New Challenges in Distributed Sensing, Processing and Query of Spatial Data

Jie Gao
Stony Brook University
ALGO’17, 9/8/2017
Wireless and Sensor Networks 2000-2010

• Lightweight wireless sensor nodes embedded in the environment.
• Scientific data collection, environment monitoring, etc.
Distributed Algorithms for Data Collection, Processing and Query

• Sensor deployment and coverage.
• Network management: topology control, power control.
• Routing: one-to-one routing, data collection and aggregation.
• Distributed storage & data-centric routing.
Wireless and Sensor Networks 2010-now

- Smart phone sensing
- Wearable devices
Wireless and Sensor Networks 2010-now

- Cyber-physical systems, Internet of Things (IoT)
- Sensing + communication & processing + control
Algorithmic Problems?

Communication & networking:
• Sensor duty cycle scheduling
• Ultra-low delay 10~100ms → 1ms
Algorithmic Problems?

Communication & networking:
• Sensor duty cycle scheduling
• Ultra-low delay

Data processing and query:
• Big data:
• Spatial data & distributed query

Security and Privacy
Location and Trajectory Privacy

• GPS is everywhere.
• Locations/trajectories are collected.
Many Applications of Trajectory Data

• Traffic analysis and mining
• Optimization of transportation system
• Anomaly detection
• Crime investigation
Trajectories are sensitive & identifying

- Frequently visited locations $\rightarrow$ home/work address; predictability of location > 93%

Limits of Predictability in Human Mobility, Science, 2010.
Trajectories are sensitive & identifying

- Frequent co-location patterns $\rightarrow$ social ties
Trajectories are sensitive & identifying

- Motifs – revealing activities

Trajectories are sensitive & identifying

• Unique signature.

Trajectories are sensitive & identifying

- Unique signature – 4 spatial-temporal points are enough to identify 95% trajectories in 1.5 million users.
Next

• Privacy models
• Settings
• Case studies
Privacy Model: k-anonymity [Sweeney02]

• Output perturbation to a database: each row is the same with at least k-1 other rows.
ε-Differential Privacy [Dwork06]

Given a database $S$, return a query $f(S)$ such that for any database $S'$ that differs from $S$ by one element, $\Pr[f(S) \text{ in } A] \leq e^\varepsilon \Pr[f(S') \text{ in } A]$, for any $A$ in $\text{img } f()$.

Example: total salary of $S$?
Return: $\text{TS}(S) + \text{Lap}(\Delta f/\varepsilon)$
K-anonymity vs differential privacy

- Data publication
- Protects data & ID association
- Weaker protection
- NP-hard to minimize # changes
- Inference attack

- Data is not published
- Interactive query
- Protects the data itself
- Privacy loss w. # queries.
- Noise added can be high.
Ex: Location-based queries

• Where is the closest coffee shop?

• Protecting location & ID association: spatial “cloaking”.
Ex: Location-based queries

• Where is the closest coffee shop?

• Protecting location & ID association: spatial “cloaking”.

• Protecting location itself: add perturbation.
Location/Trajectory Collection Settings

- Location/trajectory collected by GPS and stored on the device.
  - Users voluntarily contribute such data.

- Wireless devices leave traces behind.
  - Cell towers.
  - WiFi AP.
Privacy Preserving with Sensing

1. Collect data;
2. Run anonymization;
Or, answer statistical queries with privacy added.

1. Collect **little** data
2. Derive group behaviors or statistical patterns.
One Network Setting; Two Case Studies

• Smart city environment: many checkpoints that record user mobility.
• What shall be collected at these checkpoints?
  • Low cost, w/ privacy protection.

1. Distributed trajectory clustering.
2. Popular path mining and query.
Part I: Trajectory Clustering
Clustering Mobile Nodes with k-anonymity

Static r-gather problem:
• each cluster has at least r nodes
• the maximum radius of the cluster is minimized.

