Trajectory Pattern Mining

Fosca Giannotti et al,.

- Presented by Shuo Miao

Conference on Knowledge discovery and data mining, 2007

- OUTLINE

1. Motivation
2. T-Patterns: definition
3. T-Patterns: the approach(es)
    - Regions-of-Interest approach
    - RoI extraction
    - Step-wise refinement of RoI
4. Experiments
5. Conclusion

Motivation - 1

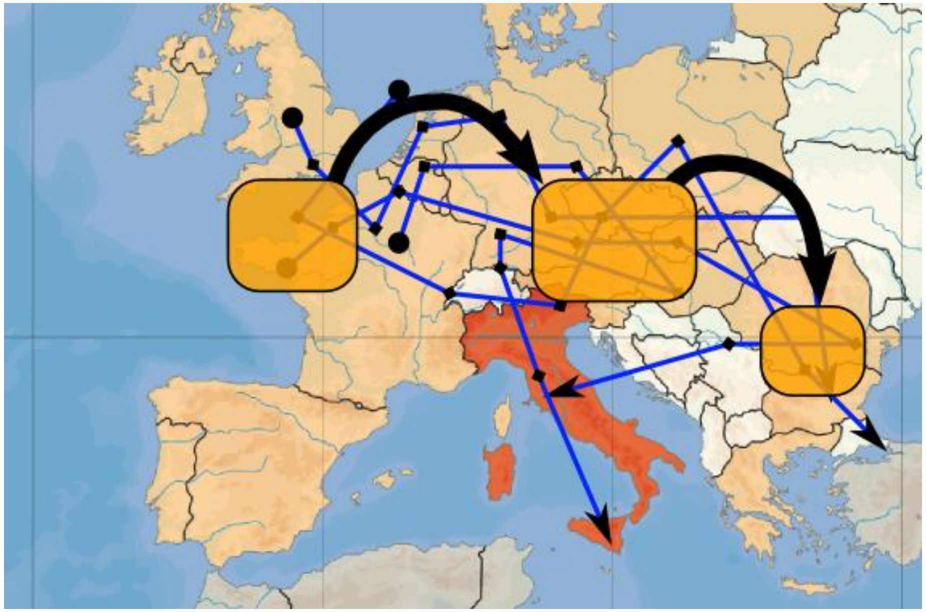
- Large diffusion of mobile devices, mobile services and location-based services

- **Motivation - 1**
  - A bit more about GPS
  - **Galileo** is a global navigation satellite system (GNSS) currently being built by the European Union (EU) and European Space Agency (ESA). The €3.4 billion project is an alternative and complementary to the **U.S. Global Positioning System (GPS)** and **the Russian GLONASS**. On 30 November 2007 the 27 EU transportation ministers involved reached an agreement that it should be operational by 2013 ---- from Wikipedia
  - Galileo provides more **accuracy** and **Altitude**

- **Motivation – 1**
  - About Accuracy
    - About 90% confidence within 15 meters in an open sky
    - Error from Timing, satellites, and many other factors

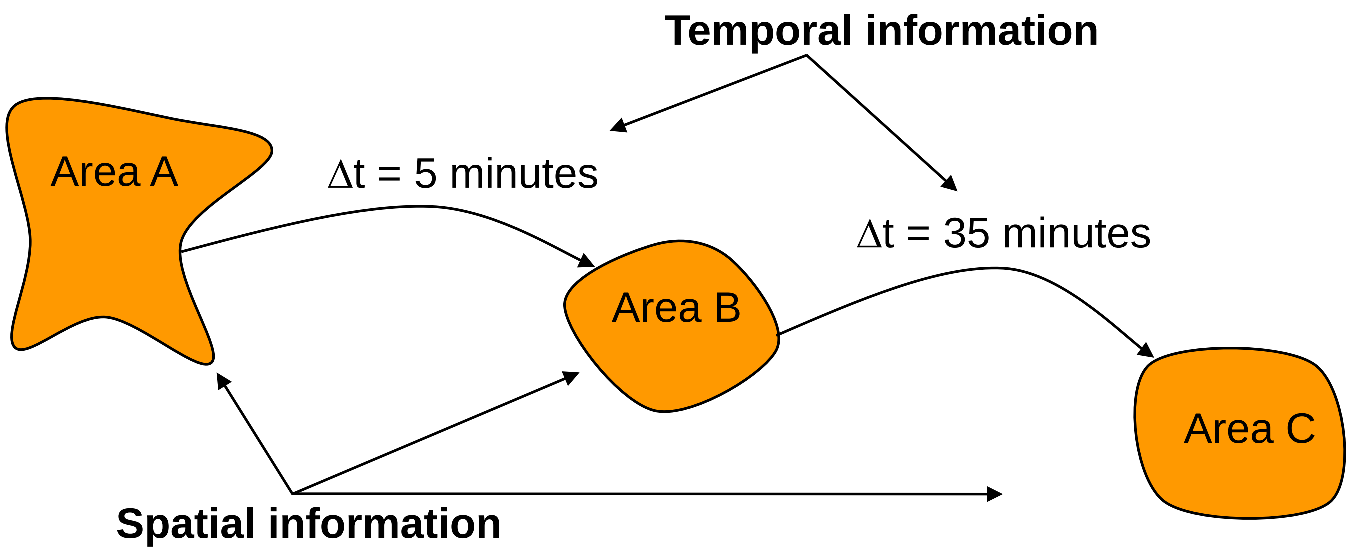
      http://www.gpsnuts.com/myGPS/GPS/Technical/gps_receiver_accuracy_by_c.htm
    - Wide Area Augmentation System (WAAS) is an air navigation aid developed by the Federal Aviation Administration to augment the Global Positioning System (GPS), with the goal of improving its accuracy, integrity, and availability.
    - WAAS send Deviation Correction → WAAS satellites → GPSs
    - WAAS: < 3 meters, 95% typical

