Algorithms and Data Structures for Embedded Network Data

Minkyoung Cho, David Mount, and Eunhui Park

Department of Computer Science
University of Maryland, College Park

MURI Meeting – December 7, 2009

Motivation

- Social networks are used to represent a variety of relational data.
  - Interconnections in social organizations, groups, and families
  - Spread of infectious diseases
  - Telephone calling patterns
  - Dissemination of information
- Social networks exhibit structural features:
  - Transitivity
  - Homophily on attributes
  - Clustering
- The likelihood of a tie is often correlated with the similarity of attributes of the actors. (E.g., geography, age, ethnicity, income).
- These attributes may be observed or unobserved.
- A subset of nodes with many ties between them may indicate clustering with respect to an underlying social space.

Latent Space Embedding (LSE)

Hypothesis

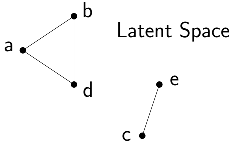
The likelihood of a relational ties depends on the similarity of attributes in an unobserved latent space.

Problem Statement

Given a network $Y = [y_{i,j}]$ with n nodes). Estimate a set of positions $Z = \{z_1, \ldots, z_n\}$ in $\mathbb{R}^d$ that best describes this network relative to some model.

Network

	a	b	c	d	e
a	-	1	0	1	0
b	1	-	0	1	0
c	0	0	-	0	1
d	1	1	0	-	0
e	0	0	1	0	-



Latent Space Embedding (LSE)

Usefulness of LSE

- Provides a parsimonious model of network structure ($O(dn)$ rather than $O(n^2)$)
- Allows for natural interpretation of geometric relations, such as "betweenness," "surroundedness," and "flatness"
- Provides a means to perform visual analysis of network structure through spatial relationships (when dimension is low), and outlier detection.
- Can be adapted to cluster the data [HRT07].
- The model is flexible and extensible.

Talk Overview

- LSE model and estimation
- Efficient incremental cost computation
- Nets and net trees
- Incremental motion model
- Maintaining nets for moving points
- Concluding remarks

LSE — Stochastic Model [HRH02]

Input

- Y, an $n \times n$ sociomatrix ($y_{i,j} = 1$ if there is a tie between i and j)
- Additional covariate information X (ignored here)

Model Parameters

- Z: The positions of n individuals, $\{z_1, \ldots, z_n\}$
- α: Real-valued scaling parameter

Stochastic Model

Ties are independent of each other, but depend on Z and α.

$$\Pr[Y \mid Z, \alpha] = \prod_{i \neq j} \Pr[y_{i,j} \mid z_i, z_j, \alpha]$$

LSE — MCMC Algorithm

Objective

Given an $n \times n$ matrix Y, determine Z and α to maximize $\Pr[Y \mid Z, \alpha]$.

MCMC — Metropolis Hastings Algorithm

- An iterative algorithm for drawing a sequence of samples $Z_0, Z_1, Z_2, \ldots$ from a distribution [MRR+53]
- Simplified View: For $k = 0, 1, 2, \ldots$
  - Sample a proposal Z from some distribution $J(Z \mid Z_k)$
  - Evaluate the decision variable

$$\rho = \frac{\Pr[Y \mid Z, \alpha_k]}{\Pr[Y \mid Z_k, \alpha_k]} \qquad (\leftarrow \text{Bottleneck})$$

  - Accept Z as Z_{k+1} with probability $\min(1, \rho)$
- Convergence may require many iterations. Efficiency is critical.

LSE — Efficient cost computation

- The LSE cost computation involves computing proximity relations among pairs of points, conditioned on the existence of an tie.
- This computation can be greatly accelerated by storing points in a spatial index, from which distance relations can be extracted.
  - Well-separated pair decomposition (WSPD): Maintain $O(n)$ clustered pairs that cover all $O(n^2)$ pairs.
  - Approximate range searching: Count the number of points lying within a spherical region of space.
- Dynamics is essential: After each iteration, points positions are perturbed. Index needs to be updated.

Talk Overview

- LSE model and estimation
- Efficient incremental cost computation
- Nets and net trees
- Incremental motion model
- Maintaining nets for moving points
- Concluding remarks

Computing Costs (Incrementally)

The spatial data structures for LSE cost computations must be highly dynamic.

