Tracking Groups in Mobile Network Traces

Kun Tu*, Bruno Ribeiro**, Ananthram Swami***, Don Towsley*
*University of Massachusetts, Amherst
**Purdue University
***Army Research Lab

Presented by Gayane Vardoyan

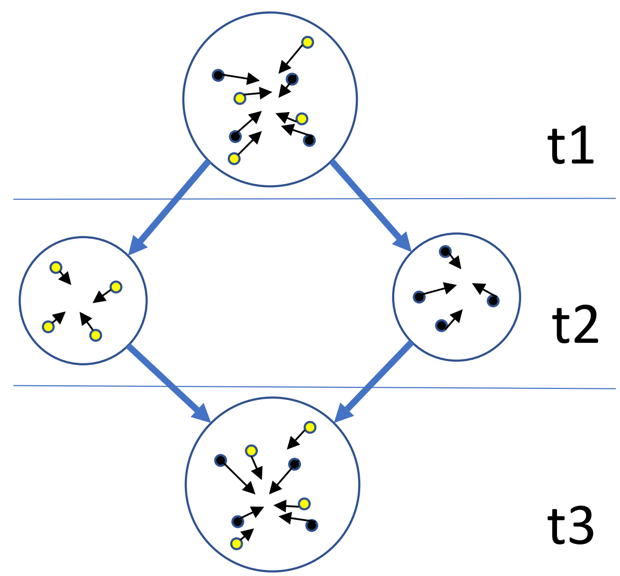
Groups in Mobile Network Trace

- Most mobility models assume independent movements
- Several ad hoc mobility models
  - Random direction, waypoint model
  - Leader based group models

Q: what is a realistic group mobility model?

Answering question requires obtaining group information from mobility data

How to do so – focus of talk

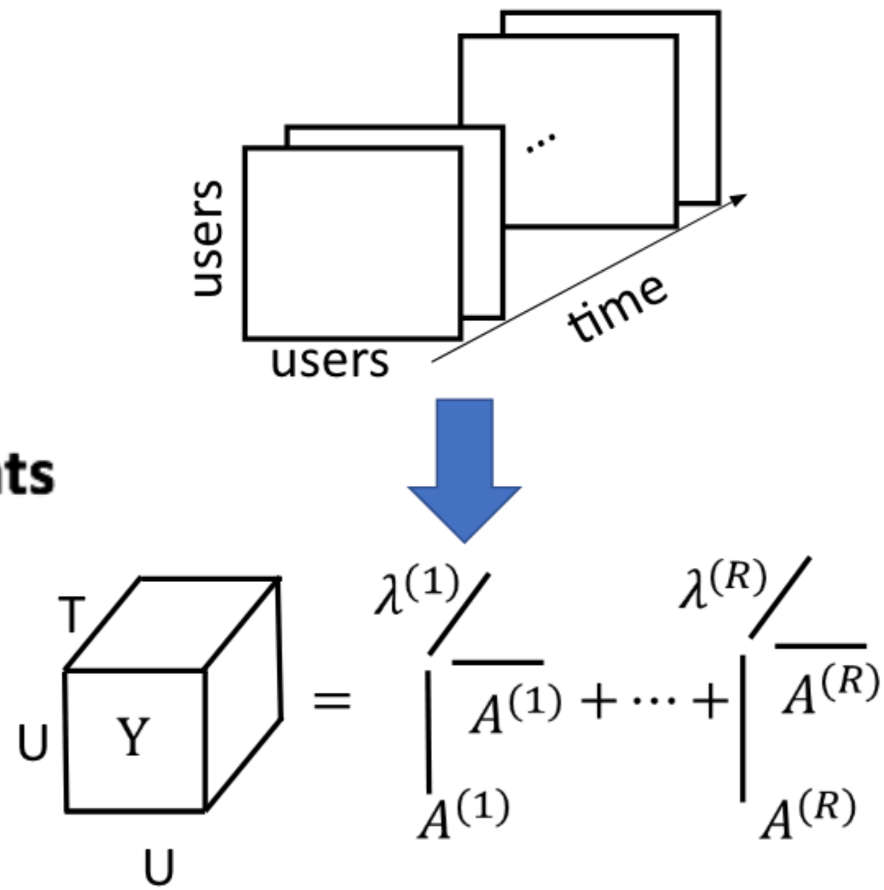


Outline

- Model and problem formulation
- Tensor decomposition
- Extracting group information from tensor components
- Experiments
- Conclusion