- **Motivation - 2**
- Such devices leave digital traces that can be collected to for trajectories describing the mobility behavior of its owner
- From this large amount of data, high level information should be extracted, e.g., patterns describing mobility behaviors



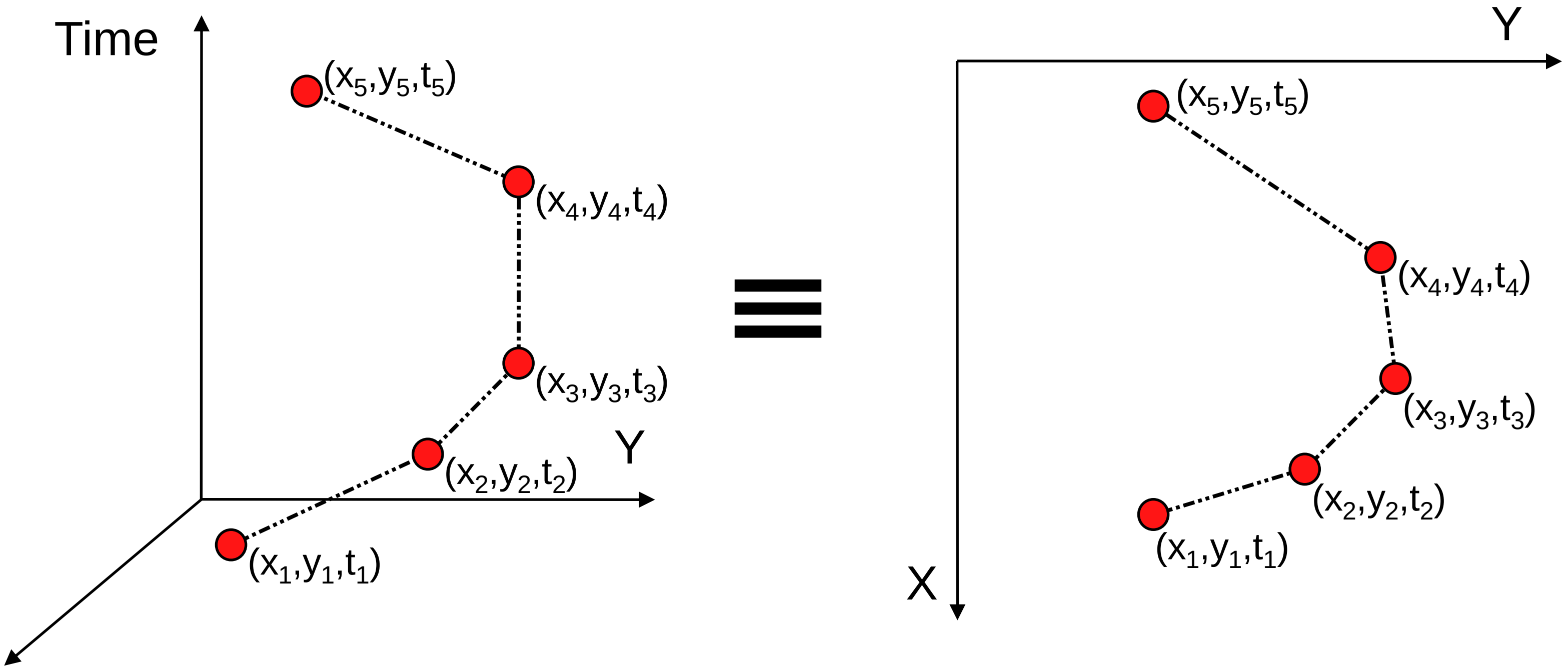
Sequential patterns for trajectories

- Question: what should a sequential pattern about moving objects look like?
  - it should describe their movements in space and in time



Sequential patterns for trajectories

- Trajectories are usually given as *spatio-temporal (ST) sequences:* $<(x_1,y_1,t_1), \ldots, (x_n,y_n,t_n)>$



T-Patterns for trajectories

- A set of individual trajectories that share the property of visiting the same sequence of places with similar travel times.
- A **Trajectory Pattern** (T-pattern) is a couple **(s, α)**:
  - **s** = <(x_0,y_0),…, (x_k,y_k)> is a sequence of k+1 locations
  - **α** = <α_1,…, α_k> are the transition times (*annotations*)

also written as: $(x_0, y_0) \xrightarrow{\alpha_1} (x_1, y_1) \xrightarrow{\alpha_2} \cdots \xrightarrow{\alpha_k} (x_k, y_k)$

- **A T-pattern T_p occurs** in a trajectory if it contains a sub-sequence S such that:
  - each (x_i,y_i) in T_p ***matches*** a point (x_i',y_i') in S, and
  - the transition times in T_p **are similar to those in S**

Continuity issues

- The same exact spatial location (x,y) usually never occurs twice
  - yet, close locations essentially represent the same place, so they should match
- The same exact transition times usually do not occur often
  - same as above
- Solution: allow approximation
  - a notion of *spatial neighborhood (Fuction)*
  - *a notion of temporal tolerance (Threshold)*