Incremental Hypothesis

If point perturbations are small, then relatively few changes to spatial index.

Incremental Approach

(After each perturbation):

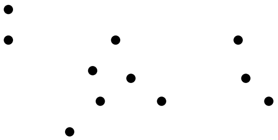
- Update spatial index (← this talk)
- Update spatial index
- Update decision variable

Nets

Net

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

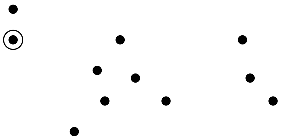
Net

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

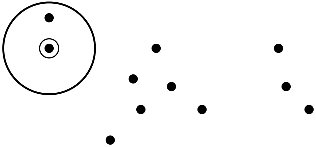


Nets

Net

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

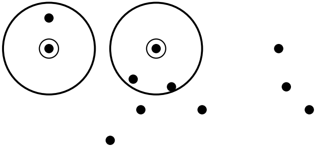
Nets

Net

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

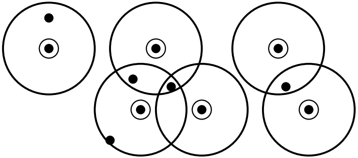
Nets

Net

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

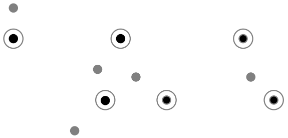
Nets

Net

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,

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

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



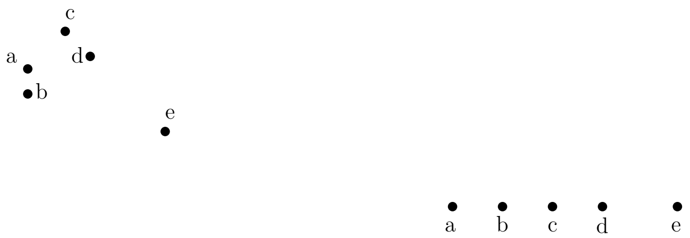
Features

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

Net Tree

Net Tree

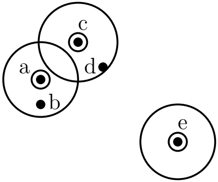
- The leaves of the tree consists of the points of P.
- The tree is based on a series of nets, $P^{(1)}, P^{(2)}, \ldots, P^{(h)}$, where $P^{(i)}$ is a (2^i)-net for $P^{(i-1)}$.
- Each node on level $i-1$ is associated with a parent, at level i, which lies lies within distance 2^i.

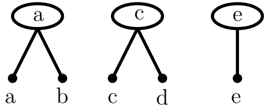


Net Tree

Net Tree

- 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.

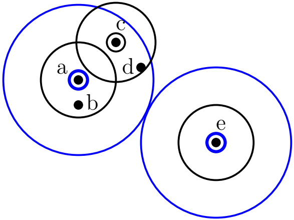


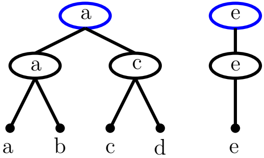


Net Tree

Net Tree

- 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.

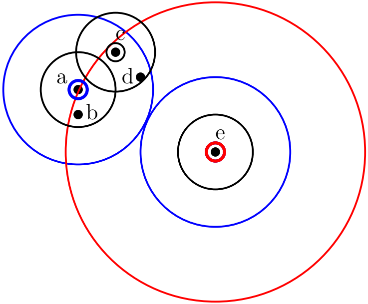


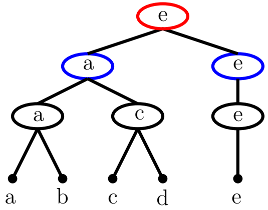


Net Tree

Net Tree

- 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.





