# Clustering Mobile Nodes

◇◇

Based on the paper: **Discrete Mobile Centers**

J. Gao, L. Guibas, J. Hershberger, L. Zhang, and A. Zhu

DCG 2002

◇◇

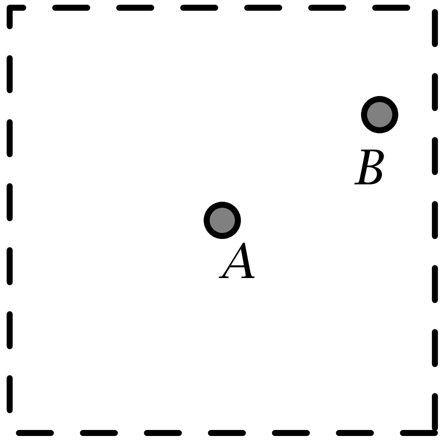
# Motivation

We have an *ad hoc* mobile communication network. Clustering the mobile nodes can serve many functions:

1. Simple intra-cluster communication
2. Remove redundancy when nodes have sensors
3. Data aggregation
4. Recycle scarce resources (e.g., frequency spectrum)
5. Provide a hierarchical structure for network management

## The Setting

We have $n$ mobile nodes (points) in the plane. Each node can communicate with other nodes in its neighborhood.



Nodes that can communicate are said to be *visible* to each other.

# Clustering Mobile Nodes

We wish to create clusters of nodes that are visible from a particular node, called the *cluster center*.

A good clustering should

- use a number of clusters within a constant factor of the optimum
- be *stable* under node motion
- be computable through simple, local operations

# Background

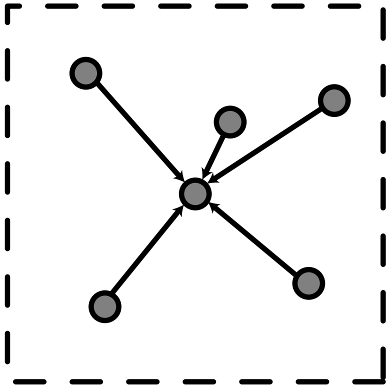
The static clustering problem has a vast literature. It is NP-complete, but a variety of polynomial-time approximation schemes (PTAS) have been proposed.

Some mobile clustering algorithms have been considered in the *ad hoc* mobile networking literature. However, no prior theoretical analysis has been given.

# Randomized Leader Election (RLE)

At the beginning of time each mobile node is assigned a random UID.

Each node nominates as a leader the node of highest UID visible to it (this could be the node itself). These leaders become the cluster centers. A cluster is defined by a leader and all its nominators.



# 1-Dimensional RLE Analysis

Among mutually visible points only one can be nominated by the others.

**Theorem 1** *1-D RLE produces an* $\Theta(\log n)$*– approximation clustering in expectation.*

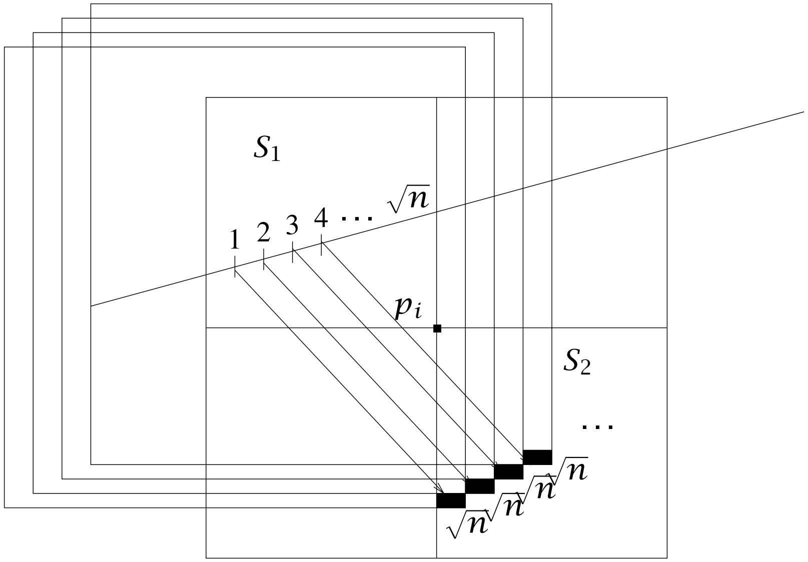
In fact, the probability that the number of the selected centers exceeds $ck \log n$ is $O(1/n^{\Theta(c^2)})$ ($k =$ optimal number of centers).

## 2-Dimensional RLE Analysis

**Theorem 2** *2-D RLE produces an $\Theta(\sqrt{n})$– approximation clustering. Furthermore, the probability that there are more than $\sqrt{n}\log n \cdot k$ centers is $O(1/n^{\log n - 1})$, where $k$ is the optimal number of centers.*

# A Bad Situation in 2-D

Here is how many centers can arise.



Lower bound for the 2-D case.

# Hierarchical Clustering

The hierarchical clustering algorithm (HRLE) proceeds in a number of *rounds*. At each round we apply the RLE algorithm to the centers produced by the previous round, but using a larger and larger covering ball.

