Exploiting Temporal Stability and Low Rank Structure for Mobile Network Localization

Swati Rallapalli

Joint work with
Lili Qiu  Yin Zhang  Yi-Chao Chen
The University of Texas at Austin

ACM MobiCom 2010
September 22, 2010
Motivation

- Localization for mobile networks is important
  - Many applications in mobile networks
    - Underwater surveillance, vehicle tracking, habitat monitoring, disaster recovery networks, geographic routing ..

- GPS is not always suitable due to
  - high cost & power requirement
  - lack of line of sight to satellites
State of the art

- **Most works focus on static networks**
  - Centroid [Bulusu2000]: node location is center of anchors heard
  - MDS [Shang03]: find locations to fit distance estimates
  - Sextant [Guha05]: region based localization

- **Few works on mobile network localization**
  - MCL [Hu04], MSL* [Rudafshani07]...
    - estimate current location based on previous location and maximum velocity

1. Do not fully use information within each time interval
2. Only use maximum velocity information
Our contributions

• Identify structure in real mobility

• Develop novel localization algorithms to exploit this structure

• Experimentally show they out-perform by 2-6x
**Structure in mobility**

- **Stacked node coordinate matrix**

  $\begin{bmatrix}
  x(1, t_1) & x(1, t_2) & \cdots & x(1, t_n) \\
  \vdots & \vdots & \ddots & \vdots \\
  x(n, t_1) & x(n, t_2) & \cdots & x(n, t_n) \\
  y(1, t_1) & y(1, t_2) & \cdots & y(1, t_n) \\
  \vdots & \vdots & \ddots & \vdots \\
  y(n, t_1) & y(n, t_2) & \cdots & y(n, t_n)
  \end{bmatrix}$

  - **X coordinates of n nodes**
  - **Y coordinates of n nodes**

- **We expect temporal stability & low rank structure**

- **Intuition:** Nodes move at same velocity for a while $\Rightarrow$ temporal stability
  - i.e. current location of a node can be approximated as mid point of its location at previous & next time interval
Structure in mobility (cont.)

- This also means mobility exhibits low rank structure
  - Low rank structure

\[ M_{n \times m} \approx U_{n \times r} \ast V_{r \times m} \]

- Why is \( M \) low rank?
  - Suppose nodes move with constant velocities \( \Rightarrow M \) is rank 2
    \[ M = z \cdot 1^T + v \cdot t^T \]
    \( z \): initial coordinates, \( v \): velocities, \( t \): current time, \( t \): time vector
  - Different nodes may share the same trajectory
    - e.g. vehicles/people on roads, animals/people on popular routes
Traces used in mobility analysis

- **Real traces**

<table>
<thead>
<tr>
<th>Trace name</th>
<th>Description</th>
<th>Matrix size (nodes x intervals)</th>
<th>Interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cabspotting</td>
<td>Taxis in SF Bay area</td>
<td>100 x 300</td>
<td>1 min.</td>
</tr>
<tr>
<td>Shanghai taxi</td>
<td>Taxis in Shanghai</td>
<td>100 x 300</td>
<td>1 min.</td>
</tr>
<tr>
<td>Seattle bus</td>
<td>Buses in Seattle</td>
<td>545 x 300</td>
<td>1 min.</td>
</tr>
<tr>
<td>ZebraNet</td>
<td>Traces of zebras</td>
<td>61 x 90</td>
<td>8 mins.</td>
</tr>
<tr>
<td>Human mobility</td>
<td>KAIST students</td>
<td>92 x 499</td>
<td>30 secs.</td>
</tr>
</tbody>
</table>

- **Synthetic traces**
  - **Standard random waypoint**
    - select random destination in deployment area
    - select random velocity and move towards destination
  - **Modified random waypoint [Hu04]**
    - different velocity selected every time interval
Analysis of temporal stability