Idea

- Represent dataset as 3-D tensor, Y
  - Snapshots over time
  - Snapshot: adjacency matrix, Euclidean distances
- Decompose tensor into R **components**
- From component r
  - Identify groups from $A^{(r)}$
  - Identify group formation, dissolution times from $\lambda^{(r)}$



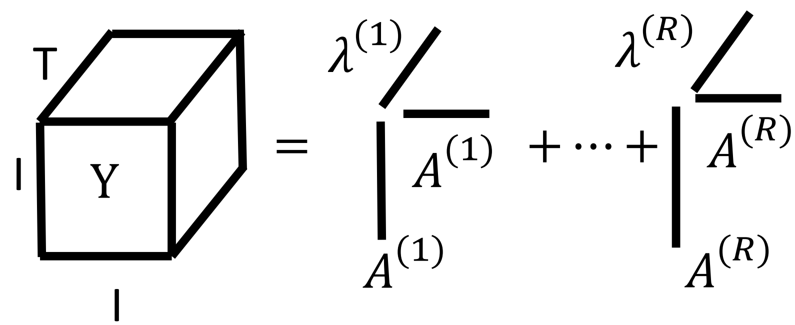
Challenges

- Time granularity of snapshots
  - Fine time scale: sparse snapshot, difficult for group detection
  - Coarse time scale: loss of detailed changes, resulting in high error for lifetime detection
- Tracking changes in groups
  - creation/dissolution
  - changes in group composition
  - membership in multiple groups

Our model

- Tensor $\boldsymbol{Y} = [Y_{ijt}]$, Y_{ijt}- closeness of user i to user j at time t
- Approximate Y_{ijt} by R components

$$\hat{Y}_{ijt} = \sum_{r=1}^{R} a_{ir} a_{jr} \lambda^{(r)}(t)$$

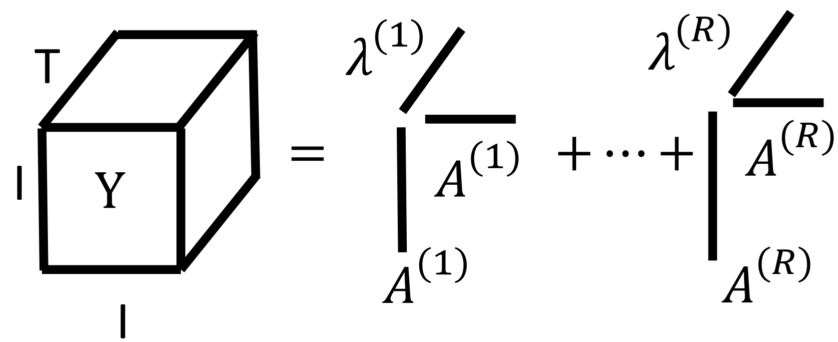


  - $a_{ir} \in A^{(r)}$: probability of user i in component r
  - $\lambda^{(r)}$: time series representing node similarities at different time steps