T-Pattern: *approximate* occurrence

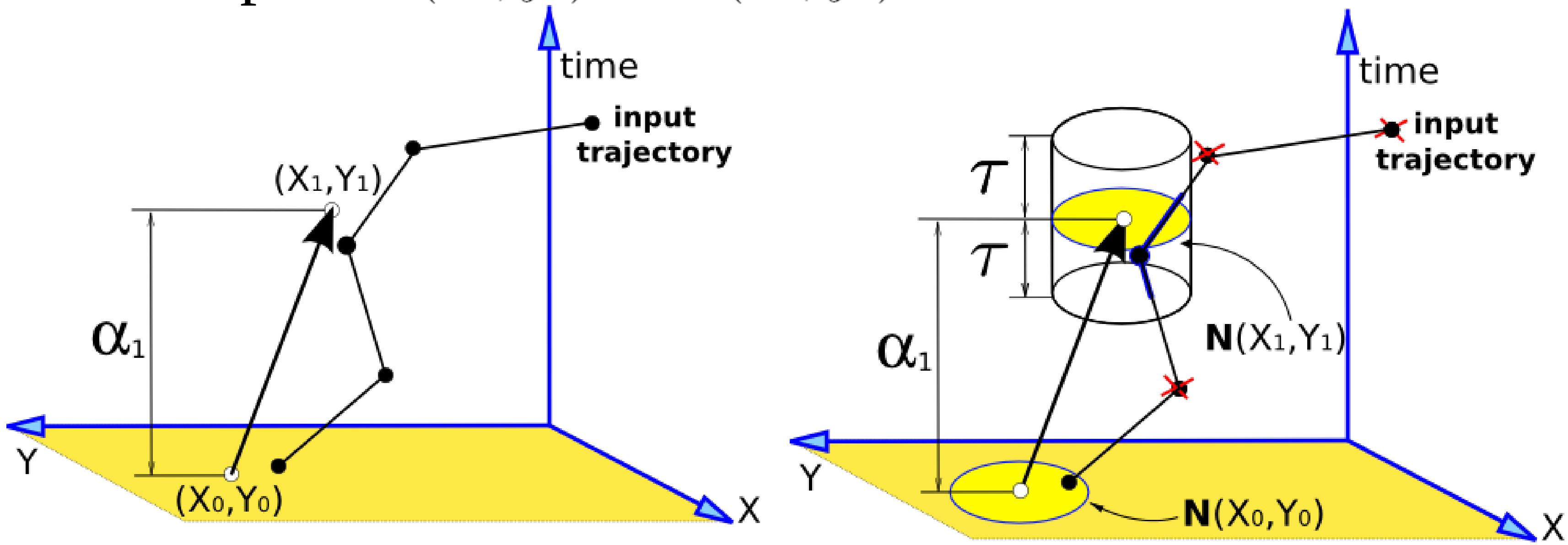
- Given a set of *Regions of Interest R*, define the neighborhood of (x,y) as:

$$N_R(x,y) = \begin{cases} A & \text{if } A \in R \;\&\; (x,y) \in A \\ \varnothing & \text{otherwise} \end{cases}$$

  - Neighbors ⇔ belong to the same region
  - Points in no region have no neighbors

T-Pattern: *approximate* occurrence

- Two points match if one falls within a **spatial neighborhood N()** of the other
- Two transition times match if their **temporal difference is ≤ τ**
- Example: $(x_0, y_0) \xrightarrow{\alpha_1} (x_1, y_1)$

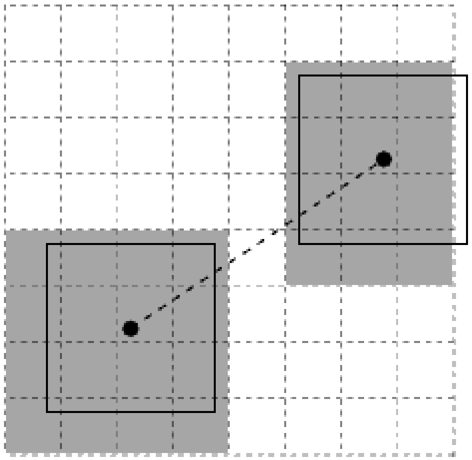


Static REGIONS-OF-INTEREST

- A priori knowledge of suitable Regions-of-Interests
  - manually obtained by experts in the application domain
  - or simply through commonsense.
- What if RoI are not known as a priori
  - define heuristics for automatic RoI extraction from data
    - Geography-based (e.g., crossroads)
    - **Usage-based (e.g., popular places)**
    - Mixed (e.g., popular squares)

Calculate Density

- Apply grids into space
- density of cells is computed by taking each single trajectory and incrementing the density of all the cells



Calculate Region of Interest

Algorithm: PopularRegions($\mathcal{G}$,δ)

Input: A grid $\mathcal{G}$ with densities $\mathcal{G}(i,j)$, a density threshold δ

Output: A set R of rectangular regions over $\mathcal{G}$.