Metric setting [Aggarwal et al., Armon]:
• r>2, NP-hard to app better than 2.
• Alg w/ 2-approx. using network flow
• r=2, in P, matching.
Clustering Mobile Nodes with k-anonymity

Static r-gather problem:
• each cluster has at least r nodes
• the maximum radius of the cluster is minimized.

Euclidean setting [MobiHoc17]:
• r>2, NP-hard to app better than 1.932 for max diameter, and 1.802 for minimum enclosing ball radius.
Clustering Mobile Nodes with k-anonymity

- Offline clustering: r-gather for trajectories
- m regroupings: dynamic programming.

- Kinetic clustering: smoothly reorganize the nodes into clusters of size at least \( r \).
Kinetic r-gather

1. Compute r-NN graph.
2. Find maximal independent set
3. Assign remaining nodes to nearest cluster.

• 4-approximation.
• # changes: $O(n^2)$ for poly motion.
• Distributed algorithm.
Clustering by Topology

(a) 4 trajectories with different homology types

(b) Trajectory flow
Clustering by Topology
Clustering by Topology
Homotopy Type
Homotopy Type
Homotopy vs Homology

- **Homotopy:**
  - One can deform a curve to another continuously;
  - A cycle can shrink to a point.
  - Stronger notion.

- **Homology:**
  - The ‘order’ or ‘orientation’ does not matter.
  - Easier to compute.
Sensing Homotopy/Homology Types

• Considers sensors densely exist in the environment tracking nearby targets.

• Goal: cluster the trajectories into homologous types.
  • Local, in-network processing.
  • Low cost in computation/communication.
  • Distributed.
Sensing Homotopy Types

- 2D domain with holes.
- Cut the domain open as simply connected.
- Trajectory is represented by how they go through the cuts.
- Simplification: $a + a - a + = a^+$
Differential 1-Form

- Planar graph G with **faces**.
- One-form: “**directed**” weights \( f \) on edges.
- Dual graph \( G' \): face \( \rightarrow \) vertex; vertex \( \rightarrow \) face; edges rotated by 90°.
Harmonic 1-Form

1. **Divergence-free**: $\sum_{\text{neighbor } v} f(u, v) = 0$
   i.e., no sources, no sinks

2. **Curl free**: $\sum_{\text{edge } e \text{ on a face}} f(e) = 0$
   i.e., divergence-free in dual graph
Use Harmonic 1-form

• For a cycle not enclosing any hole, the integration of the harmonic 1-form is zero.

• **Preprocessing**: Compute a harmonic 1-form on the graph s.t. only cycles enclosing holes integrate to non-zero values

• **Homology check**: Simple integration along the trajectories.

• Distributed storage & computation.

• How to compute a harmonic 1-form? By Hodge decomposition.
Hodge Decomposition

• Start w/ an arbitrary 1-form $\omega$.
• Hodge decomposition
  $$\omega = \alpha + \beta + \gamma$$
• $\alpha$: gradient flow, $\alpha(u, v) = \tau(u) - \tau(v)$, $\tau$ is a potential function on vertices, 0-form.
• Operation $\delta$: Integration along a face
  $$= \tau(u_1) - \tau(u_2) + \tau(u_2) - \tau(u_3) + ...$$
  $$+ \tau(u_k) - \tau(u_1).$$
  $$= 0$$
Hodge Decomposition

• Hodge decomposition
  \[ \omega = \alpha + \beta + \gamma \]

• \( \beta: \) curl flow, i.e., gradient flow in the dual graph, \( \beta(u, v) = \eta(x) - \eta(y) \), \( x \) is the face to the right, \( y \) is the face to the left. \( \eta \) is a function on faces, 2-form.