Normalized Velocity Change

\[ NVC(i, t) \triangleq \frac{||\vec{v}(i, t) - \vec{v}(i, t - 1)||_2}{\text{mean}_{i,t}(||\vec{v}(i, t)||_2)} \]

// node i, time t

In 4 out of 5 real traces over 36% times velocity is identical.
Analysis of low rank structure

Rank-K approximation error

\[ 1 - \left( \frac{\sum_{i}^{K} s_{i}^{2}}{\sum_{i} s_{i}^{2}} \right) \]

Rank-5 approximation error

- 5.1% - 23.4% in real traces
- 2.8% - 37.9% in synthetic traces.
Observations

- Mobility traces exhibit different degrees of temporal stability and low rank structure.

- Localization algorithms therefore:
  - can exploit these properties
  - should adapt to different degrees of properties
Our localization algorithms

• Minimize\(_{\{M,U,V\}}\) \[\sum_t f(M, t) + \alpha \cdot g(M) + \beta \cdot h(M, U, V)\]

- \(\alpha, \beta\): relative weights of temporal stability & low rank constraints

• Three variants:
  - Low Rank based Localization (LRL) \(\Rightarrow\) \(\alpha = 0\)
  - Temporal Stability based Localization (TSL) \(\Rightarrow\) \(\beta = 0\)
  - Temporal Stability & Low Rank based Localization (TSLRL) \(\Rightarrow\) \(\alpha, \beta \neq 0\)
Fitting error: range based schemes

- Problem formulation
  - Given connectivity/distance information in a network, find nodes’ locations that minimize the fitting error $f(M,t)$

- Range-based localization (use distance information)
  - Equality constraints: derive using RSS measurements
    - RSS is a function of the distance
  - Upper bound constraints: for nodes that hear each other
    - directly $\rightarrow$ communication range
    - indirectly $\rightarrow$ shortest path distance
  - Lower bound constraints: for nodes that do not hear each other $\rightarrow$ communication range

Total violation of
Fitting error: range free schemes

• Useful when nodes only get connectivity information but not the exact distance estimation
  – For example - no RSS measurement
• Difference from range based schemes
  – Remove equality constraints from \( f(M,t) \)
  – Change shortest path distance in the upper bound constraints
    • For example - if 2 nodes are 3 hops away, use 3R as upper bound
Incorporate temporal stability

$$\sum_t f(M, t) + \alpha \cdot g(M) + \beta \cdot h(M, U, V)$$

$$g(M) \triangleq \sum_{i,t} \{M(i, t - 1) + M(i, t + 1) - 2 \cdot M(i, t)\}^2$$

- In matrix form

$$g(M) \triangleq ||M \ast T^T||_F^2$$

$$T = \begin{bmatrix}
1 & -2 & 1 & 0 & \cdots \\
0 & 1 & -2 & 1 & \vdots \\
0 & 0 & 1 & -2 & \vdots \\
\vdots & \vdots & \vdots & \vdots & \ddots
\end{bmatrix}$$

Small $g(M) \Rightarrow$ similar velocities at adjacent time intervals
Incorporate low-rank structure

\[ \sum_{t} f(M, t) + \alpha \cdot g(M) + \beta \cdot h(M, U, V) \]

\[ h(M, U, V) \triangleq \left\| M - U \ast V^T \right\|_F^2 \]

\( M: (d \cdot N) \times t_{\text{max}}: \) coordinate matrix

\( U: (d \cdot N) \times r: \) unknown factor matrix

\( V: t_{\text{max}} \times r: \) unknown factor matrix

\( r: \) desired low rank

Small \( h(M, U, V) \) \( \Rightarrow \) \( M \approx U \ast V^T \) \( \Rightarrow \) \( M \) has a good rank-\( r \) approximation
Optimization

• Challenges
  – Non-linear optimization
  – Potentially a large number of variables and constraints

• Our approach
  – Use quasi-Newton algorithm (L-BFGS) to find local optimum
    • Memory efficient
    • Suitable for large optimization problems
  – Compute good initial solution
    • Solve the optimization problem in a higher dimension
    • Project the solution onto a lower dimension and use it as the initial solution
Simulation

