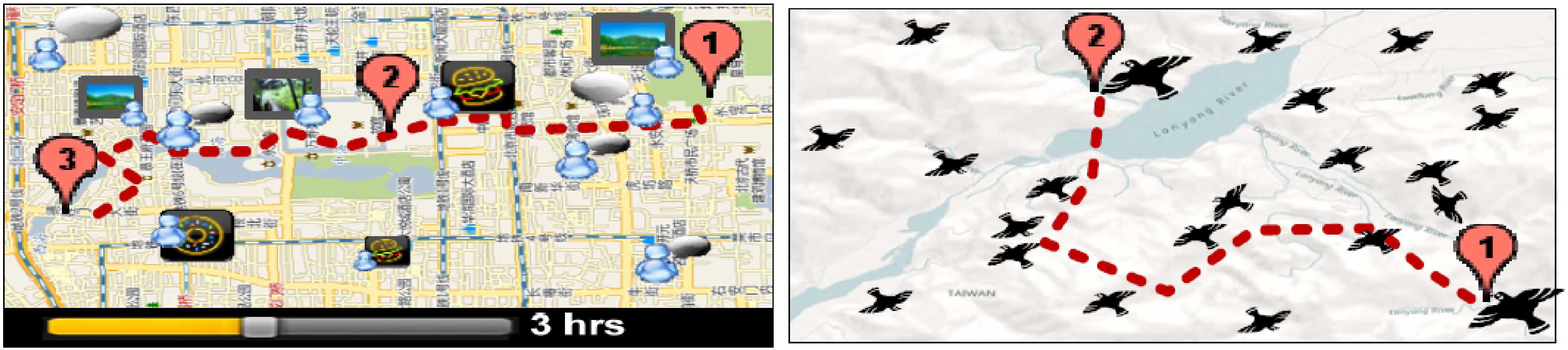
Constructing Popular Routes from Uncertain Trajectories

Ling-Yin Wei, Yu Zheng, Wen-Chih Peng

presented by Slawek Goryczka

Scenarios

- A **trajectory** is a sequence of data points recording location information and a time-stamp.
- In the prior work authors assumed that road network information is available, and this assumption is still impacting their research.
- Scenarios:

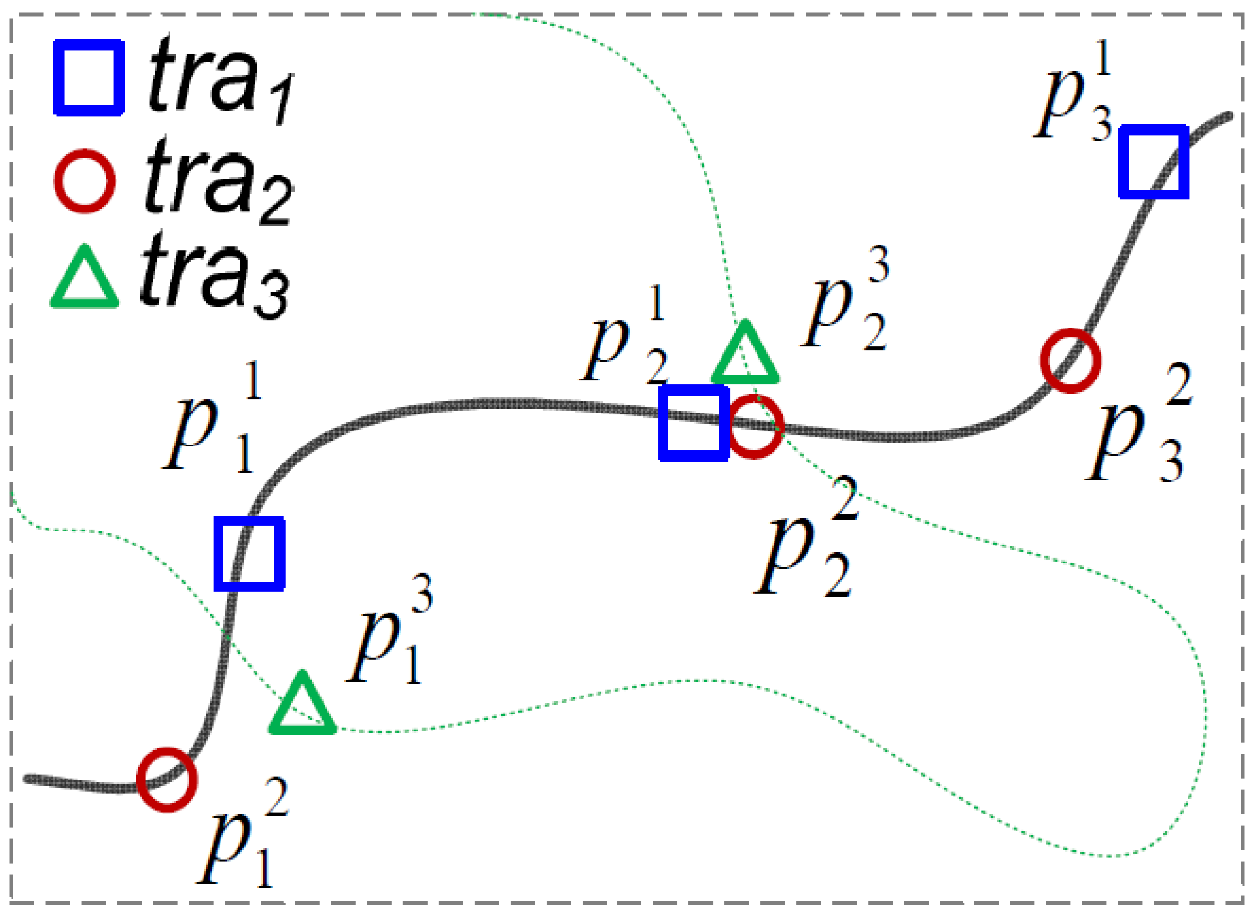


(a) (b)

Figure 3. Scenarios of applications.

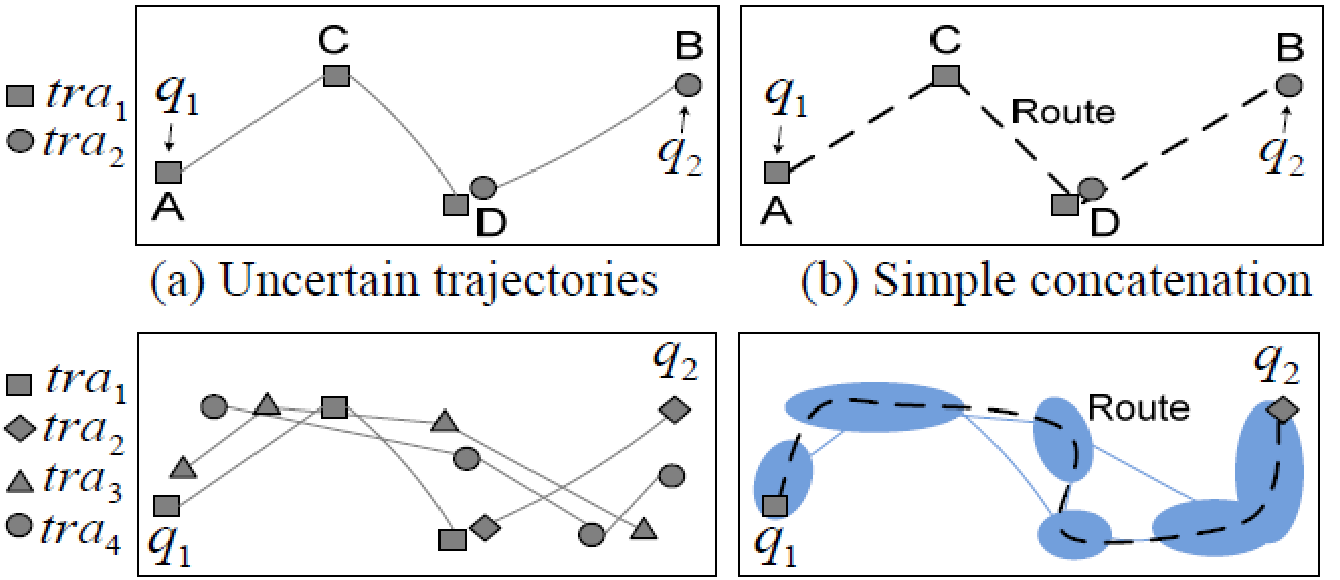
Time interval

If a time interval between two consecutive sampled points is large, the uncertainty of the route between the two points would increase.



