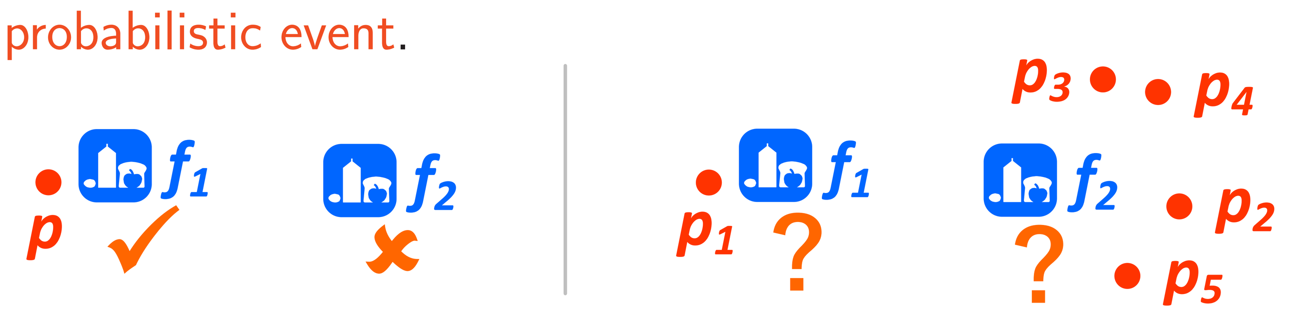
PINOCCHIO: Probabilistic Influence-based Location Selection over Moving Objects

Meng Wang[1], Hui Li[1], Jiangtao Cui[1], Ke Deng[2], Sourav S Bhowmick[3], Zhenhua Dong[4]

[1]Xidian University, China
[2]RMIT University, Australia
[3]Nanyang Technological University, Singapore
[4]Huawei Noah's Ark Research Lab, China

Introduction

- The location selection (LS) problem aims to mine the optimal location from a set of candidates to place a new facility such that a score (i.e., benefit or influence on some given objects) can be maximized.
- State-of-the-art LS studies assume each object is stationary and can be definitely influenced by only a single facility (e.g., by the nearest facility).
- In real-world scenarios, objects (e.g., people, vehicles) are mobile and are influenced by multiple facilities simultaneously, where the influence is in fact a probabilistic event.



(a) for a static object (b) for a moving object

- In this paper, we propose a novel and generalized LS problem called PRIME-LS, which takes mobility and probability factors into account.
- We present an algorithm called PINOCCHIO, which leverages two pruning rules based on a novel distance measure, to solve PRIME-LS, and further extend it by incorporating two optimization strategies.

PRIME-LS Problem

- We use a set of discrete positions $O = \{p_1, p_2, \ldots, p_n\}$ to denote a moving object (continuous data can be discretized by sampling).
- As the probability of an object to access a facility is negatively correlated to the distance, we assume the influence probability only depends on distance, and use a monotonically decreasing probability function *PF* to depict the behavior pattern of influence.
- At any position of a moving object, the probability that she is influenced by a facility is independent. Inspired by Influence Model, we define the probability as cumulative influence probability:

$$Pr_c(O) = 1 - \prod_{i=1}^{n}(1 - Pr_c(p_i)), \text{(candidate facility } c \text{ influences } O)$$

- Given a user-specified threshold τ, a company can set a baseline of minimum expectation of $Pr_c(O)$, where facility c influences moving object O *iff* $Pr_c(O) \geq \tau$. This means companies can tune τ to make a trade-off between quantity and quality of influenced customers.
- The influence value is the number of moving objects that are influenced by a candidate facility c, denoted as $inf(c)$.
- The **PR**obabilistic **I**nfluence-based **M**obility-awar**E** **L**ocation **S**election (PRIME-LS) problem aims to mine the optimal candidate $c \in C$ such that $\forall c' \in C - \{c\}$, $inf(c) \geq inf(c')$.

