseteq X$ of possible representatives for p



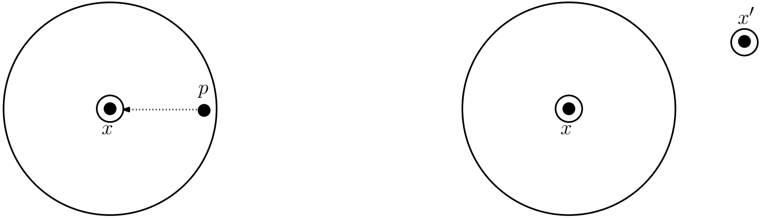
$X = \{x_1, x_2, x_3\}$

$rep(p) = x_1$
$cand(p) = \{x_1, x_2\}$

Incremental Online Algorithm for Maintaining an r-Net

Certificates

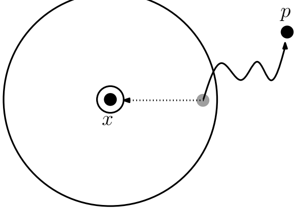
- For $p \in P$, Assignment Certificate(p): $\text{dist}(p, \text{rep}(p)) \leq r$ (representative is close enough)
- For $x \in X$, Packing Certificate(x): $\min_{\substack{x,x' \in X \\ x \neq x'}} \text{dist}(x, x') \geq r$ (no other net-point is too close)

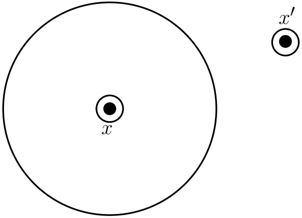


Incremental Online Algorithm for Maintaining an r-Net

Certificates

- For $p \in P$, Assignment Certificate(p): $\text{dist}(p, \text{rep}(p)) \leq r$ (representative is close enough)
- For $x \in X$, Packing Certificate(x): $\min_{\substack{x,x' \in X \\ x \neq x'}} \text{dist}(x, x') \geq r$ (no other net-point is too close)

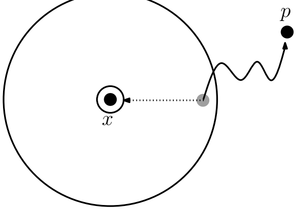


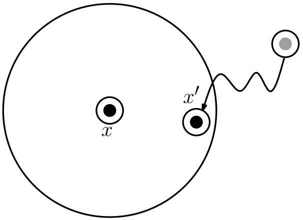


Incremental Online Algorithm for Maintaining an r-Net

Certificates

- For $p \in P$, Assignment Certificate(p): $\text{dist}(p, \text{rep}(p)) \leq r$ (representative is close enough)
- For $x \in X$, Packing Certificate(x): $\min_{\substack{x,x' \in X \\ x \neq x'}} \text{dist}(x, x') \geq r$ (no other net-point is too close)



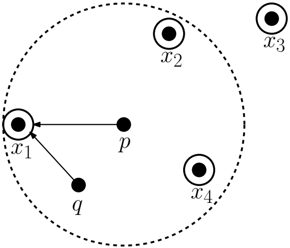


Incremental Online Algorithm for Maintaining an r-Net

Assignment Certificate Violation(p)

Point p has moved beyond distance r from its representative:

- If cand(p) has a representative x within distance r, x is now p's new representative.
- Otherwise, make p a net point (add it to X) and add p to candidate lists of points within distance r of p



$cand(p)$
x_1
x_2
x_4

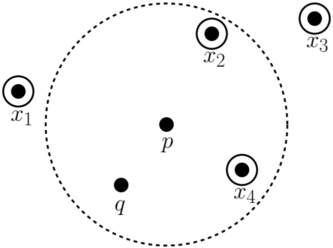
$cand(q)$
x_1
x_4

Incremental Online Algorithm for Maintaining an r-Net

Assignment Certificate Violation(p)

Point p has moved beyond distance r from its representative:

- If cand(p) has a representative x within distance r, x is now p's new representative.
- Otherwise, make p a net point (add it to X) and add p to candidate lists of points within distance r of p



$cand(p)$
x_1
x_2
x_4

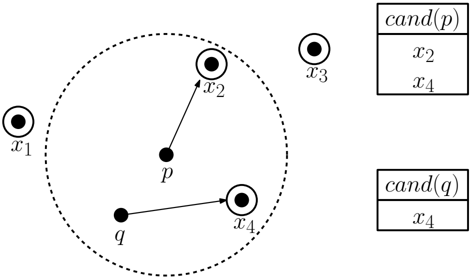
$cand(q)$
x_1
x_4

Incremental Online Algorithm for Maintaining an r-Net

Assignment Certificate Violation(p)