Motivation and goals

Many trajectories (e.g., GPS) are usually generated at a low frequency due to energy saving and features of applications, resulting in the uncertainty of a moving object's mobility in a trajectory.



(a) Uncertain trajectories (b) Simple concatenation
(c) Uncertain trajectories (d) Mutual reinforcement

Figure 2. Examples of uncertain trajectories.

RICK

Route Inference framework based on Collective Knowledge (abbreviated as RICK) constructs the top-*k* **popular** routes from uncertain trajectories, which sequentially pass through the locations within the specified time span by aggregating such uncertain trajectories in a mutual reinforcement way
(i.e., uncertain + uncertain → certain)

- construct a routable graph by collaborative learning among the uncertain trajectories
- construct the top-*k* routes according to a user-specified query

RICK overview

A) **Routable graph construction**:

A.1 (Geographical) region (cluster) construction

A.2 Edges inference: among regions & within a region

B) **Route inference**:

B.1 Route generation (top-*k* rough routes)

B.2 The route refinement to derive a detailed route

RICK overview

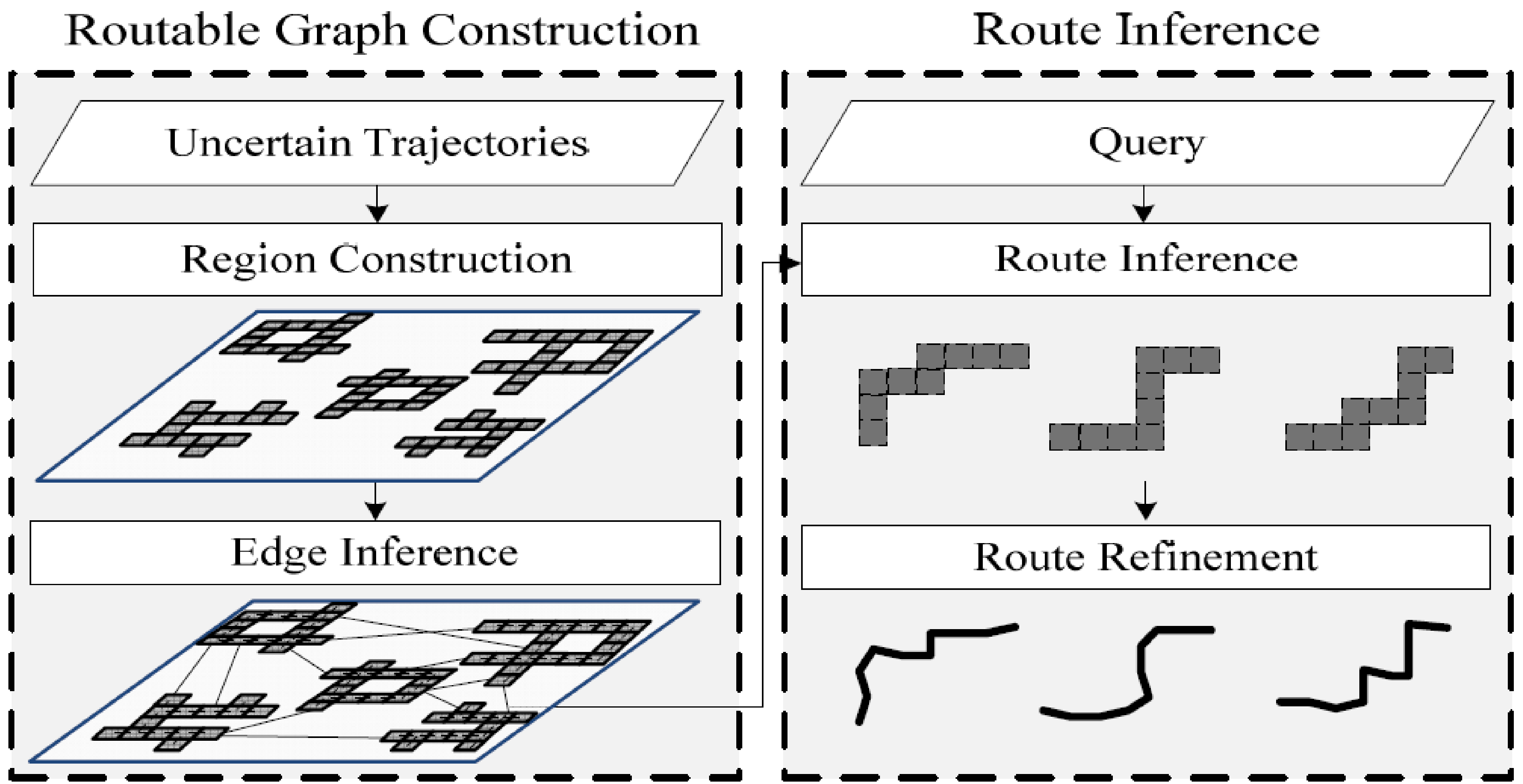


Figure 4. Overview of RICK.

Definitions

Given two trajectories A, B, and their subtrajectories *a*, *b*, resp.; *a* is **spatio-temporally correlated** with *b* if:

- traversing them takes similar time: $(t_a - t_b)/\max(t_a, t_b) \leq \theta$
- one end points of *a* and *b* are in the same cell, why other is spatially close (they coordinates differ by ≤ 1)

The **connection support of the cell pair (g, g')** is defined as $|T_1 \cup T_2|$, where:

- T_1 = {(A, B)|*a* and *b* are st-correlated, g→g'' in *a*, g'→g'' in *b* for some g'' in G - {g, g'}}
- T_2 = {(A, B)|*a* and *b* are st-correlated, g←g'' in a, g'←g'' in *b* for some g'' in G - {g, g'}}

Definitions (2)

If the connection support of the cell pair (g, g') is greater than or equal to a threshold *C*, g and g' are **neighbors** (gNg').

Given a set of cells G', G' forms a **region** if for any two cells g, g' in G', there exists a chain of cells (g=) $g_1 = g_2 = \dots = g_k$ (=g') s.t. $g_i N g_{i+1}$ for each g_i in G' and i in [1, k).