$R = \emptyset$; $\quad \mathcal{G}^* = \{(i,j) \in \mathcal{G} | \mathcal{G}(i,j) \geq \delta\}$;
foreach $(i,j) \in \mathcal{G}$ **do** $used(i,j) = false$;
foreach $(i,j) \in \mathcal{G}^*$ in descending order of $\mathcal{G}(i,j)$ **do**
  if $\neg used(i,j)$ **then**
    $r = \{(i,j)\}$;
    repeat
      foreach $dir \in \{left, right, up, down\}$ **do**
        $r_{dir} = r$ extended on direction dir;
      $ext = \{\ dir |\ r_{dir} \subseteq \mathcal{G} \wedge \underline{avg\_density(r_{dir}) \geq \delta}$
        $\wedge\ \exists (h,k) \in (r_{dir} \setminus r).\mathcal{G}(h,k) \geq \delta$
        $\wedge\ \forall (h,k) \in r_{dir}.\neg used(i,j)\ \}$;
      if $ext \neq \emptyset$ **then**
        $dir = \underline{\arg\max_{d \in ext} avg\_density(r_d)}$;
        $r = r_{dir}$;
    until $ext = \emptyset$;
    foreach $(i,j) \in r$ **do** $used(i,j) = true$;
    $R = R \cup \{r\}$;
return R;

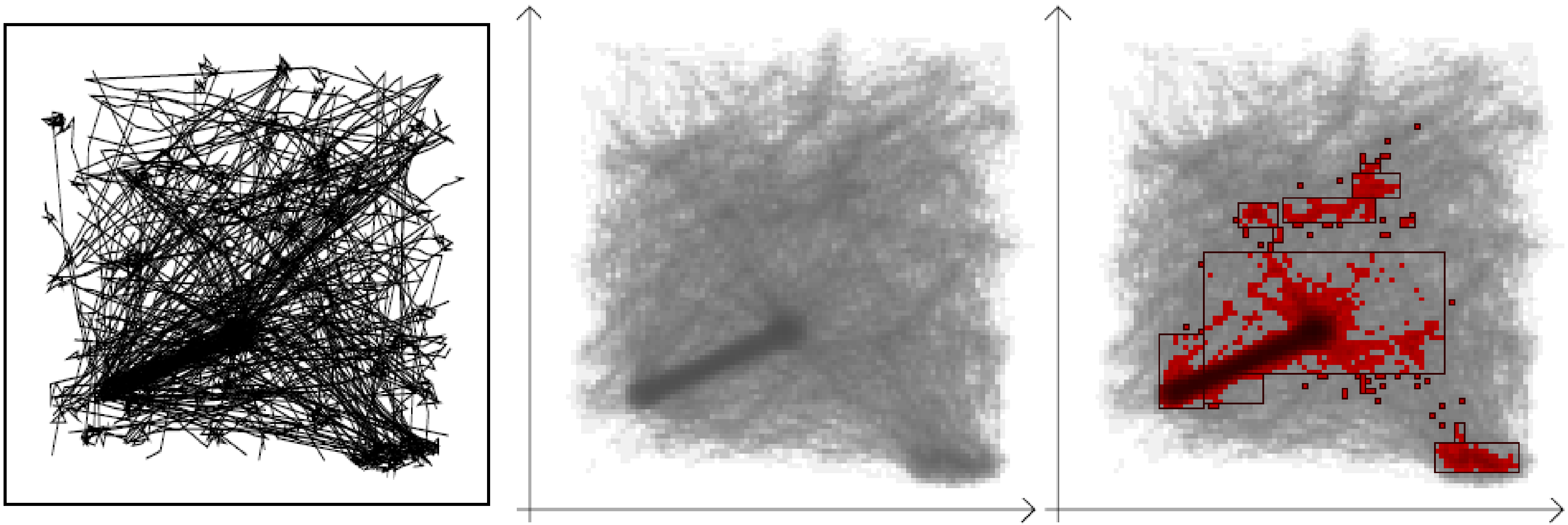
- start from densest cell
- consider any direction that (i) adds a dense cell, (ii) keeps avg density high, (iii) avoids overlap of regions
- select locally best direction

Minimum support threshold provided by user

Density threshold δ provided by user

A usage-based heuristic

1. Impose a regular grid over space
2. Find dense cells (i.e., touched by many trajs.)
3. Coalesce cells into rectangles of bounded size

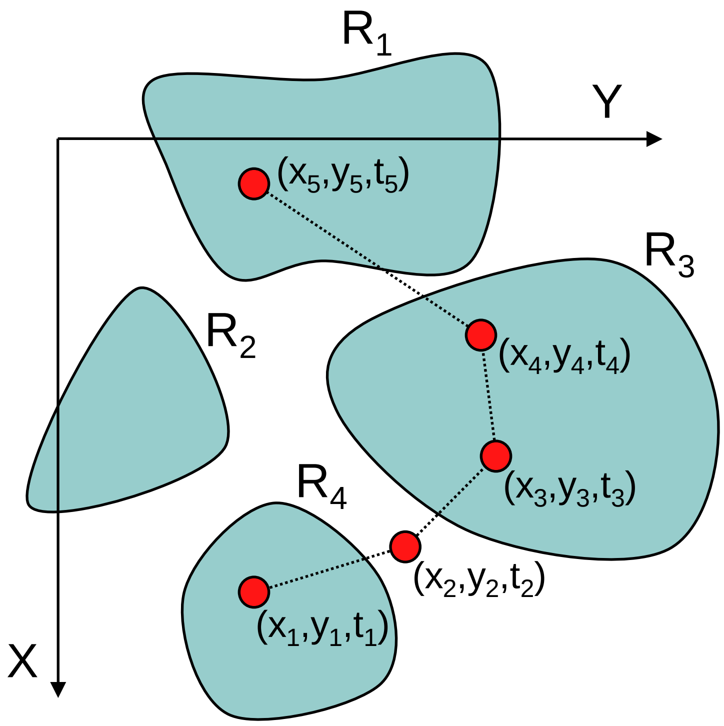


(a) input trajectories (b) density distribution (c) dense cells and extracted RoI