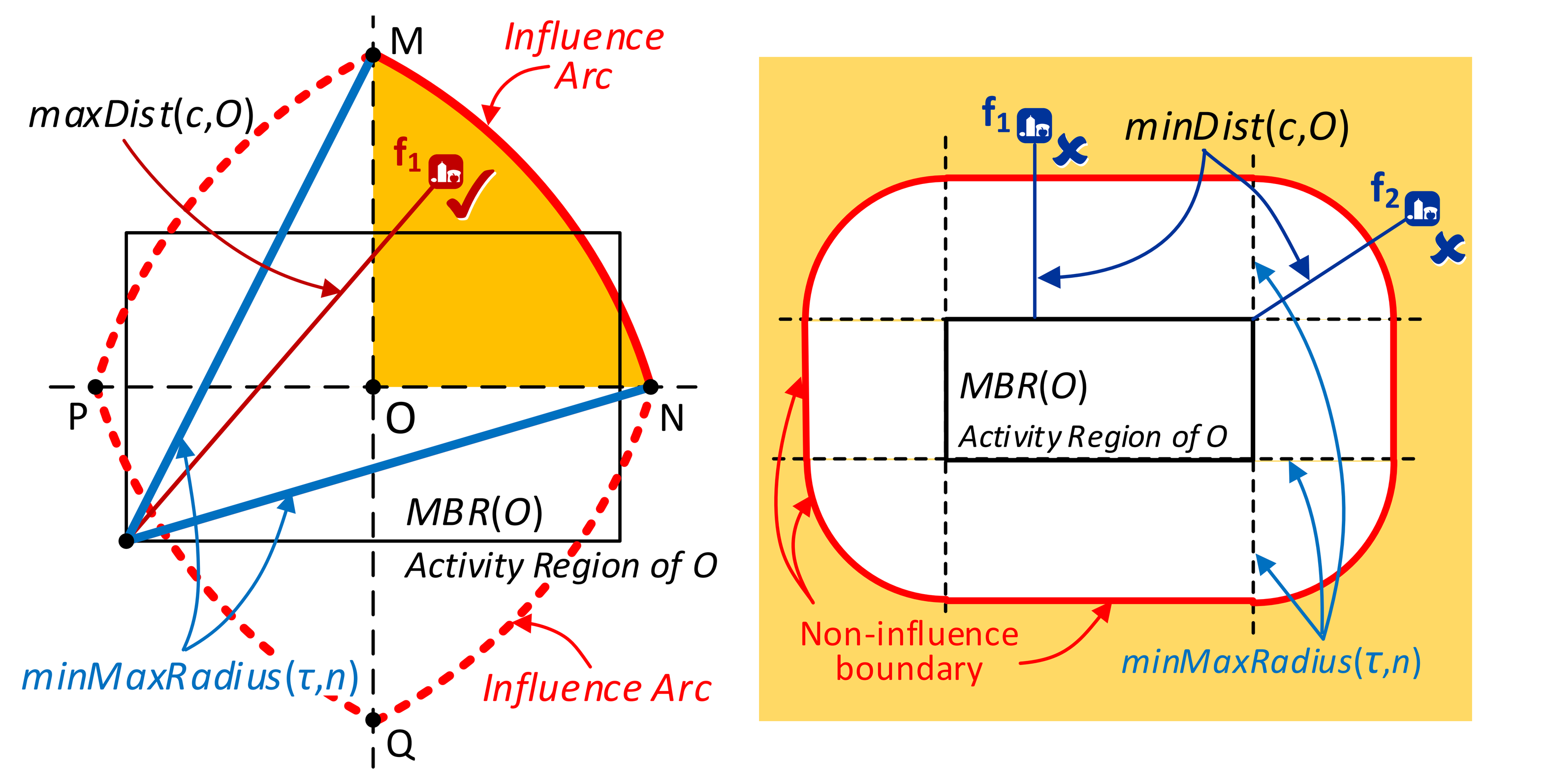
Solution to PRIME-LS

- As activity regions of moving objects are highly overlapping and may enclose some candidates, where existing pruning techniques cannot be adopted, we propose a novel pruning measure, minMaxRadius, to quantify the cumulative influence probability.