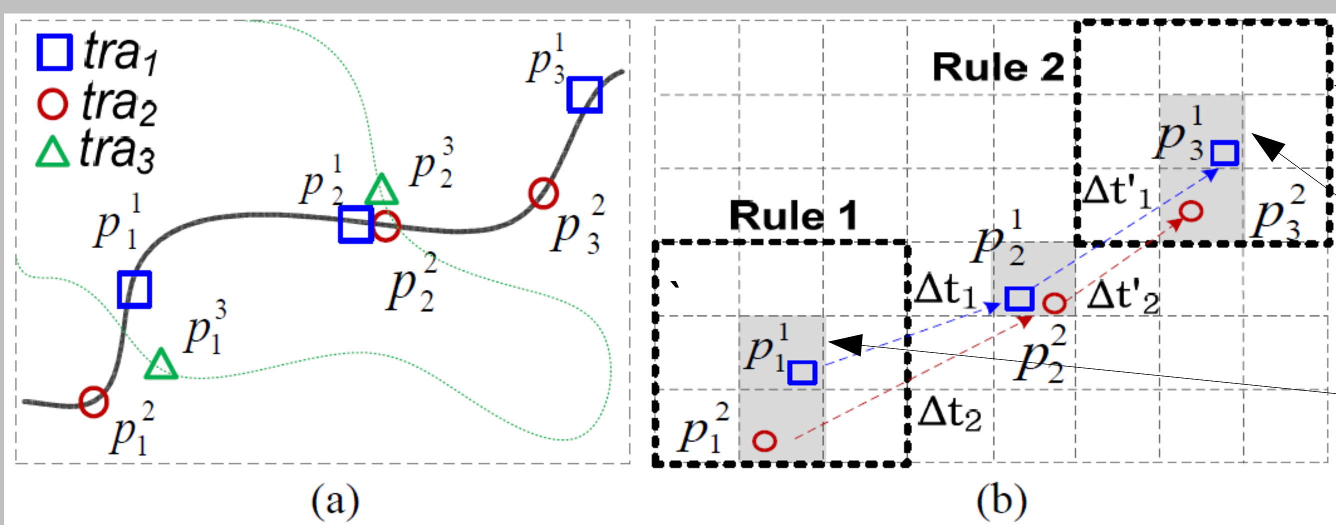


Figure 5. Spatio-temporally correlated uncertain trajectories.

Spatially close cells

Spatio-temporal correlation equal to 1

A.1 Region construction (1)

- Naïve method: consider all pairs of cells: $O(n^2)$ cells
- Using grid index structure:

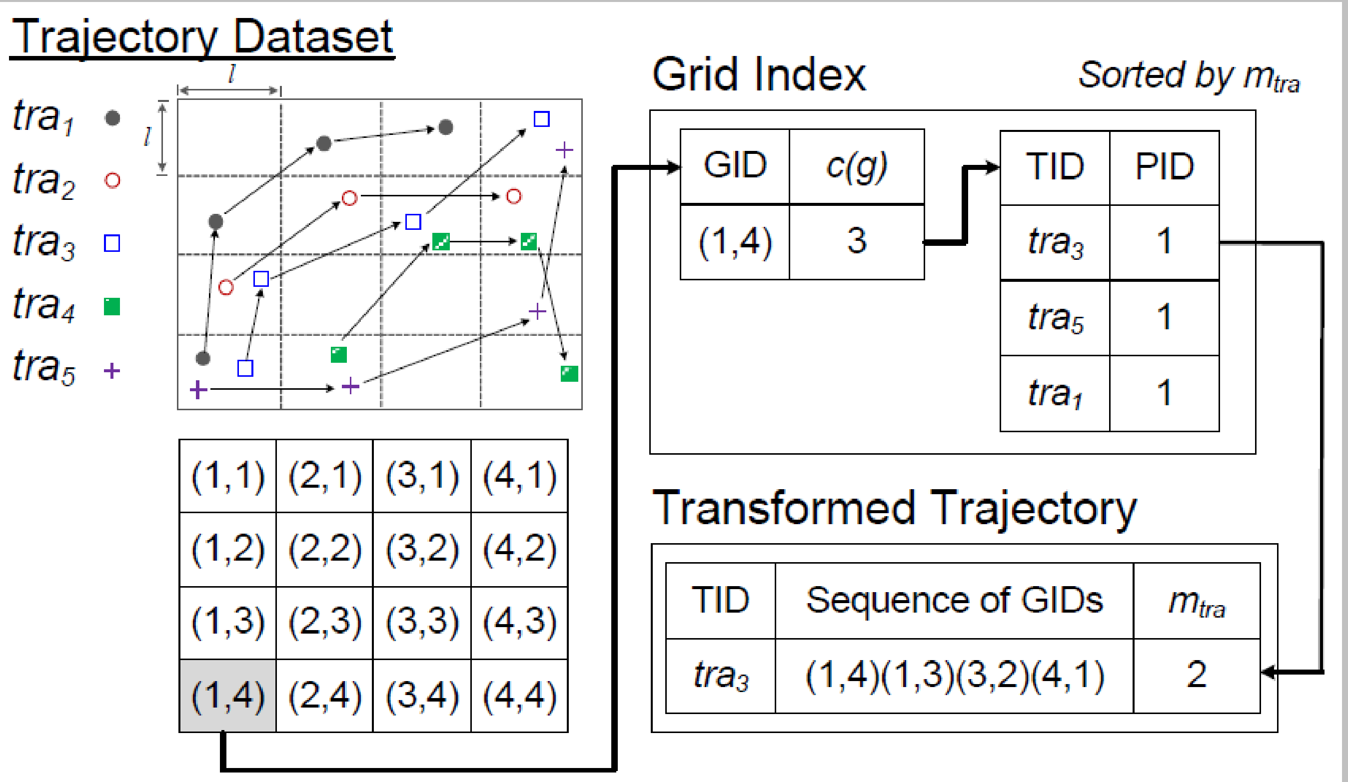


Figure 6. An example of an index structure.

A.1 Region construction (2)

Algorithm 1: Region Construction

Input: An uncertain trajectory dataset D, a set of cells $\mathcal{G}$, a temporal constraint θ, and a minimum connection support C.
Output: A set of regions R.

$\mathcal{G}' \leftarrow$ Sort cells in $\mathcal{G}$ in a decreasing order of $c(g)$;
Do
    $g \leftarrow$ Pop the cell from $\mathcal{G}'$;
    Foreach *tra* traversing g by the order stored in the grid index
        $\tau(g) \leftarrow \{p \mid p.g = g \text{ and } p \in tra\}$;
        Foreach $p \in tra - \tau(g)$ and $p.g$ is not enclosed
            If $p.g$ is contained in some region
                $r \leftarrow$ The region contains $p.g$;
            Else
                $r \leftarrow \emptyset$;
            If p is before p' for all $p' \in \tau(g)$
                $r \leftarrow \mathrm{CM}(r, p.g, g, \theta, C, Rule1)$;
            ElseIf p is after p' for all $p' \in \tau(g)$
                $r \leftarrow \mathrm{CM}(r, p.g, g, \theta, C, Rule2)$;
            Else
                $r \leftarrow \mathrm{CM}(r, p.g, g, \theta, C, Rule1)$;
                $r \leftarrow \mathrm{CM}(r, p.g, g, \theta, C, Rule2)$;
            $R \leftarrow R \cup \{r\}$;
Until $\mathcal{G}'$ is empty or each cell in $\mathcal{G}$ is in some $r \in R$;
Return R;

Naïve solution: $O(n^3m)$

This solution:
$O(n(\text{long } n + cm^2))$

In all regions of g there is at least one other cell g', which is not sp-close to g.

Cell merging to build a region (cluster).