Translating ST-sequences

- a wide range of alternatives are available in the spatio-temporal data modeling literature
  - linear regression, Bezier's curves, probabilistic models, constraints models
  - linear regression is adopted
    - simplest and most frequently adopted model
- Associate with a time-stamp
  - if the trajectory starts at time t from a point already inside a region A, yield the couple (A, t);
  - in all other cases, take entering times of the trajectory for each region

Translating ST-sequences



$$S=\langle(x_1,y_1,t_1), \ldots, (x_5,y_5,t_5)\rangle$$

$$\langle(R_4,t_1), (R_3,t_3), (R_3,t_4), (R_1,t_5)\rangle$$

Algorithm

Algorithm: Static_RoI_T-pattern(T_{in}, $\mathcal{G}_0$, δ, ϵ, τ)

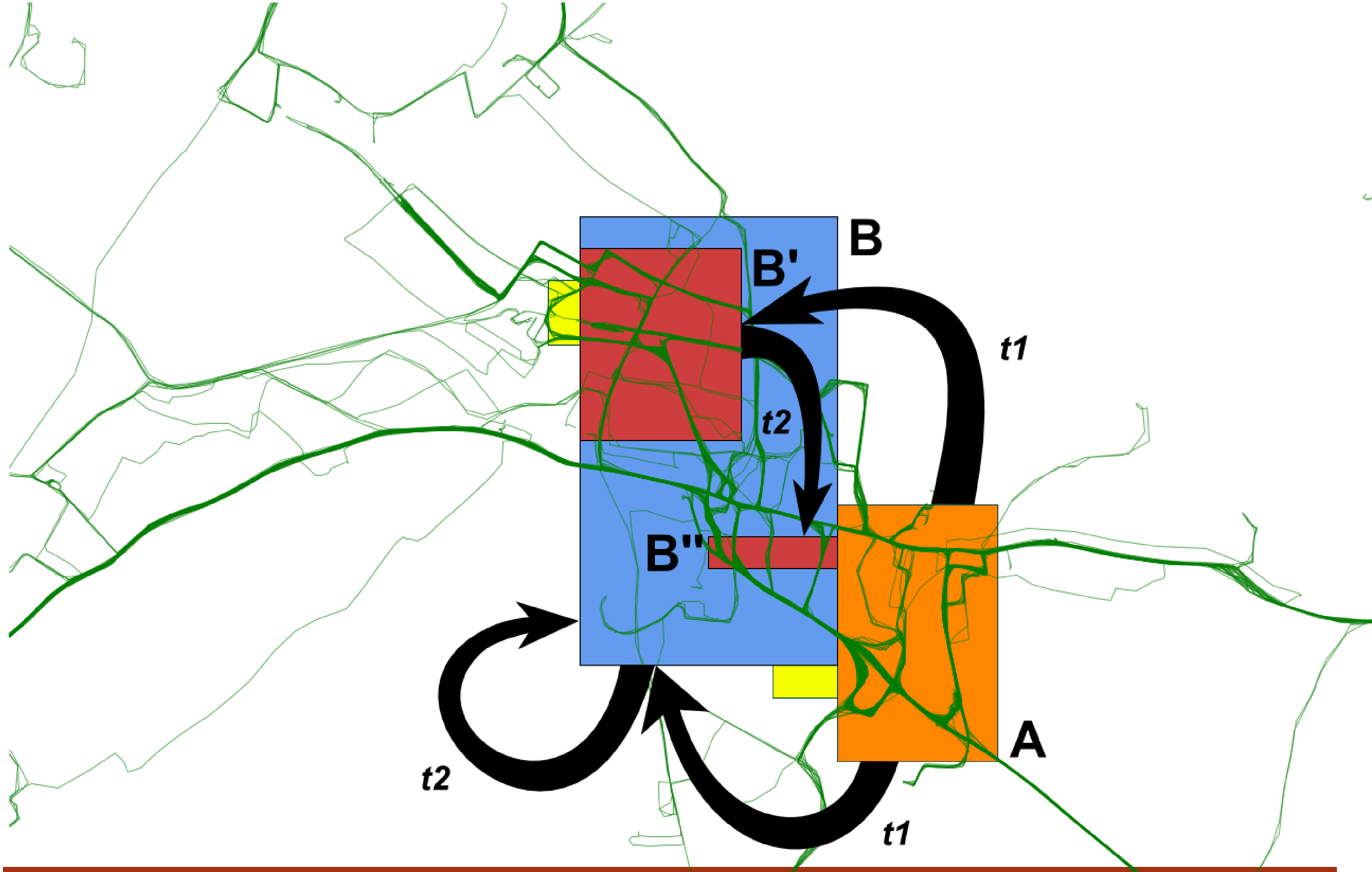
Input: A set of input trajectories T_{in}, a grid $\mathcal{G}_0$, a minimum support/density threshold δ, a radius for spatial neighborhoods ϵ, a temporal threshold τ.
Output: A set of couples $(S, \mathcal{A})$ of sequences of regions with temporal annotations.