• Operation \( d: \sum \beta \) on edges of vertex \( u \)
  \[ = \sum \beta \text{ dual edges on face } u^* \]
  \[ = 0 \]
Hodge Decomposition

• Hodge decomposition
  \[\omega = \alpha + \beta + \gamma\]
• \(\gamma\): harmonic 1-form.
• Integration along a face = 0 (curl-free)
• Integration on edges of a vertex = 0 (divergence-free)
Gossip-style Implementation

- Goal: find **0-form** $\tau$ and 2-form $\eta$.
- $d$: Integration of the edges of a vertex
  
  \[ d\omega = d\alpha + d\beta + d\gamma \]
  
  \[ \sum w(e) = \sum_{(u, v)} \tau(u) - \tau(v) \]
  
  \[ \tau(u) = \frac{\sum w(e) + \sum_{(u, v)} \tau(v)}{d(u)} \]

- Initialize all $\tau(u) = 0$
- Run gossip with neighbors.
Gossip-style Implementation

• Goal: find 0-form $\tau$ and 2-form $\eta$.
• $\delta$: Integration along a face $f$
• $\delta \omega = \delta \alpha + \delta \beta + \delta \gamma = \delta d \tau$
• $\sum_{e \text{ on } f} w(e) = \sum_i \eta(f) - \eta(f_i)$
• $\eta(f) = [\sum_{e \text{ on } f} w(e) + \sum_i \eta(f_i)]/d(f)$

• Initialize all $\eta(f) = 0$
• Run gossip with neighbors.
Homology Basis

• Harmonic 1-forms form a linear space of dim k, for k holes, or 2g for a closed surface with genus g.

• Linear dependency can be checked locally.

• Homology signature of a trajectory: k-vector integration along k harmonic 1-forms.
Practical Considerations

• Homology test: Integration of a cycle is sufficiently close to zero.
• Two trajectories are homologous if they integrate to the same values.
Vehicle Trajectories

• 243 trajectories in a city.

Table III. Descriptive nature of homology types.

<table>
<thead>
<tr>
<th>#holes</th>
<th>#homology types</th>
<th>max. # trajectories in the same type</th>
<th>#trajectories with unique value</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>41</td>
<td>48</td>
<td>21</td>
</tr>
<tr>
<td>5</td>
<td>105</td>
<td>26</td>
<td>69</td>
</tr>
<tr>
<td>7</td>
<td>146</td>
<td>22</td>
<td>119</td>
</tr>
</tbody>
</table>
Part II: Traffic Pattern Query
Popular Paths

• A path travelled by $\phi$-fraction of all vehicles that appear on the path.
  
  • A subpath of a popular path is still popular;
  
  • A node stays on at most $1/\phi$ maximally popular paths.
MinHash Signature

• Sensor i sees a set of vehicles $V$, and stores the min hash value $h_i(V)$, for $k$ hash functions.
MinHash Signature

- MinHash estimates set cardinality.
- Minhash estimates path popularity by Jaccard coefficient:

\[ J(A, B) = \frac{|A \cap B|}{|A \cup B|}. \]

- MinHash estimates \( J(A, B) \): \( X/k \), \( X \): \# minhash values A, B agree with.
MinHash Signature

- # common MinHash entries along a path estimates path popularity.
MinHash and Privacy

- If two sets of trajectories differ by 1, with good chance their signatures are the same, upon randomness of the seeds.

\[
\Pr\{S(D) = S^*\} \leq e^{\varepsilon} \Pr\{S(\tilde{D}) = S^*\},
\]

\[
\varepsilon = km/n'.
\]

# checkpoints # vehicles each node has seen
MinHash Hierarchy

• Recursively subsample checkpoint.
• Edge (u, v): if there is at least one popular path from u to v
Traffic Pattern Queries

• By careful search in the hierarchy of m nodes.
  • Popular paths for (s, t) – $O(\log m)$
  • Popularity for a path P. – $O(\log m)$
  • All popular paths from s. - $O(\log^2 m)$
Summary

• Sensing with privacy consideration.
• Reduced communication cost.
Questions & Comments?

- [http://www.cs.stonybrook.edu/~jgao](http://www.cs.stonybrook.edu/~jgao)