RICK overview

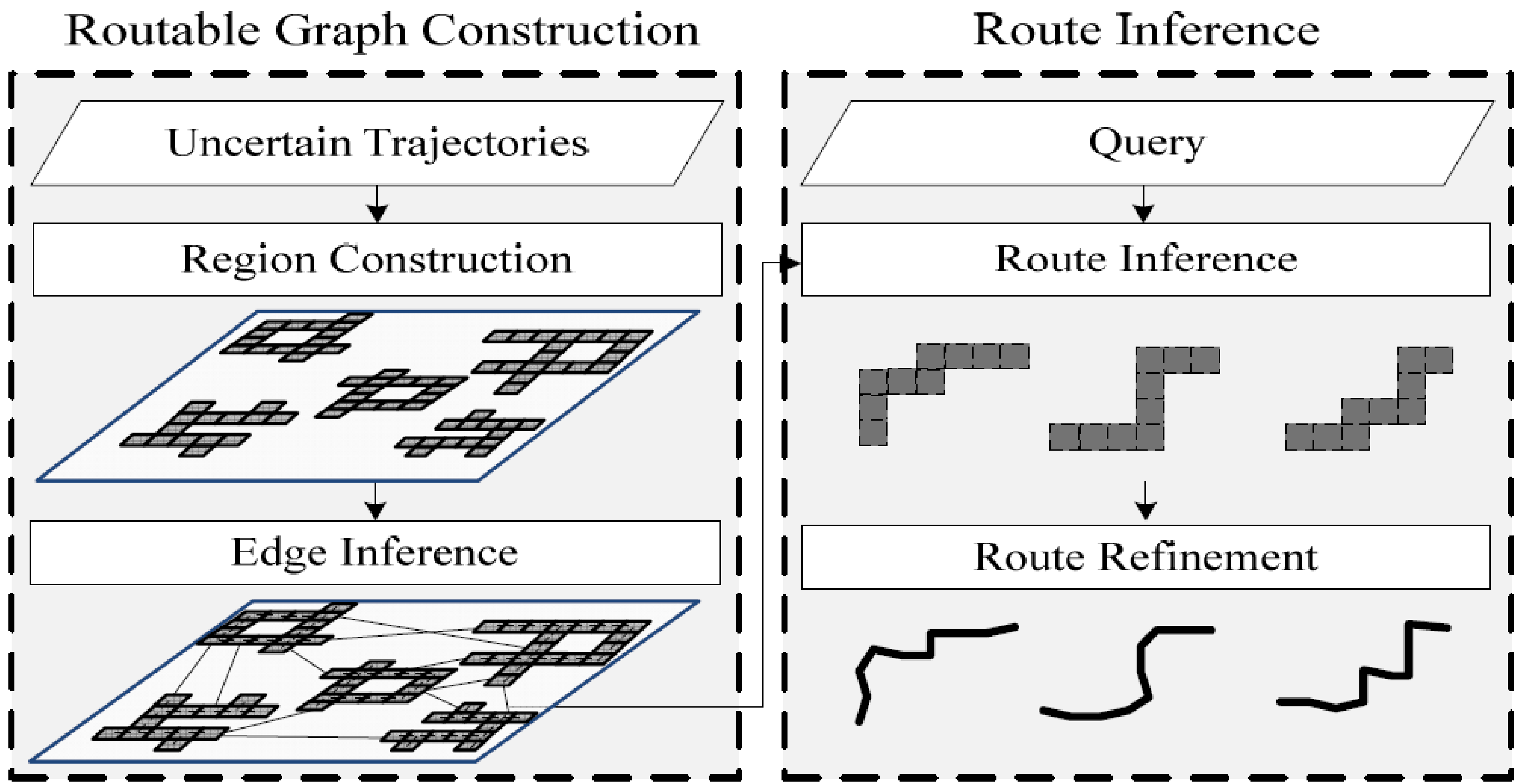


Figure 4. Overview of RICK.

A.2 Edge inference in a region

- Each edge between cells *e* has the transition support *e.s* (# trajectories traversing the edge) and the travel time *e.t*.
- For each trajectory traversing the region, we infer the shortest path between any two consecutive points of the trajectory, by treating the region as a directed graph.
- Drop ***redundant*** edges, i.e., edges that can be substituted by a sequence of edges with smaller average transition support

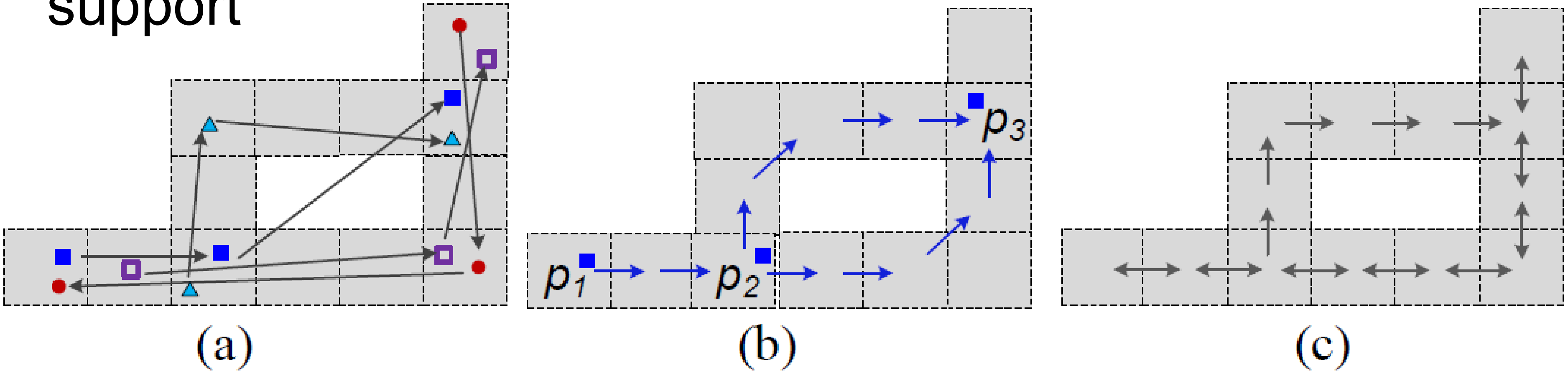


Figure 8. Edge inference in a region.

A.2 Edge inference among regions

- Generate edges similar as edges inside a region.
- Remove redundant edges similar as redundant edges inside a region

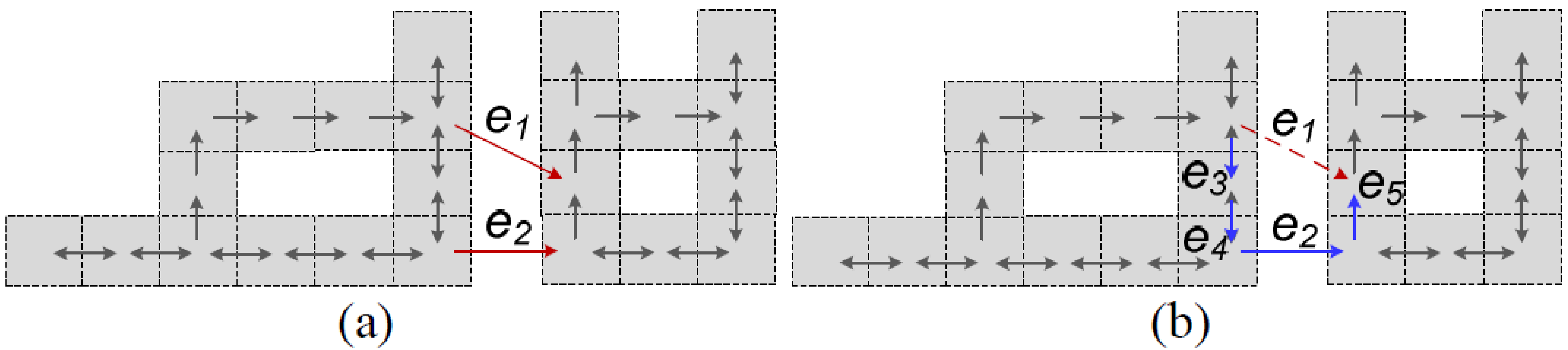


Figure 9. Edge inference between regions.