1. $\mathcal{G}$ = ComputeDensity(T_{in},$\mathcal{G}_0$,ϵ); *(Sect. 4.2.1)*
2. RoI = PopularRegions($\mathcal{G}$,δ); *(Sect. 4.2.2)*
3. $\mathcal{D}$ = Translate(T_{in},RoI); *(Sect. 4.1)*
4. $\mathcal{TAS}$_mining($\mathcal{D}$, δ, τ); *([5])*

Figure 5: Mining frequent T-patterns with static Regions-of-Interests

- TAS : *temporally-annotated sequences*
  - *Efficient mining of sequences with temporal annotations*
  - extension of sequential patterns
  - *Only take time dimension into account*

Result



Dynamic REGIONS-OF-INTEREST

- Static RoI
  - Cells approximate single points, regions group points that are likely to form similar patterns
  - Yet, they should regard only trajectories that support the discovered pattern, not all database
- Towards general T-patterns
  - Check & update dense cells and regions of each pattern against the trajectories that support it
  - Approximation: Perform the update as step-wise refinement as patterns grow

Step-wise dynamic RoI

Algorithm: Dynamic_RoI_T-pattern(T_{in}, $\mathcal{G}_0$, δ, ϵ, τ)

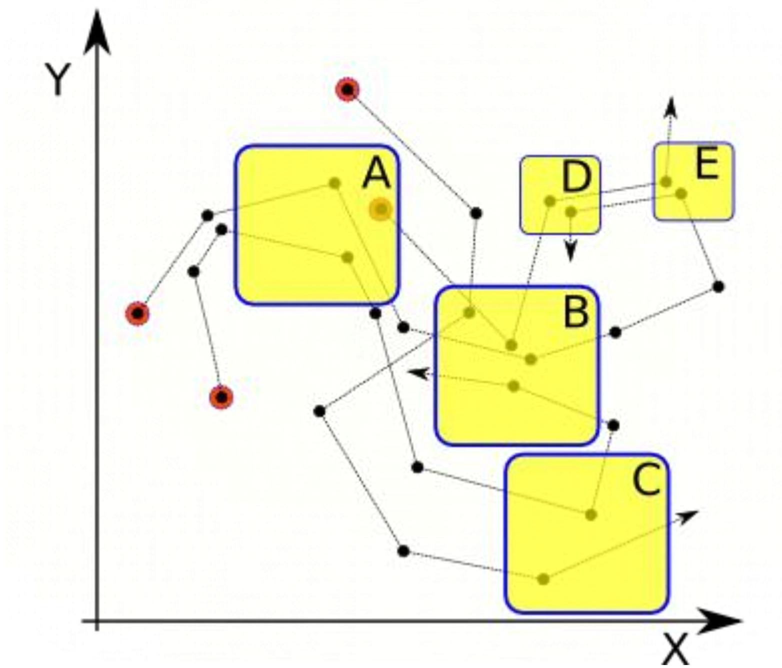
Input: A set of input trajectories T_{in}, a grid $\mathcal{G}_0$, a minimum support/density threshold δ, a radius for spatial neighborhoods ϵ, a temporal threshold τ.

Output: A set of couples $(S, \mathcal{A})$ of sequences of regions with temporal annotations.

$L = 0$; $T_0 = \{(T_{in} \times \{\emptyset\}, \langle\rangle)\}$;
while $T_L \neq \emptyset$ **do**
  $T_{L+1} = \emptyset$;
  foreach $(T, \mathit{prefix}) \in T_L$ **do**
    if $|\mathit{prefix}| \geq 2$ **then**
      $\mathcal{A}$ = ExtractFrequentTimings(T); *([5])*
      Output ($\mathit{prefix}, \mathcal{A}$);
      T = PruneEmptyAnnotations(T, $\mathcal{A}$); *([5])*
    $\mathcal{G}$ = ComputeDensity(T,$\mathcal{G}_0$,ϵ); *(Sect.4.2.1)*
    RoI = PopularRegions($\mathcal{G}$,δ); *(Sect.4.2.2)*
    $\mathcal{D}$ = Translate(T,RoI); *(Sect.4.1)*
    foreach $r \in \mathit{RoI}$ **do**
      if $\mathit{support}_{\mathcal{D}}(r) \geq \delta$ **then**
        $\mathcal{D}'$ =ExtendProjection($\mathcal{D}$,r); *([5])*
        $T' = \{\ (\mathit{traj}, \mathcal{A}')\ |\ (\mathit{traj}, \mathcal{A}) \in T$
          $\wedge\ (S', \mathcal{A}') \in \mathcal{D}' \wedge\ \mathit{traj}.ID = S'.ID$
          $\wedge\ \mathit{traj}' = \text{Cut}(\mathit{traj}, \mathcal{A}')\ \}$
        $T_{L+1} = T_{L+1} \cup \{(T', \mathit{append}(\mathit{prefix}, r))\}$;
  L++;