Our Model

- Tensor $\boldsymbol{Y} = [Y_{ijt}]$, Y_{ijt}- closeness of user i to user j at time t
- Approximate Y_{ijt} by R components

$$\hat{Y}_{ijt} = \sum_{r=1}^{R} a_{ir} a_{jr} \lambda^{(r)}(t)$$



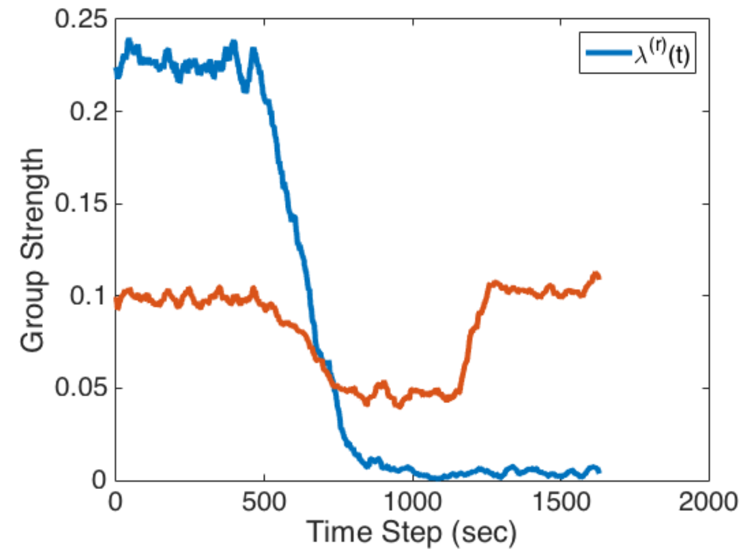
- $a_{ir}, \lambda^{(r)}(t)$ obtained from minimizing

$$\sum_{i,j \in V} \sum_{t} (Y_{ijt} - \sum_{r} a_{ir} a_{jr} \lambda^{(r)}(t))^2$$

- Use alternating least squares algorithm to solve
  - gradient descent method to compute a_{ir} and $\lambda^{(r)}(t)$ iteratively

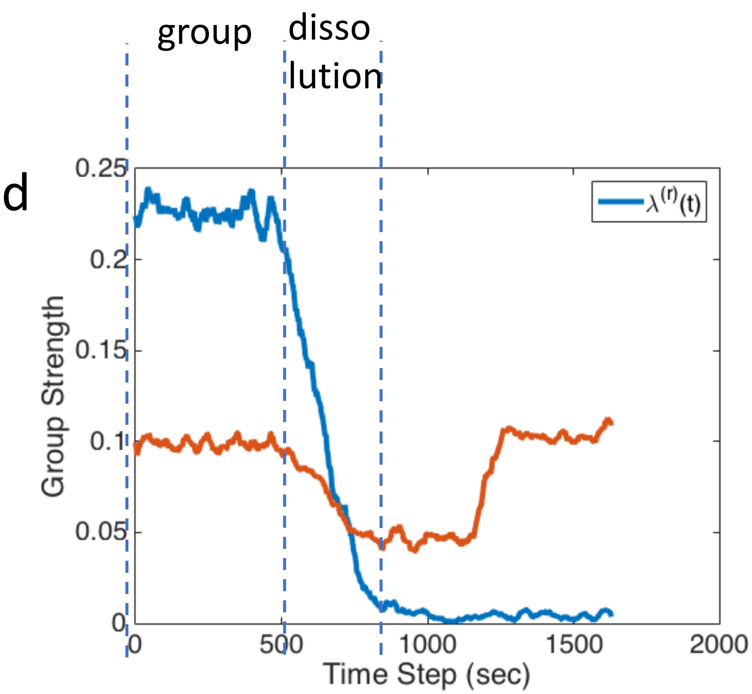
Interpretation

- Use K-means to find group(s) in $A^{(r)} = [a_{ir}]$
  - silhouette clustering criterion used to choose number of groups
- Temporal mode $\lambda^{(r)}(t)$ represents strength of group
- When R chosen properly, one meaningful group per component
- If not, can order groups according to strength using similarity ordering score



Group Lifetime Detection

- $\lambda^{(r)}(t)$ as a time series
- Compare against adaptive threshold based on average similarity
  - above – formation of group
  - below – no group
- Can detect formation, dissolution times