Point p has moved beyond distance r from its representative:

- If cand(p) has a representative x within distance r, x is now p's new representative.
- Otherwise, make p a net point (add it to X) and add p to candidate lists of points within distance r of p

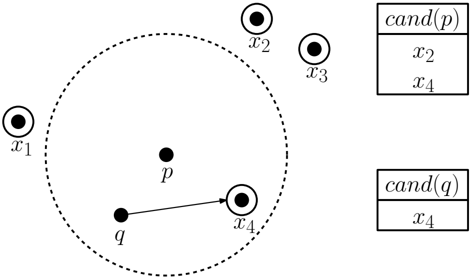


Incremental Online Algorithm for Maintaining an r-Net

Assignment Certificate Violation(p)

Point p has moved beyond distance r from its representative:

- If cand(p) has a representative x within distance r, x is now p's new representative.
- Otherwise, make p a net point (add it to X) and add p to candidate lists of points within distance r of p

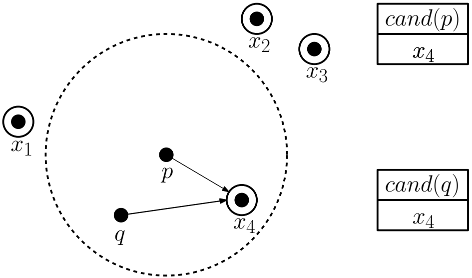


Incremental Online Algorithm for Maintaining an r-Net

Assignment Certificate Violation(p)

Point p has moved beyond distance r from its representative:

- If cand(p) has a representative x within distance r, x is now p's new representative.
- Otherwise, make p a net point (add it to X) and add p to candidate lists of points within distance r of p

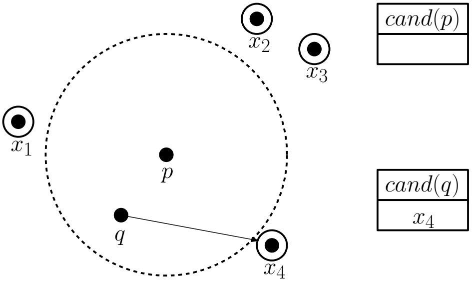


Incremental Online Algorithm for Maintaining an r-Net

Assignment Certificate Violation(p)

Point p has moved beyond distance r from its representative:

- If cand(p) has a representative x within distance r, x is now p's new representative.
- Otherwise, make p a net point (add it to X) and add p to candidate lists of points within distance r of p

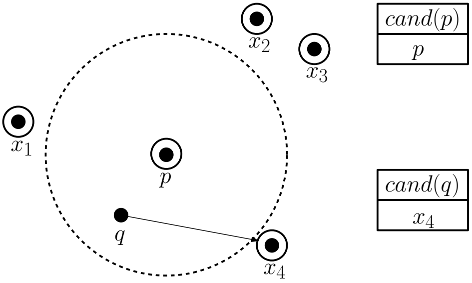


Incremental Online Algorithm for Maintaining an r-Net

Assignment Certificate Violation(p)

Point p has moved beyond distance r from its representative:

- If cand(p) has a representative x within distance r, x is now p's new representative.
- Otherwise, make p a net point (add it to X) and add p to candidate lists of points within distance r of p

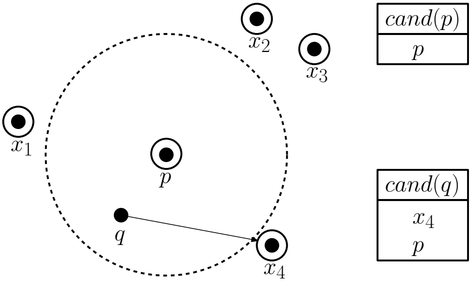


Incremental Online Algorithm for Maintaining an r-Net

Assignment Certificate Violation(p)

Point p has moved beyond distance r from its representative:

- If cand(p) has a representative x within distance r, x is now p's new representative.
- Otherwise, make p a net point (add it to X) and add p to candidate lists of points within distance r of p

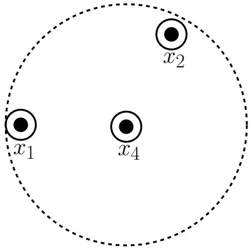


Incremental Online Algorithm for Maintaining an r-Net

Packing Certificate Violation(x)