Talk Overview

- LSE model and estimation
- Efficient incremental cost computation
- Nets and net trees
- Incremental motion model
- Maintaining nets for moving points
- Concluding remarks

Incremental Motion — Observer-Builder Model

Incremental (Black-Box) Motion

- 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

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

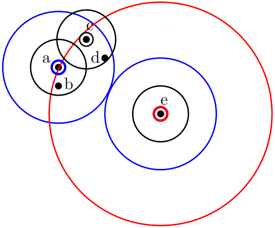
Incremental Model — Observer-Builder Model

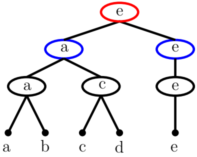
Communication Protocol

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

Observer

Builder



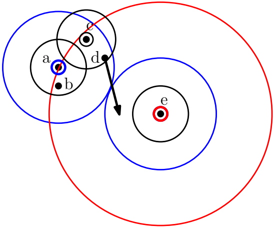


Incremental Model — Observer-Builder Model

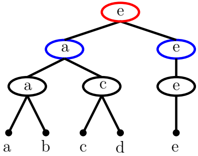
Communication Protocol

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

Observer



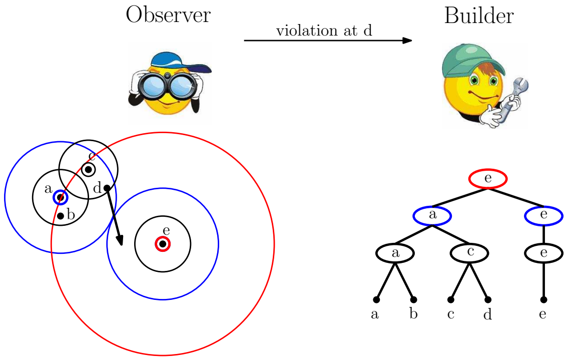
Builder



Incremental Model — Observer-Builder Model

Communication Protocol

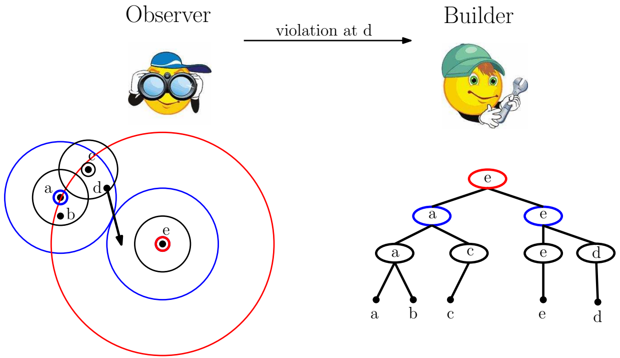
- Builder maintains structure and issues certificates
- Observer notifies builder of any certificate violations
- Builder then fixes the structure and updates certificates



Incremental Model — Observer-Builder Model

Communication Protocol

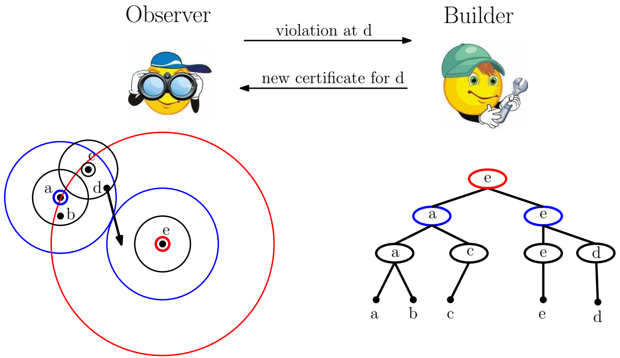
- Builder maintains structure and issues certificates
- Observer notifies builder of any certificate violations
- Builder then fixes the structure and updates certificates



Incremental Model — Observer-Builder Model

Communication Protocol

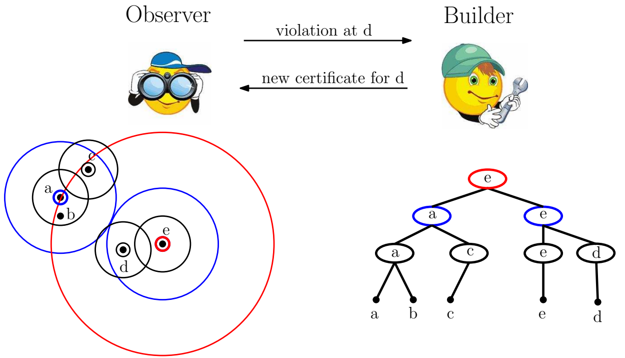
- Builder maintains structure and issues certificates
- Observer notifies builder of any certificate violations
- Builder then fixes the structure and updates certificates