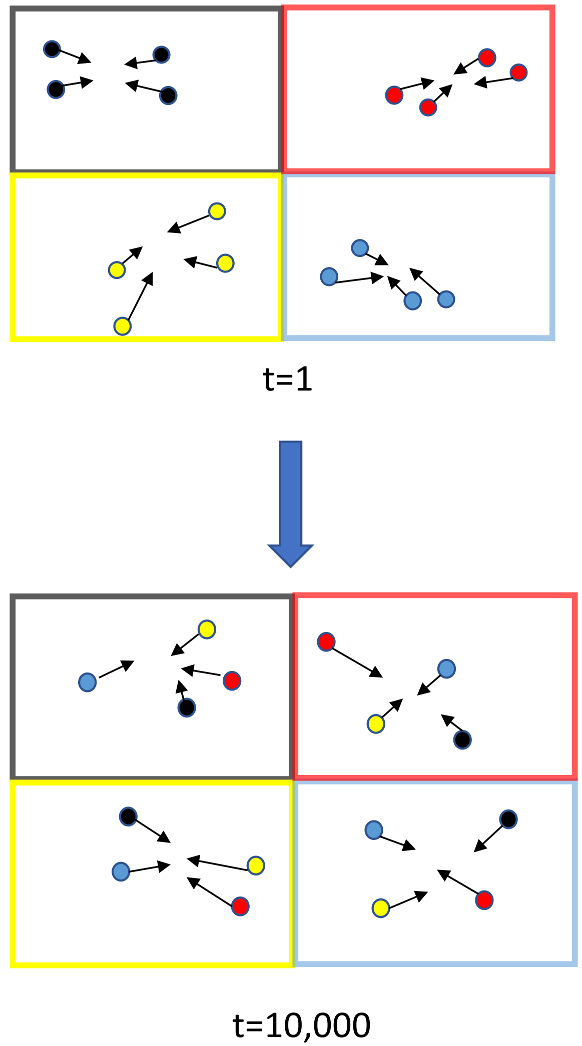


Experiments

- Synthetic datasets
- Lakehurst dataset
  - Military training exercise

Synthetic Dataset

- 400 nodes in 4 initial groups move according to random direction model (RD) for 10,000 seconds
- Each group divides into 4 subgroups, subgroups move to different areas, form new groups
- 1000 repetitions, different parameter settings



Group member detection

- $\hat{C}_k$: set of members in k-th detected group
- C_k^*: set of members in ground truth group mapped to $\hat{C}_k$ using Jaccard index (intersection of two sets over their union)
- Precision: $P = \frac{|\hat{C}_k \cap C_k^*|}{|\hat{C}_k|}$
- Recall: $\mathrm{R} = \frac{|\hat{C}_k \cap C_k^*|}{|C_k^*|}$
- F1 score: $\mathrm{F}_1 = \frac{2PR}{P+R}$
- Precision Recall Curve (PR-curve): evaluate precision of methods when recall is similar
- Similar metrics for group lifetime: F1 score

Baseline methods:

Evolutionary Clustering (EC) (Deepayan et al., 2006)

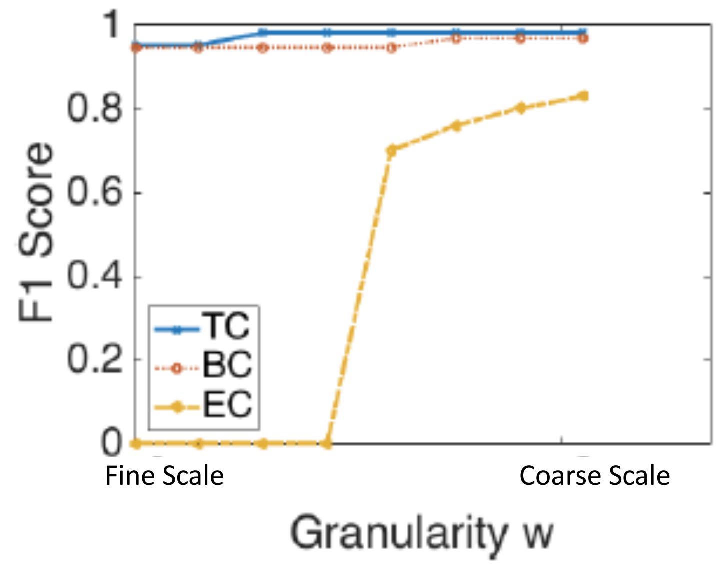
- Clustering on each network snapshot
  - Pros: fast
  - Cons: fails in multi-membership, sparse network, tracking cluster changes