There exists another net point within distance r of x:

- Remove all net points within radius r of x. (This may induce many assignment violations)
- Handle all assign certificate violations

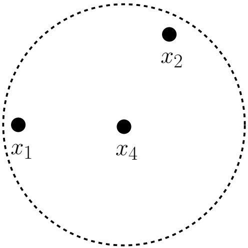


Incremental Online Algorithm for Maintaining an r-Net

Packing Certificate Violation(x)

There exists another net point within distance r of x:

- Remove all net points within radius r of x. (This may induce many assignment violations)
- Handle all assign certificate violations

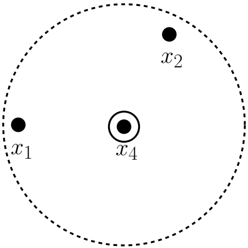


Incremental Online Algorithm for Maintaining an r-Net

Packing Certificate Violation(x)

There exists another net point within distance r of x:

- Remove all net points within radius r of x. (This may induce many assignment violations)
- Handle all assign certificate violations



Slack Net

Slack Net

- To obtain a competitve ratio, we relaxed the r-net definition slightly.
- Assume that P is from a metric space of constant doubling dimension.
- Given constants $\alpha, \beta \geq 1$, an (α, β)-slack r-net is a
- To obtain a competitve ratio, we relaxed the r-net definition slightly.

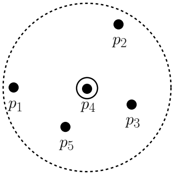
$$\max_{p \in M} dist(p, X) < \alpha\, r \quad \text{and} \quad \forall x \in X,\ |\{X \cap b(x, r)\}| \leq \beta.$$

Covering radius larger by factor α. Allow up to β net points to violate packing certificate.

Slack Net

Intuition

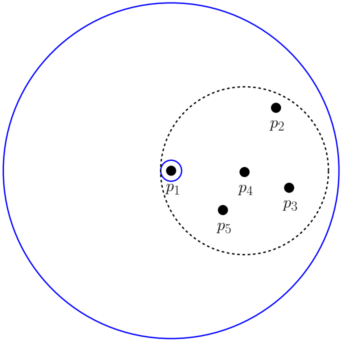
- Set $\alpha = 2$ for our net
- To give flexiablity of choosing a net point.



Slack Net

Intuition

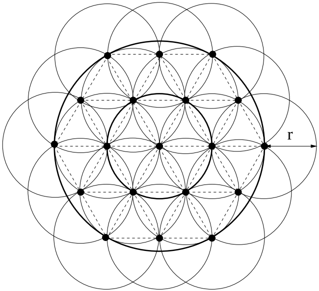
- Set $\alpha = 2$ for our net
- To give flexiablity of choosing a net point.



Slack Net

Intuition (Cont'd)

- Since we can choose any point as a net point the maximum number of net points within distance r without any violation in optimal algorithm is β, (a function of the doubling dimension constant).



Our Results

Theorem: (Slack-Net Maintenance)

There exists an incremental online algorithm, which for any real $r > 0$, maintains a $(2, \beta)$-slack r-net for any point set P under incremental motion. Under the assumption that P is a $(2, \beta)$-slack $(r/2)$-net, the algorithm achieves a competitive ratio of $O(1)$.

Theorem: (Slack-Net Tree Maintenance)

There exists an online algorithm, which maintains a $(4, \beta)$-slack net tree for any point set P under incremental motion. The algorithm achieves a competitive ratio of at most $O(h^2)$, where h is the height of the tree.

Conclusion

Summary

- Introduced a computational model for incremental motion.
- Obtain $O(1)$ competitive ratio for Nets.
- Obtain $O(h^2)$ (recently, got $O(h)$) competitive ratio for Nets.

Future Work

- Tighten competitive ratio bounds (or show they are tight)
- Implementation and testing on real data sets

Thank you!

Bibliography

- [MNP+04] D. M. Mount, N. S. Netanyahu, C. Piatko, R. Silverman, and A. Y. Wu. A computational framework for incremental motion. In *Proc. 20th Annu. ACM Sympos. Comput. Geom.*, pages 200–209, 2004.
- [YZ09] K. Yi and Q. Zhang. Multi-dimensional online tracking. In *Proc. 20th Annu. ACM-SIAM Sympos. Discrete Algorithms*, pages 1098–1107, 2009.