RICK overview

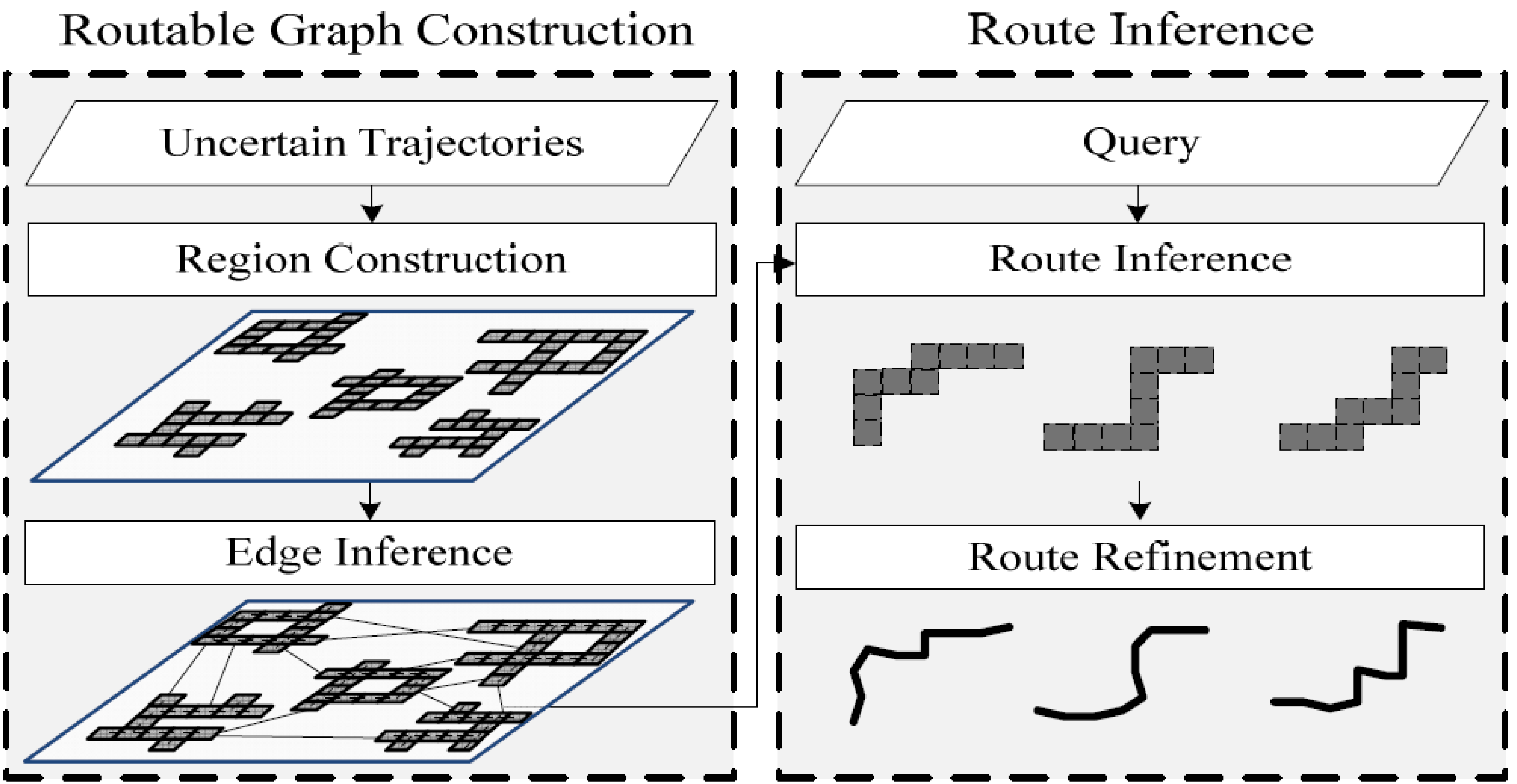
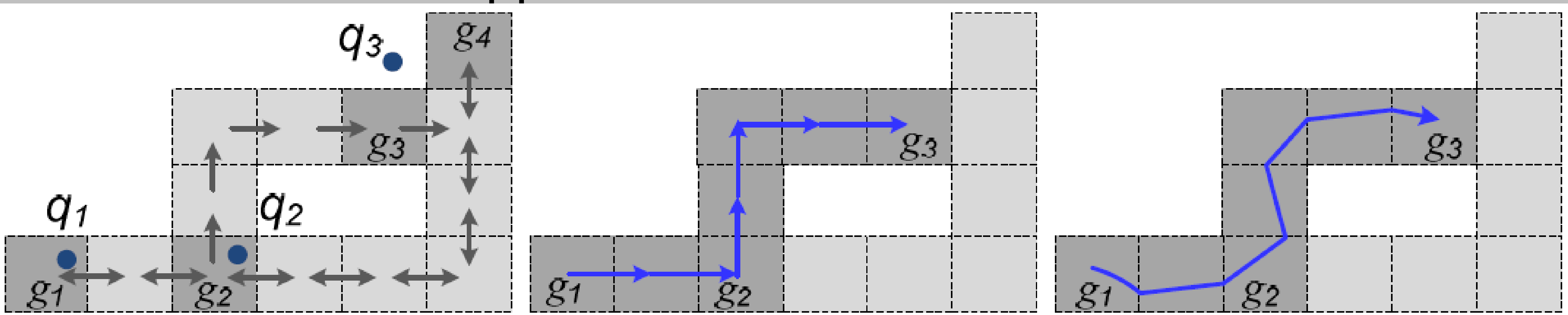


Figure 4. Overview of RICK.

B.1 Route generation

- Find cells in a region, which is closest to given locations, e.g. $g_1 \rightarrow q_1$, $g_2 \rightarrow q_2$, $\{g_3, g_4\} \rightarrow q_3$
- Find top-*k* cell sequences (with max. route score), using min. distance between any consecutive cells assigned to traversing points.
- Find top-*k* global (long) sequences using branch-and-bound search approach.

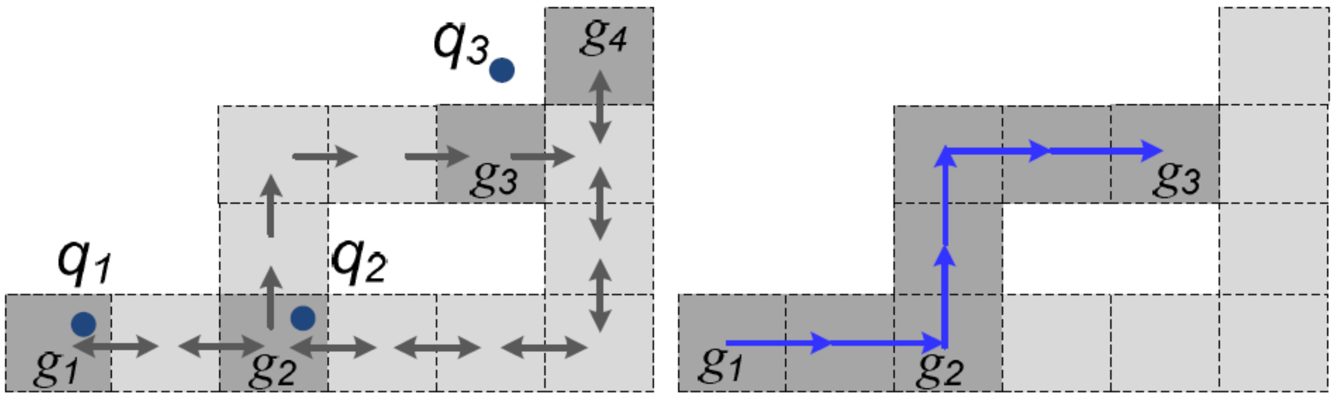


(a)Query transformation. (b)Route generation. (c)Route refinement

Figure 10. Route inference.