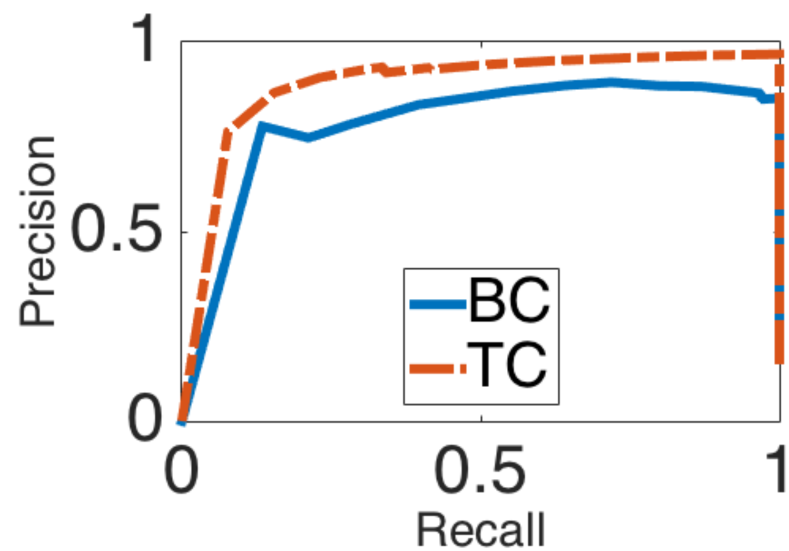
Binary clustering (BC) (Laetitia et al., 2014)

- Detect cluster on tensor factorization result with fixed threshold
  - Pros: work for multi-membership, sparse network, tracking lifetime
  - Cons: difficulty in fine tuning # groups leads to high detection error

Group Member Detection

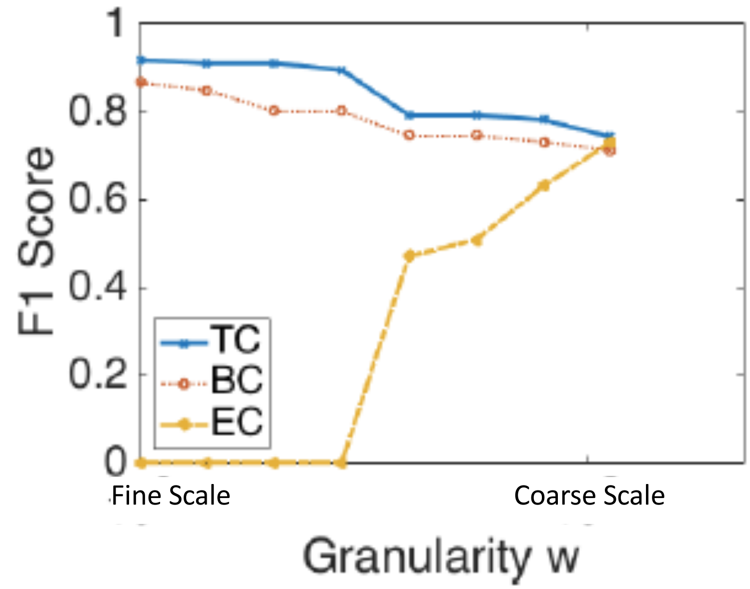
- Effect of time granularity (*w)*
  - Proposed method temporal clustering (TC) and BC robust to time granularity
  - EC works poorly with fine granularity
- TC has better precision than BC given same recall