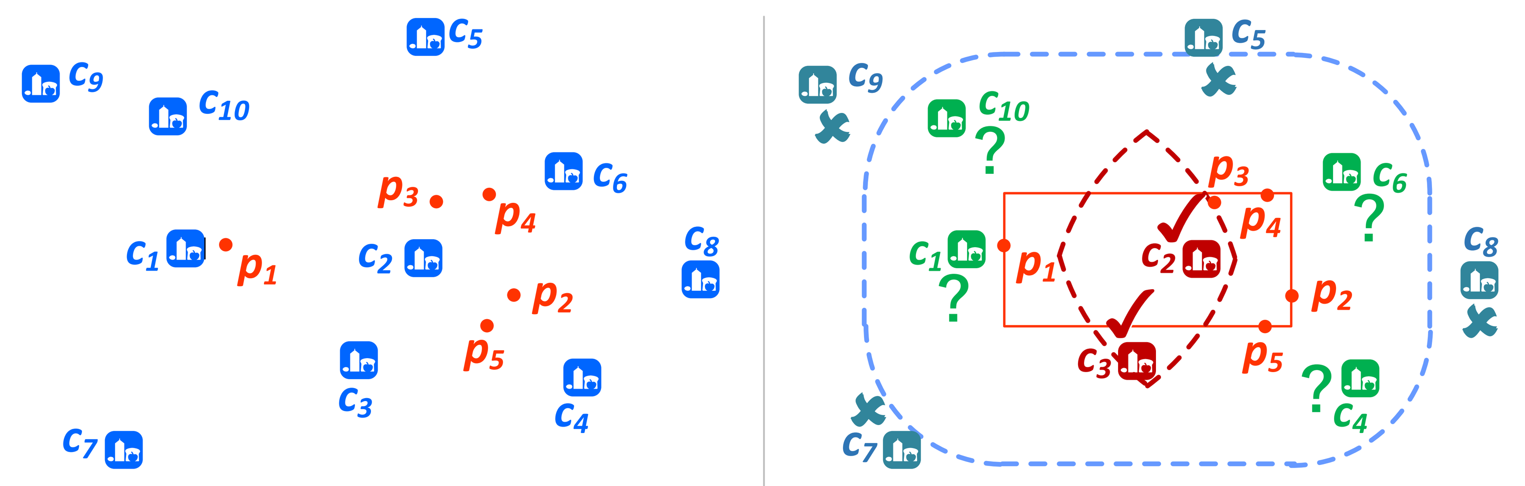
$$minMaxRadius(\tau, n) = PF^{-1}(1 - (1 - \tau)^{1/n}), (n \text{ positions})$$

- Based on minMaxRadius, we design two pruning rules, influence arcs (IA) rule and non-influence boundary (NIB) rule.
- **Key Idea** of PINOCCHIO (**P**robabilistic **IN**fluence-based **LOC**ation Sele**C**tion Tec**H**nique over Mov**I**ng **O**bjects):
  - Calculate minMaxRadius and the IA/NIB areas for each object.
  - Based on IA rule, we identify the candidates that influence the object.
  - For the remnant candidates, NIB rule is used to exclude the candidates that cannot influence the object.
  - The remnant candidates are verified using the definition of influence.

Solution to PRIME-LS (Cont.)



(a) Influence arcs (IA) rule (b) Non-influence boundary (NIB) rule



(c) Object & Candidates (d) Pruning with IA & NIB

- We enhance the PINOCCHIO algorithm, referred to as PINOCCHIO-VO, by incorporating two optimization strategies.
  - **Upper-lower bounds of influence**. If $maxInf(c) < maxminInf$, c cannot influence the largest number of objects and can be pruned, where $maxInf(c)$ is the maximum of the possible influence of c, $minInf(c)$ is the identified influence of c and $maxminInf = \max_{c \in C} minInf(c)$.
  - **Early stopping**. The validation is accomplished by computing only partial n' positions of O instead of all the n positions, if $Pr_c^{n-n'}(O) \leq 1 - \tau$, where $Pr_c^{n-n'}(O) = \prod_{i=n'+1}^{n}(1 - Pr_c(p_i))$ $(n' < n)$.