B.1 Top-*k* route generation

- **Route score** – sum of scores for all consecutive routing cells, i.e., sum of average numbers of traverses crossing connections between consecutive routing cells
- Connect centers of selected cells



(a)Query transformation. (b)Route generation.

B.1 Route generation (regions)

- Limit considered regions using lower bound of transition time (range *r*) computed from data, and generate region sequences first
- Use A*-like algorithm to find routes among regions
- Then, find routes within regions

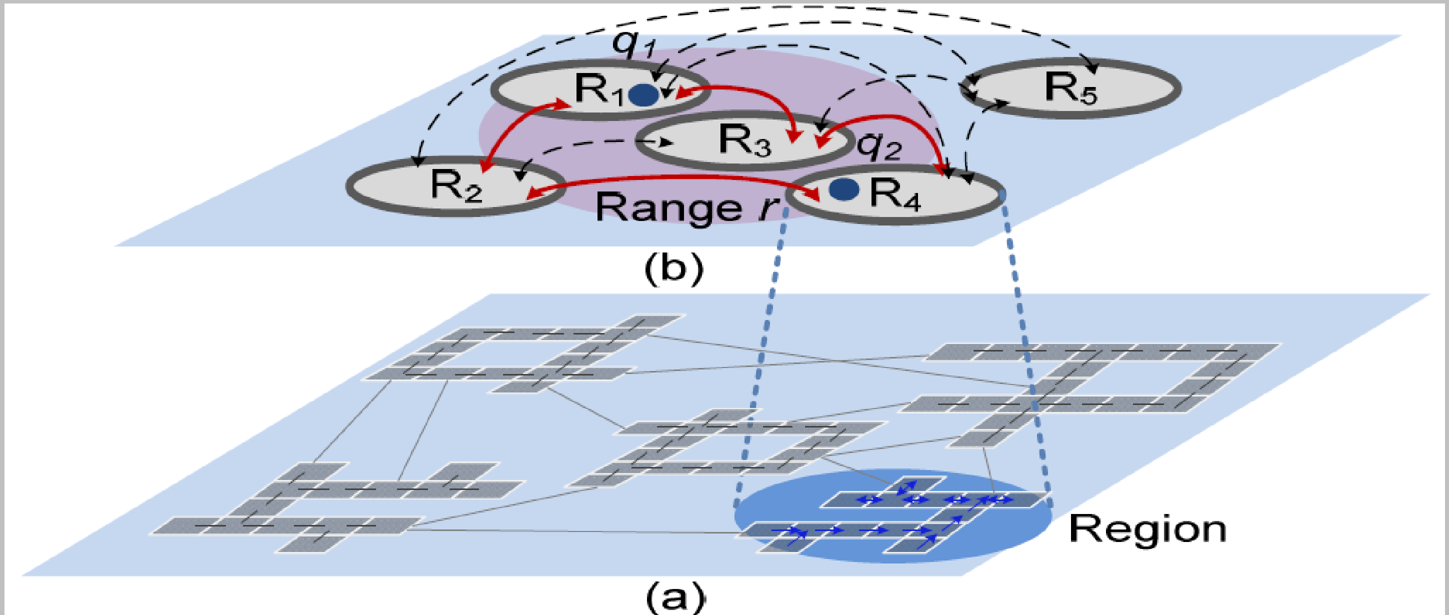


Figure 11. The scenario of the two-layer routing algorithm.

RICK overview

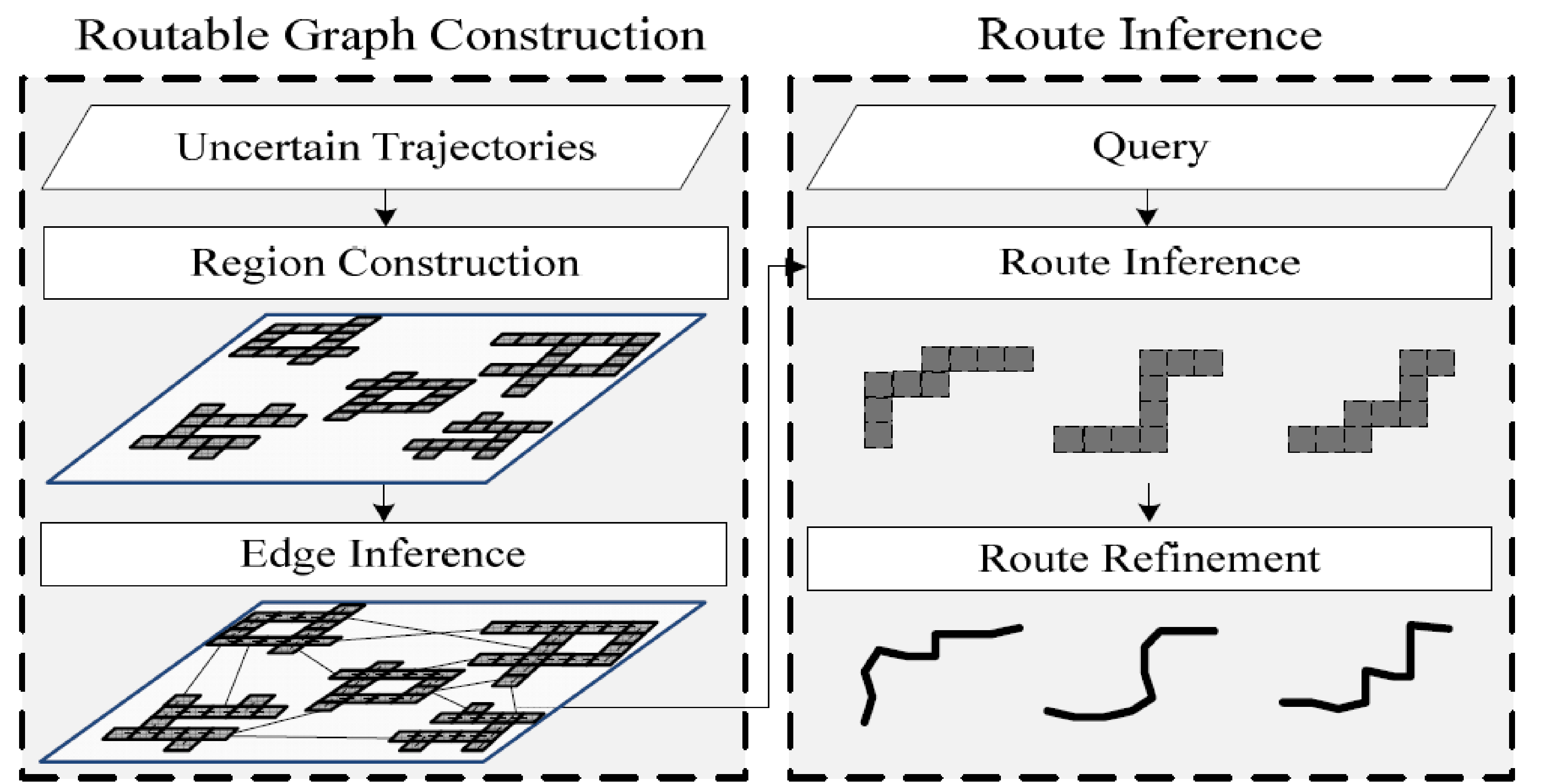
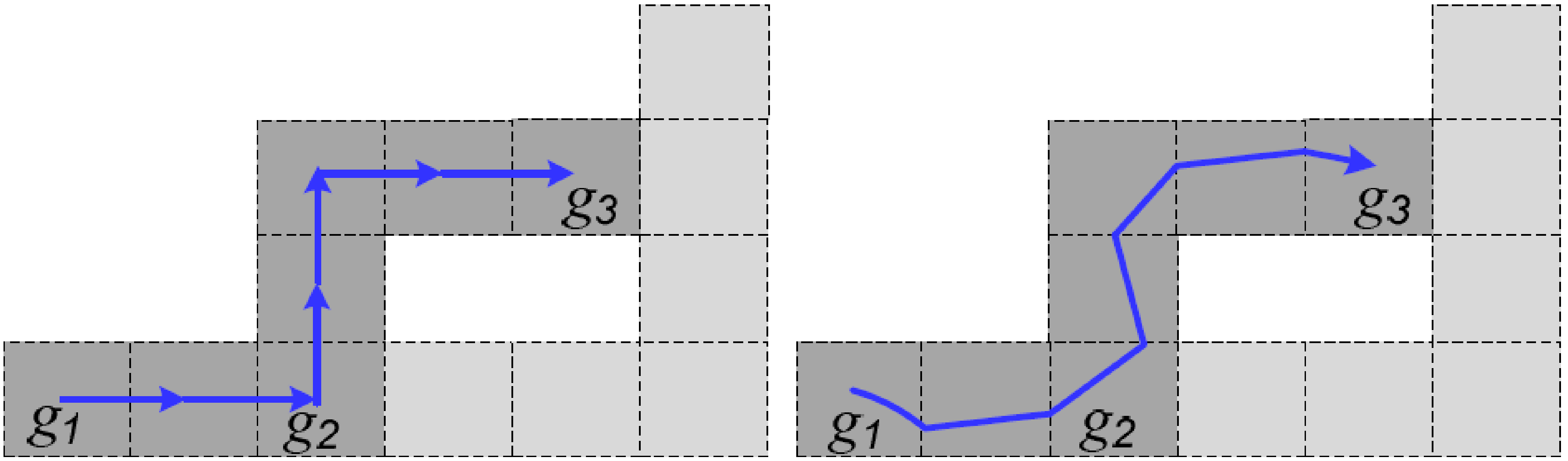