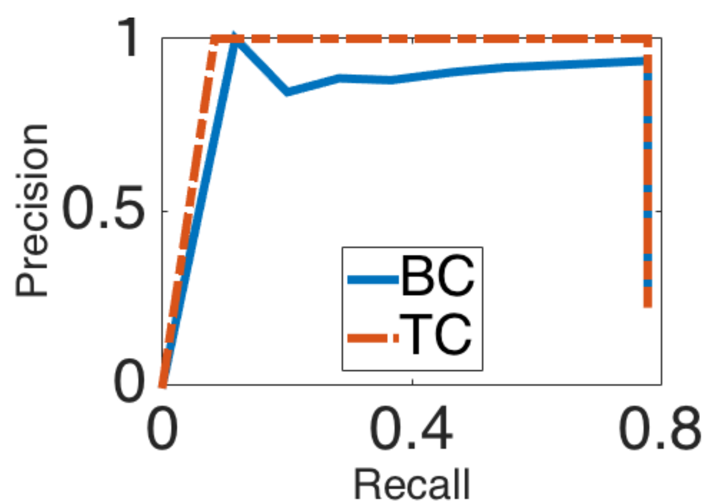




Lifetime Detection

- Coarse granularity
  - reduces accuracy of TC, BC
  - improves EC performance because of increased accuracy in member detection
- TC has better precision than BC given same recall





Summary for synthetic data

- Our temporal clustering method (TC)
  - Is robust to change in time granularity in member detection
  - Performs as well as BC and better than EC

Lakehurst Military Dataset

- Three hour trace, 70 vehicles
- 64 vehicles split into 9 platoons
- Another six vehicles move separately
- Platoons combine to form large group from time to time
- 19 groups total

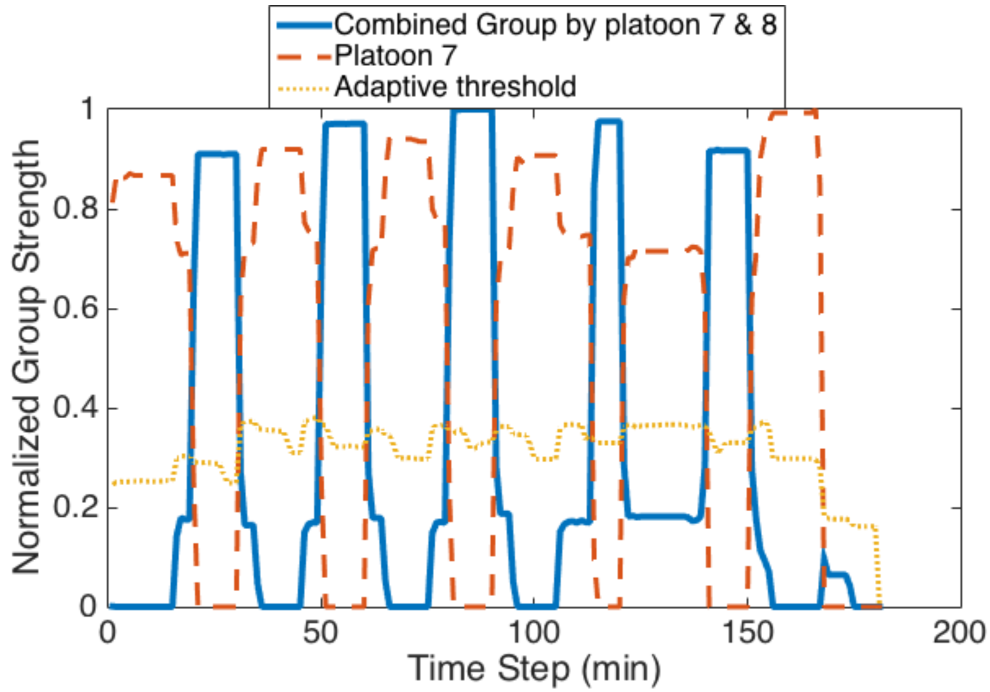
Lakehurst dataset Results

- TC performs as well or better than other methods
- Large R improves recalls for TC and BC