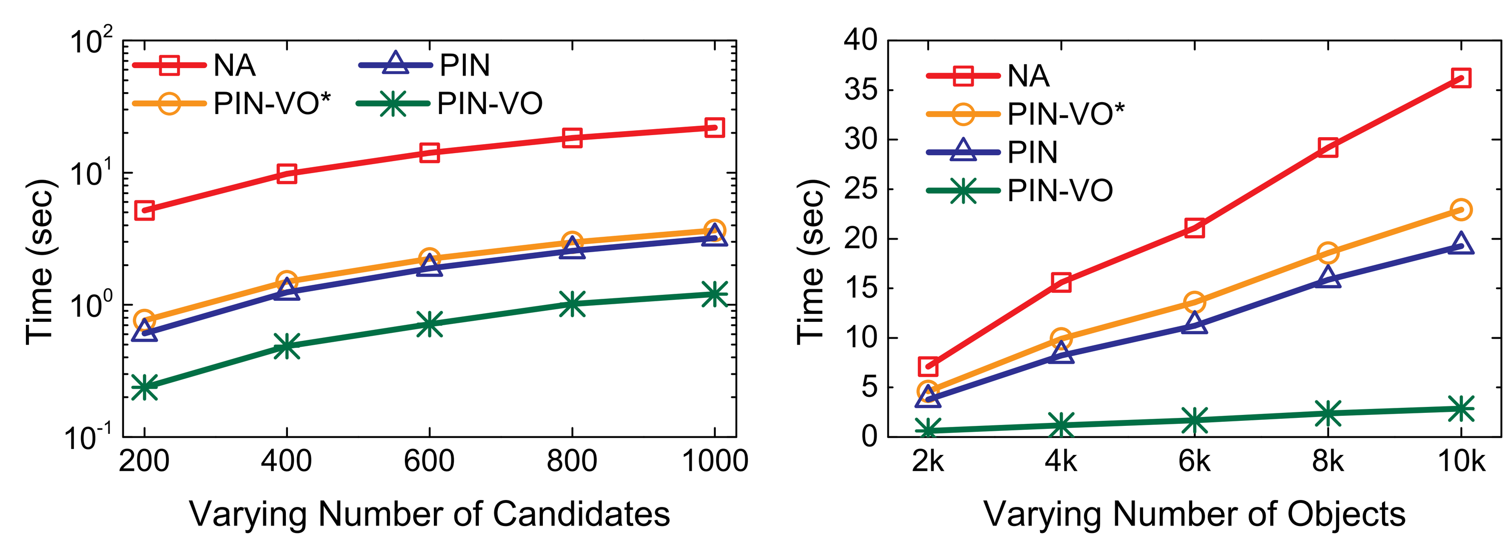
Experiment

- Two real-world check-in datasets: *Foursquare* (F) and *Gowalla* (G).
- *PF*: $Pr_c(p) = \rho(d_0 + dist(c, p))^{-\lambda}$ (The probability of a user checking-in at a POI decays as the power-law of the distance between them.)
- **Effectiveness on Different Semantics**: *PRIME-LS*, *BRNN** (adapted from NN-based MaxBRNN) and *RANGE* (range-based LS).

Avg. Precision	@10	@20	@30	@40	@50	Avg. Better
PRIME-LS	0.022	0.032	0.055	0.081	0.110	N/A
Avg. RANGE	0.020	0.031	0.050	0.071	0.092	12%
*BRNN**	0.015	0.028	0.040	0.056	0.085	35%

Average Precision Comparison

- **Performance Comparison**: Compared to NA (baseline: scan all object-candidate pairs), PINOCCHIO-VO has the best scalability, followed by PINOCCHIO and PIN-VO* (PINOCCHIO-VO without pruning). PINOCCHIO-VO prunes nearly 2/3 candidates.



- **Observation**: If we expect a certain number of objects to be influenced, the resulting locations are identical or very close with high accuracy, regardless of how n and τ are set.