Incremental Model — 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

- LSE model and estimation
- Efficient incremental cost computation
- Nets and net trees
- Incremental motion model
- Maintaining nets for moving points
- Concluding remarks

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 $\mathrm{rep}(p) \in X$, where $\mathrm{dist}(p, x) \leq r$
  - A candidate list $\mathrm{cand}(p) \subseteq X$ of possible representatives for p

Certificates

- For $p \in P$, Assignment Certificate(p): $\mathrm{dist}(p, \mathrm{rep}(p)) \leq r$ (representative is close enough)
- For $x \in X$, Packing Certificate(x): $|b(x, r) \cap X| \leq 1$ (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

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

Competitive Ratio

Competitive Ratio

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

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

  The optimal algorithm may have full knowledge of future motion
- Competitive Ratio:

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

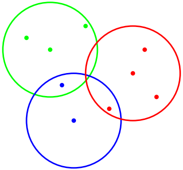
Slack Net

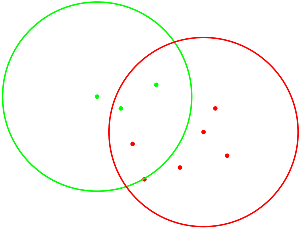
Slack Net

- To obtain a competitve ratio, we relaxed the r-net definition slightly.
- Given constants $\alpha, \beta \geq 1$, an (α, β)-slack r-net is a subset $X \subseteq P$ of points such that

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

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





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 $O(h)$, where h is the height of the tree.

Concluding Remarks

Summary

- LSE is a flexible and powerful method for producing a geometric point model for a given social network
- It estimates point positions in an unobserved social space based on a stochastic model relating network ties to distances
- Introduced a computational model for incremental motion.
- Showed how to improve efficiency of LSE computations based on MCMC approaches through the use of an online incremental algorithm (dynamically).

Future Work

- Tighten competitive ratio bounds
- Establish lower bounds (is slackness essential?)
- Implementation and tuning
- Analysis of real network data sets

Other Work Supported by this Grant

- Storing and Retrieving Information from Dynamic Data Sets:
    - Maintaining Nets and Net Trees under Incremental Motion (with M. Cho and E. Park), ISAAC'09.
    - A Dynamic Data Structure for Approximate Range Searching (with E. Park), submitted.
- Compression and Retrieval of Kinetic Data from Sensor Networks:
    - Compressing Kinetic Data From Sensor Networks (with S. Friedler), AlgoSensors'09.
    - Approximation Algorithm for the Kinetic Robust K-Center Problem (with S. Friedler), CGTA (accepted).
    - Spatio-Temporal Range Searching Over Compressed Sensor Data (with S. Friedler), submitted.
- Efficient Algorithms and Data Structures for Geometric Retrieval:
    - Space-Time Tradeoffs for Approximate Nearest Neighbor Searching (with S. Arya and T. Malamatos), JACM'09.
    - Tight Lower Bounds for Halfspace Range Searching (with S. Arya and J. Xia), submitted.
    - A Unifying Framework for Approximate Proximity Searching (with S. Arya and G. Fonseca), submitted.

Thank you!

Bibliography

- [CK95] P. B. Callahan and S. R. Kosaraju. A decomposition of multidimensional point sets with applications to k-nearest-neighbors and n-body potential fields. *J. Assoc. Comput. Mach.*, 42:67–90, 1995.
- [HRH02] P. D. Hoff, A. E. Raftery, and M. S Handcock. Latent space approaches to social network analysis. *J. American Statistical Assoc.*, 97:1090–1098, 2002.
- [HRT07] M. S. Handcock and A. E. Raftery and J. M. Tantrum. Model-based clustering for social networks. *J. R. Statist. Soc. A*, 170, Part 2, 301–354, 2007.
- [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.*, 200–209, 2004.
- [MRR+53] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21:1087–1092, 1953.
- [YZ09] K. Yi and Q. Zhang. Multi-dimensional online tracking. In *Proc. 20th Annu. ACM-SIAM Sympos. Discrete Algorithms*, 1098–1107, 2009.