# component (R)	10	15	20	25	30
TC Group Recall	**0.368**	0.421	**0.587**	0.895	1.0
BC Group Recall	0.319	0.421	0.579	0.895	1.0
TC Member Recall	0.430	**0.541**	0.841	**0.875**	0.904
BC Member Recall	0.430	0.532	0.841	0.862	0.904
EC Group Recall	0.474				
EC Member Recall	0.275				

Group Lifetime Behavior

- Lifetime tracking for a group
  - Formed by platoon 7 and platoon 8 who meet at multiple waypoints
  - Formation & dissolution with time series segmentation algorithm
  - Detect lifetime using adaptive threshold (average similarity of nodes of whole network)
- Tensor time mode facilitates lifetime identification



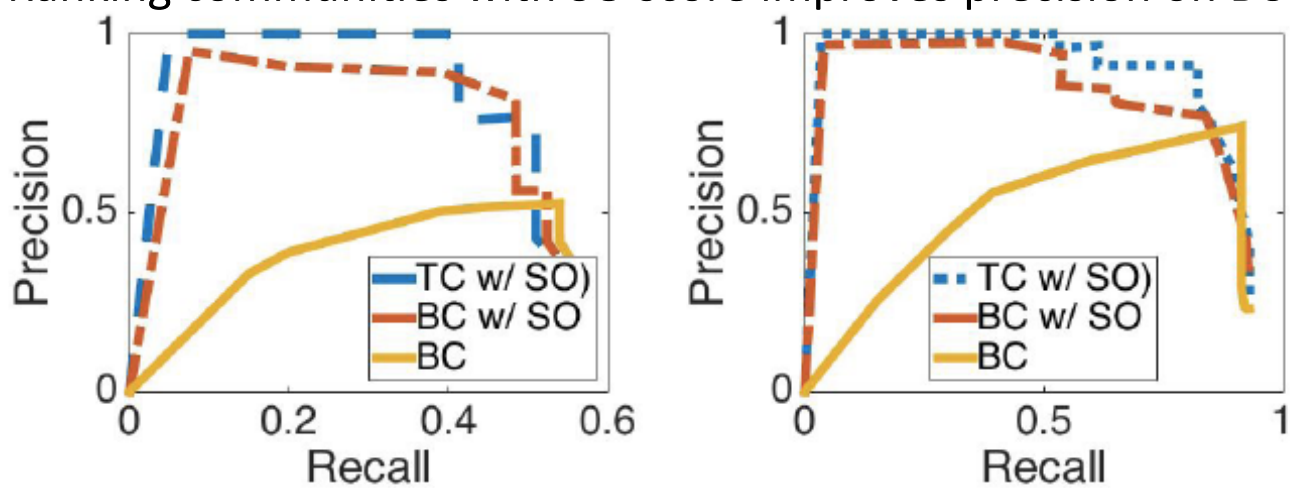
Conclusion

- Proposed temporal clustering method to detect groups in mobile trace data
- Method
  - detects multi-membership of individuals
  - robust to changes in time granularity
  - automatically determines number of groups
- Proposed method more accurate than previous methods
- Future directions
  - Model can be applied to directed temporal networks representing relations between users, location and time.

Thank you

Group Member Detection in Lakehurst

- TC has better performance measured by PR curve given different value of hyperparameter R (number of groups)
  - BC has poor precision given same Recall
  - Ranking communities with SO score improves precision on BC



(a) $R = 15$ (b) $R = 30$