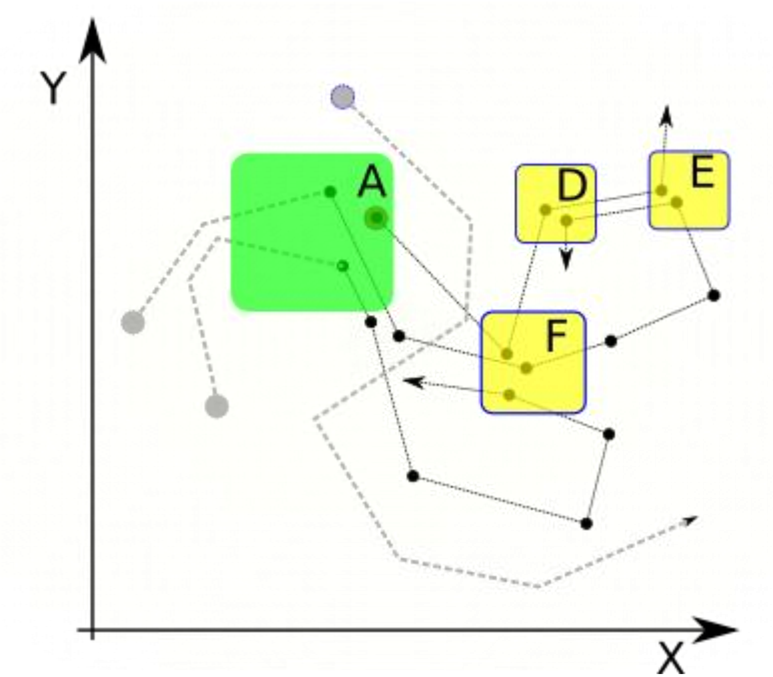
- Extract freq. transition times
- Compute up-to-date RoI
- Extend patters w.r.t. new RoI
- Focus on patterns found

Dynamic REGIONS-OF-INTEREST



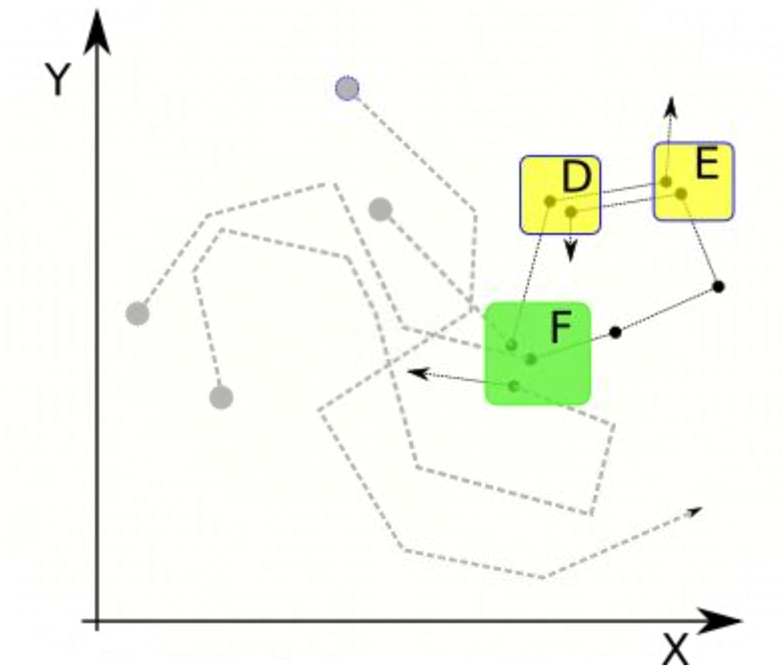
- Start computing regions as basic RoI approach
- Regions describe interesting places of *everybody*

Dynamic REGIONS-OF-INTEREST



- Focusing on A, we consider only the subset of relevant trajectories
- Regions can change (usually shrink/split)
- They are interesting only for who passes thru A