The authors use $\lg \lg n - 1$ rounds; during round $i$ the covering ball of each node is of size $\delta_i = 2^i / \lg n$.

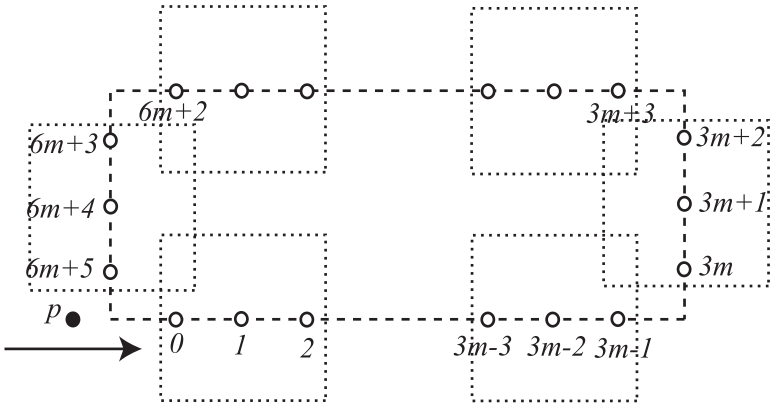
**Theorem 3** *Unit balls placed at the final centers produced by HRLE cover all the nodes.*

# Quality of HRLE

**Theorem 4** *The cover produced by HRLE is a constant factor approximation to the optimal node cover with high probability.*

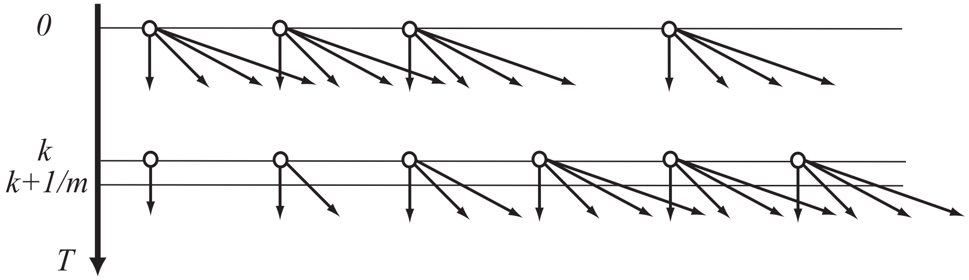
# Event Analysis — Optimal Clustering

**Theorem 5** *The number of changes of the optimal cover for $n$ points in motion is $\Theta(n^3)$ in the worst case.*



# Event Analysis — Approximate Clustering

**Theorem 6** *For any constant $c > 1$, there exists a configuration of $n$ points moving linearly on the real line so that any $c$-approximate cover undergoes $\Omega(n^2/c^2)$ changes.*



# Kinetic HRLE Analysis

**Theorem 7** *For any bounded degree algebraic motions, the number of events processed by the kinetic HRLE is at most* $O(n^2 \log\log n)$*, and hence the HRLE is efficient in the KDS sense.*

**Theorem 8** *For any bounded degree algebraic motions, kinetic HRLE maintains an* $O(1)$*-approximate covering through the entire motion, with high probability.*

# Further Kinetic HRLE Qualities

**Theorem 9** *The expected update cost for one event is* $O(\log^{\xi} n)$*, where* $\xi = 2 + \lg 3 \approx 3.6\ldots$*. Hence the KDS is responsive in an expected sense.*

The appearance/disappearance of a node $p$ at a certain level of HRLE causes at most 5 node-center pairing changes at that level, in expectation. It can also cause up to 3 center creations/destructions at that level, in expectation.

# Kinetic HRLE Qualities, continued

**Theorem 10** *The kinetic data structure uses* $O(n \log n \log \log n)$ *storage, and hence the KDS is compact.*

**Theorem 11** *Each point participates in at most* $O(\log \log n)$ *ordering certificates; therefore, the KDS is local.*

# Distributed Implementation

The kinetic HRLE algorithm can be implemented in a fully distributed manner. Each node only needs to be informed when a neighboring node enters or leaves one of its $\lg \lg n$ ranges. Note that a node can detect such changes by broadcasting ‘‘who is there’’ messages at different power levels. Thus the actual positions of the nodes need not be known, obviating the need for expensive equipment, such as GPS.

In the distributed implementation the total storage is $O(sn)$, where $s$ is the maximum number of nodes visible to a given node. Of course $s$ can be $\Theta(n)$, but in practice it is much less. This can be reduced to worst-case storage $O(n^\epsilon)$ per node, by storing just a subset of each node’s neighbors and running RLE only among those.

# Demo: Videos

## Conclusion

The HRLE algorithm can be easily extended to higher dimensions.

The clusterings it produces are *stable* in a strong sense.

Its implementation is fairly straightforward.

# Open Problems

Many open problems remain:

- differently-shaped ranges ($L_2$ balls)
- derandomization
- trade-off between stability and optimality of clustering
- related problems: route discovery and maintenance, sensor data compression, node misbehavior detection