Figure 4. Overview of RICK.

B.2 Route refinement (1)

- Select the historical uncertain trajectories that traverse the cells in the same order as the route.
- Extract the data points that locate in cells of the rough route from these selected uncertain trajectories, and thus derive a set of points for each cell of the route.
- To formulate a specific route from selected points, adopt linear regression for the set of points of each cell to derive a segment.
- Concatenate the segments in the same order as an original inferred route.

B.2 Route refinement (2)



(b)Route generation. (c)Route refinement

Experiments (settings)

- Check-in dataset from Foursquare in NYC (6,600 trajectories)
- 15,000 taxi trajectories in Beijing
- length-normalized dynamic time warping distance of raw trajectory tra, and computer route p:
  NDTW(p, tra) = DTW(p, tra) / p.length
- maximum distance (MD): max. distance measured by the optimal alignment path.

Therefore, the two measurements for evaluating the inferred top-k routes are defined as follows:

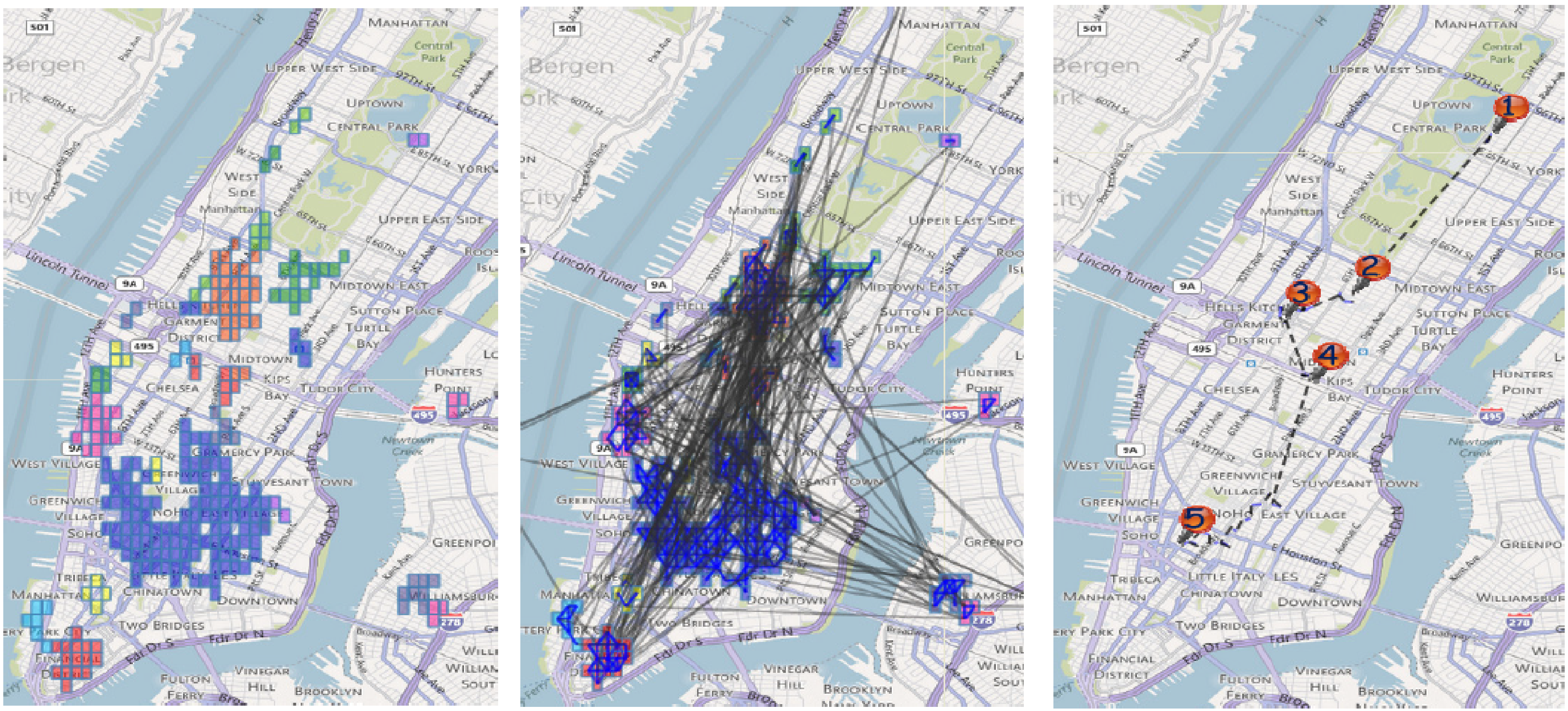
$$\mathrm{NDTW}(\mathrm{T}, \mathrm{T}') = \mathrm{Avg}_{p_k \in \mathrm{T}} \min_{tra \in \mathrm{T}'} \mathrm{NDTW}(p_k, tra), \text{ and}$$

$$\mathrm{MD}(\mathrm{T}, \mathrm{T}') = \mathrm{Avg}_{p_k \in \mathrm{T}} \mathrm{MD}(p_k, tra'),$$

where T is the set of inferred top-k routes, T′ is the set of top-k raw trajectories, and $tra' = \mathrm{Arg}\min_{tra \in \mathrm{T}'} \mathrm{NDTW}(p_k, tra)$.