Dynamic REGIONS-OF-INTEREST

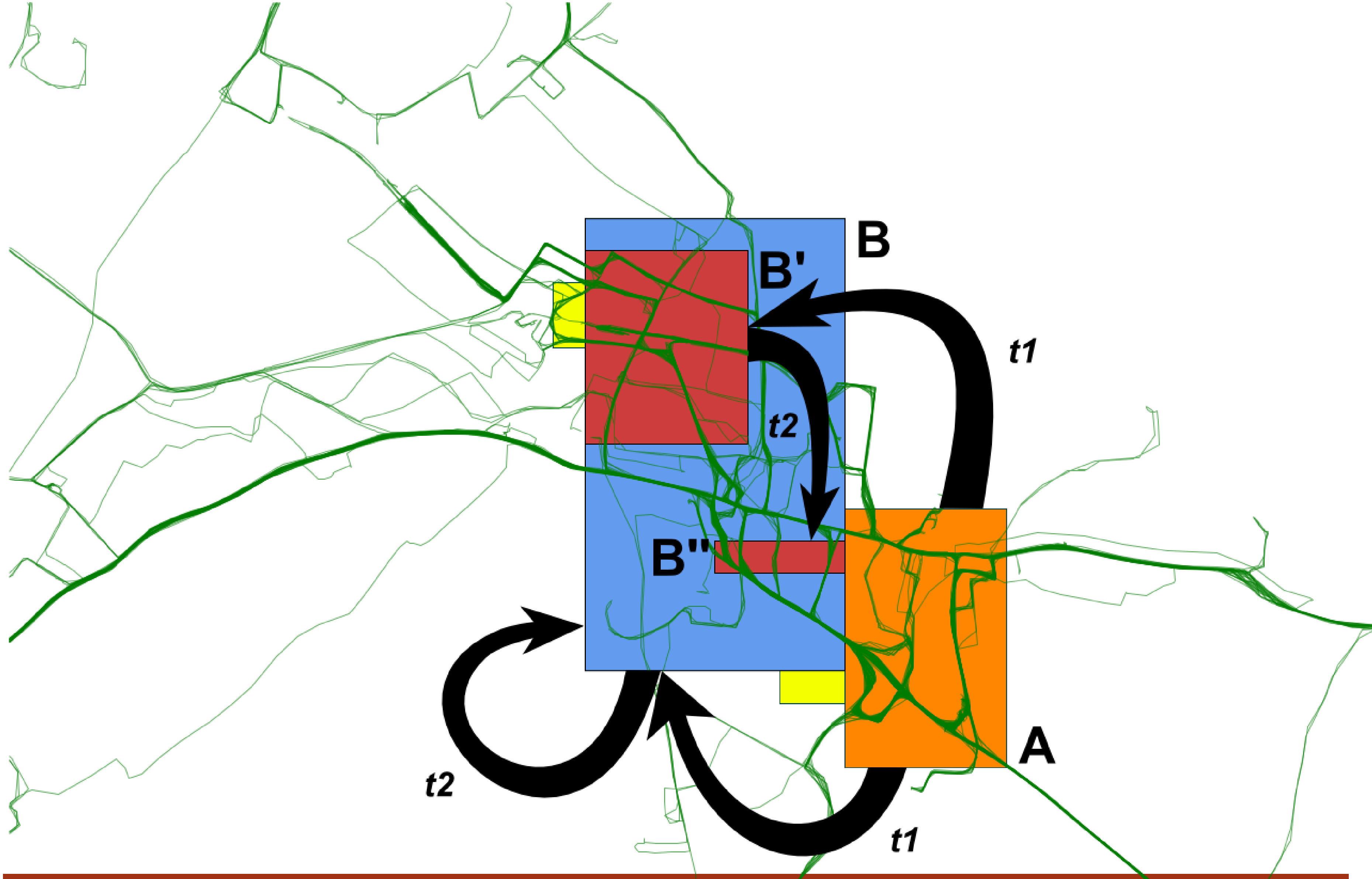


- Focusing on A->F (with some transition time), we further restrict the set of trajectories involved
- The process is repeated as far as possible

Experiment

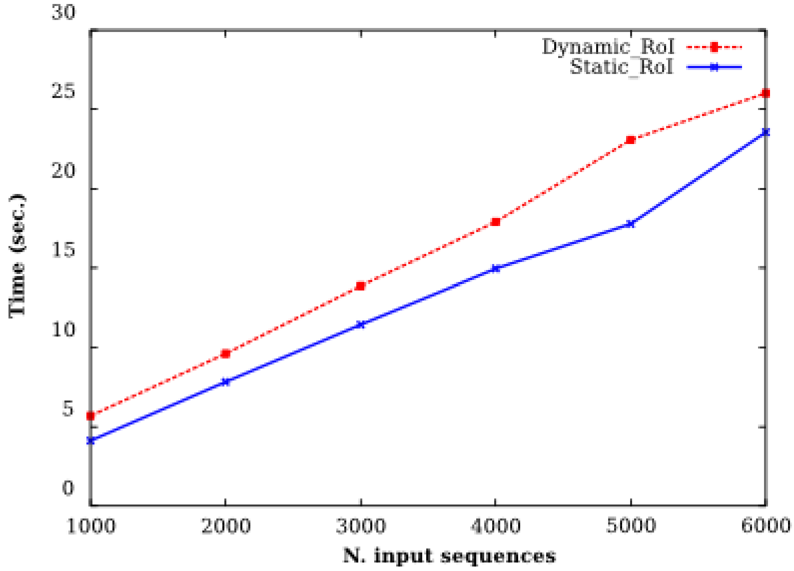
- GPS traces of a fleet of 273 trucks in Athens, Greece, for a total of 112203 point
- Data contain 50% of purely random trajectories
- 50% of trajectories that follow predefined patterns, randomly chosen among a set of 100 (random) patterns.
- grid of size 100x100

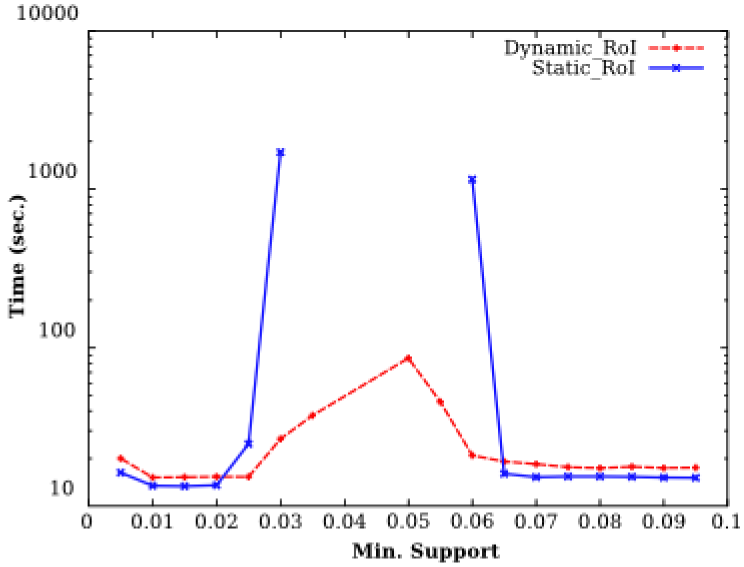
Result



Center-of-ASR attack

- Linear scalability w.r.t. number of trajs
- Quickly growing cost around (left& right) critical support thresholds
  - Dynamic approach prunes better





Conclusion

- Introduced the trajectory pattern mining
- several different methods to extract T-patterns

Future work

- Application-oriented tests on large, real datasets.
- Privacy issues
- Integration with background geographic knowledge
  - trajectory pre-processing, RoI discovery, T-patterns mining and post-processing

Q?

Thank you!