• Baseline algorithms
  – Static: Centroid, MDS, Sextant
  – Mobile: MCL, MSL*

• Default parameters as in existing work [Hu04]
  – Node density = 10; Anchor density = 1
    • Area: 200mx200m; # of nodes: regular 45, anchors: 5
  – Maximum speed = 10 m/interval
  – Range = 50 m

• Metric: Mean Absolute Error = mean_{i,t}(dist(M(:, i, t), M'( :, i, t)))

• Mobility traces
  – Real traces
    • Shanghai & SF cabs, Seattle buses, ZebraNet, KAIST students
  – Synthetic traces
    • Modified random waypoint, Standard random waypoint
Varying mobility: synthetic traces

Modified random waypoint

TSLRL improves MAE by 1.1-63.5x over existing schemes
Varying mobility: real traces

San Francisco Taxis

Existing schemes

Our range free

Our range based

Mean absolute error

TSLRL improves MAE by 6 - 8.8x
Varying density

Our schemes perform the best under all node and anchor densities!
Varying noise

Modified random waypoint

Our schemes are flexible and degrade gracefully with noise!
Testbed

- Mica2 motes with 915 MHz radios deployed on a single floor
- Mobility model: standard random waypoint
- Topology

- Distance calculation:
  - Use RSS measurements

![Diagram](Regular)

![Diagram](Irregular)

Path loss exponent = 2.7
Range = 2.3 m
Testbed: regular topology

TSLRL improves MAE by 2.2-3.4x
TSLRL improves MAE by 2.1-3.6x in irregular topology
Conclusions

• Show that real & synthetic mobility traces exhibit
  – temporal stability
  – low rank structure

• Develop novel localization schemes that exploit these properties

• Experimentally show that they out-perform existing schemes by 2-6x
  – TSLRL performs particularly well under a wide range of scenarios
Thank you!

swati@cs.utexas.edu
Convergence

Our schemes need no convergence time!
Running time

Our schemes take about a second per interval & can be optimized further

Intel Core 2 Duo CPU 2.66 GHz processor, 2 GB memory (50 node network)
Tuning parameters: $\alpha$, $\beta$ and $r$

$$\sum_t f(M, t) + \alpha \cdot g(M) + \beta \cdot h(M, U_d \cdot N \times r, V_{t_{\text{max}}} \times r)$$

- Adapt $\alpha$ according to ratio $= \frac{\sum_t f(M, t)}{g(M)}$
  - $\alpha$ increases with $f(M, t)$ to avoid being over-dominated
  - $\alpha$ decreases with large $g(M)$ since it indicates weaker temporal stability
  - Since we do not know the values of $f(M, t)$ and $g(M)$ in advance we use iteration
    - Iteration 1: initialize $\alpha = 1$
    - Iteration 2: set $\alpha = \max(\min(\text{ratio}, 10), 1)$
- Use $\beta = 0.1$ (we can potentially adapt $\beta$)
- Use $r = 3$
Varying mobility: synthetic traces

1. Results similar to modified random way point
2. TSL, TSLRL do even better
Fitting error: $f(M,t)$

$$f(M, t) \triangleq \sum_{ij} (D_{ij}(t) - D_{ij}^{eq}(t))^2 + \sum_{ia} (C_{ia}(t) - C_{ia}^{eq}(t))^2 +$$

**Equality**

$$\sum_{ij} \min\{0, D_{ij}(t) - D_{ij}^{lb}(t)\}^2 + \sum_{ia} \min\{0, C_{ia}(t) - C_{ia}^{lb}(t)\}^2 +$$

**Lower bound**

$$\sum_{ij} \max\{0, D_{ij}(t) - D_{ij}^{ub}(t)\}^2 + \sum_{ia} \max\{0, C_{ia}(t) - C_{ia}^{ub}(t)\}^2$$

**Upper bound**