Experiments (visualization)

Find a route: Central Park → The Museum of Modern Art → Times Square → Empire State Building → SoHo



(a) Regions (b) Routable graph (c) Top-1 route

Figure 12. Visualization of results in Manhattan.

Experiments (2)

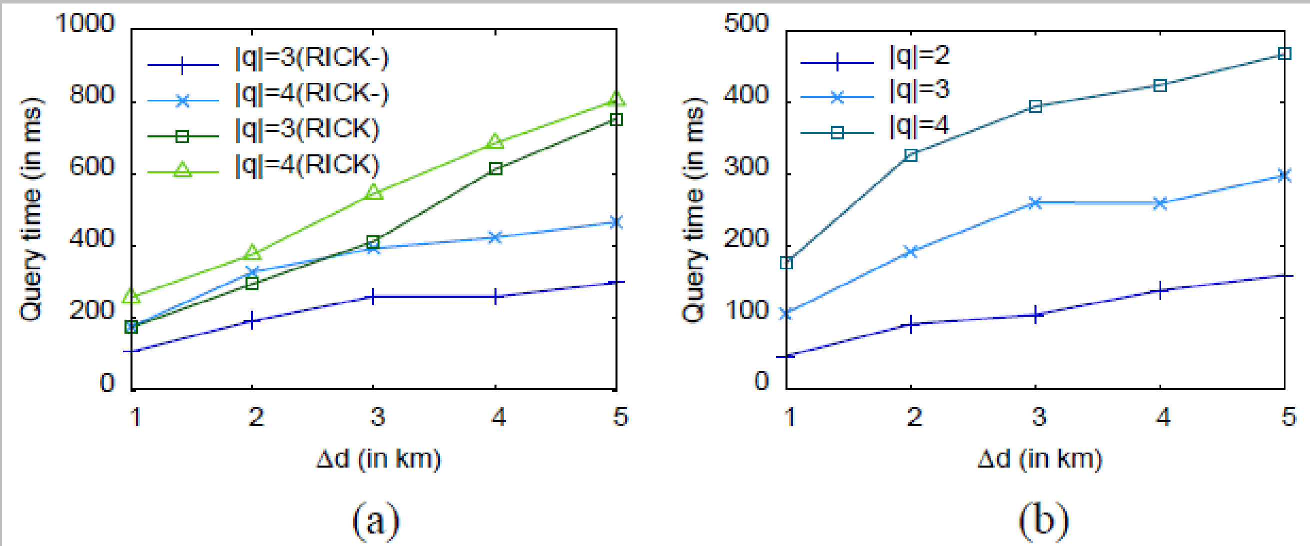


Figure 16. Efficiency evaluation.

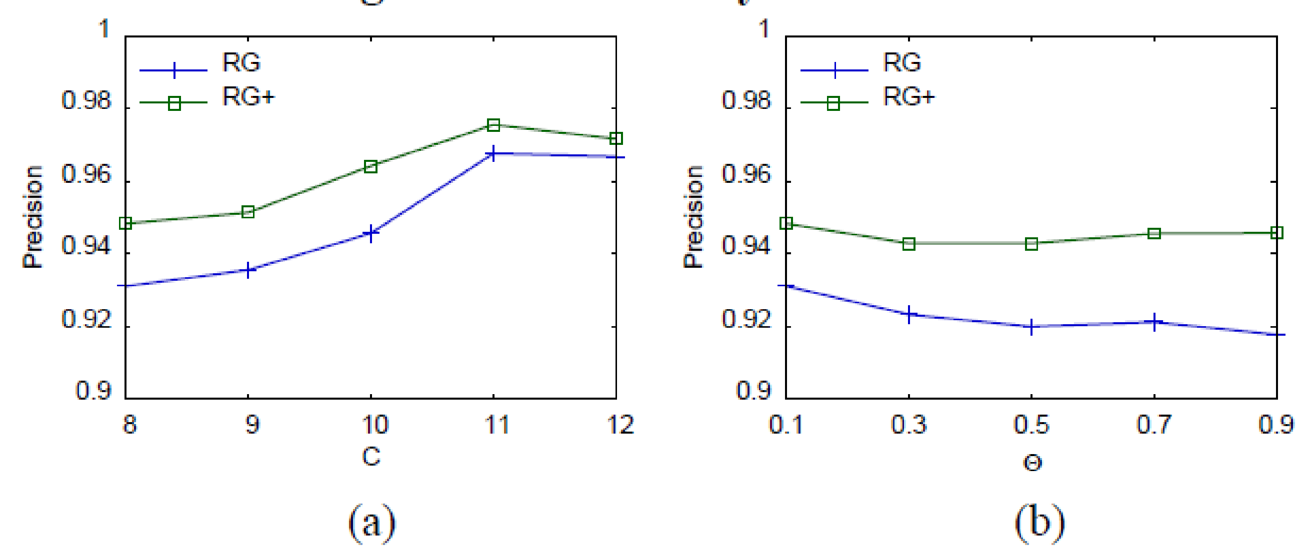


Figure 17. Connectivity evaluation.

Δd – distance between any two consecutive query locations

|q| - the length of query location sequence

RICK- → RICK without two layers routing algorithm

RG+ → with refinement

C – minimum connection support

Θ – temporal constraint

Precision → |{e | e is traversed by some *tra* in D and e in E}| / |E|

Experiments (3)

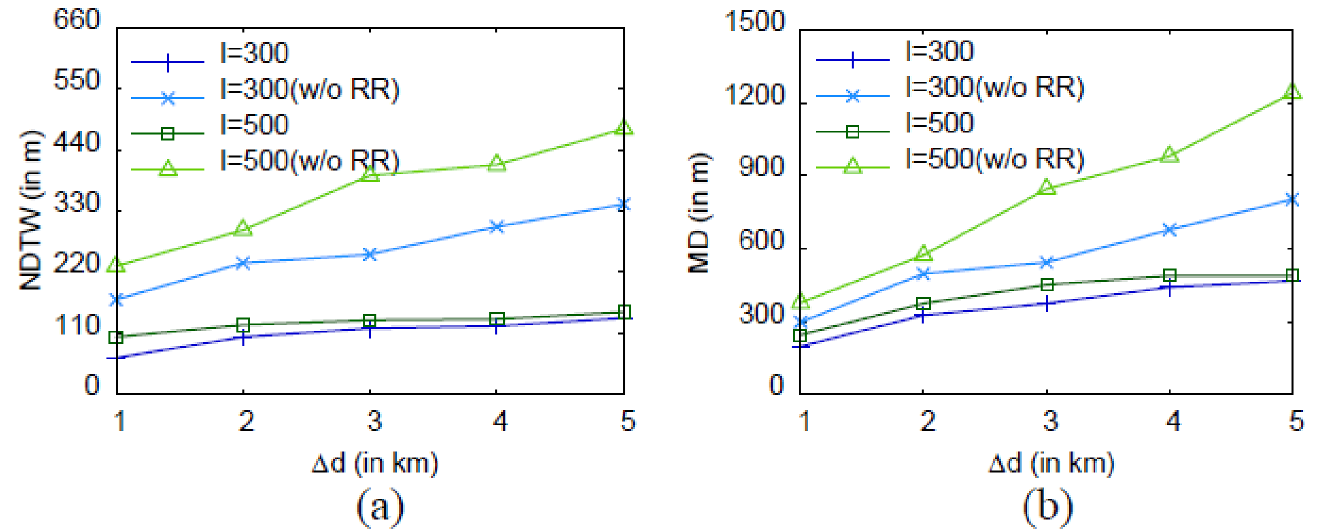


Figure 14. Effect on route refinement.

Δd – distance between any two consecutive query locations

l – cell length (meters)

Q & A