MC²LS: Towards Efficient Collective Location Selection in Competition

Xi'an Polytechnic University

Meng Wang[1], Mengfei Zhao[1], Hui Li[2,3,*], Jiangtao Cui[2], Bo Yang[1], Tao Xue[1]

1. School of Computer Science, Xi'an Polytechnic University, China
2. School of Computer Science and Technology, Xidian University, China
3. Shanghai Yunxi Technology, China / *Corresponding author

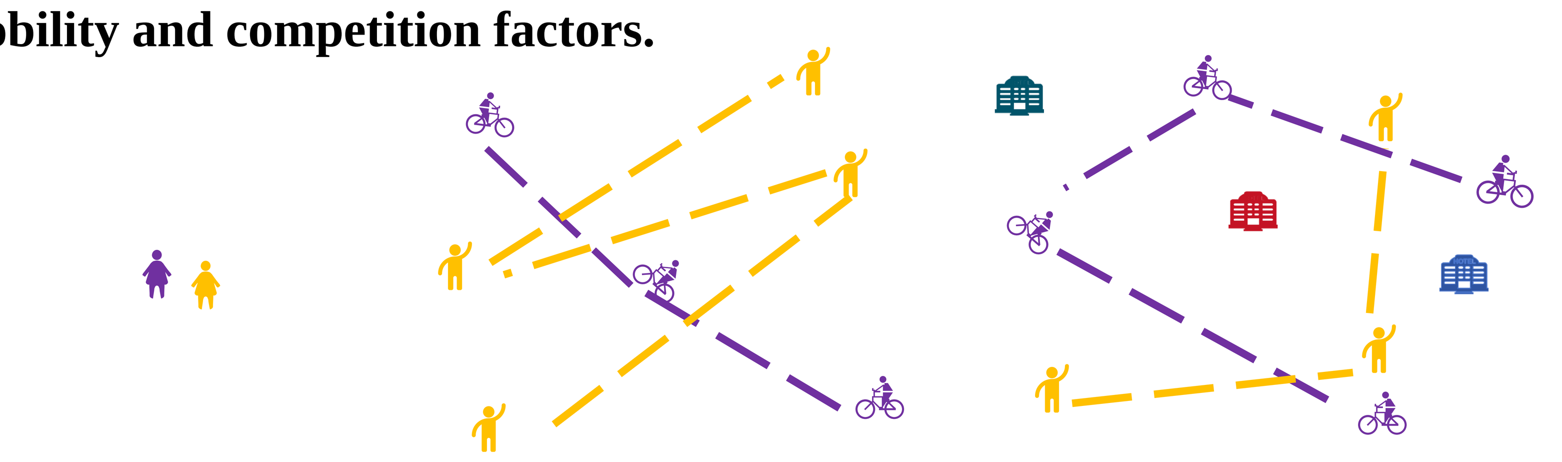
wangmeng@xpu.edu.cn, 961311016@qq.com, {hli, cuijt}@xidian.edu.cn, yangboo@stu.xjtu.edu.cn, xuetao@xpu.edu.cn

Xidian University

Introduction

Large corporations and chains, often prioritize market share over individual facility impact, leading to the development of **Collective Location Selection (CLS)**, aiming to identify ***k*** optimal sites among candidates to collectively maximize user attraction.

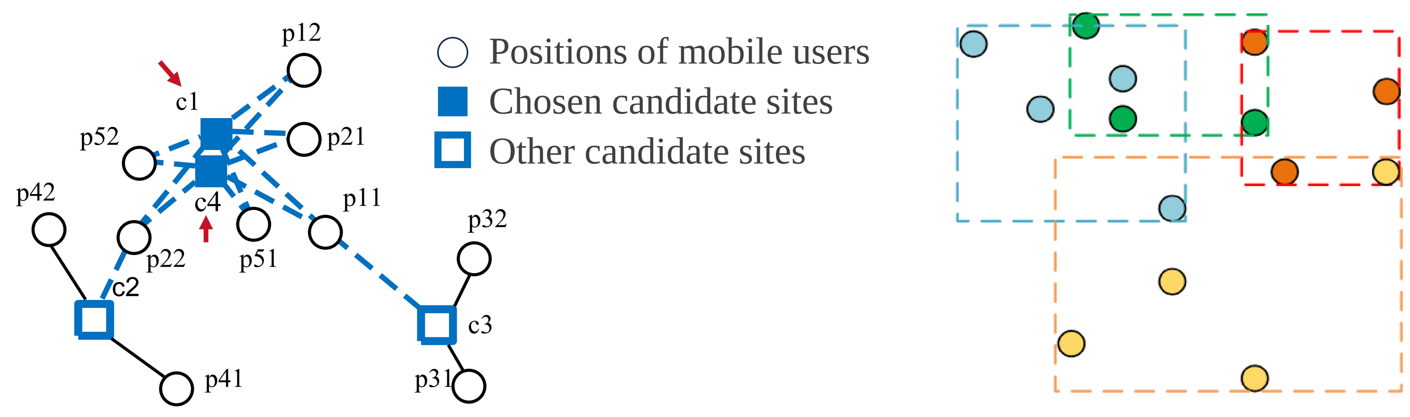
- Traditional approaches measure location attractiveness based on spatial proximity and assume users are static, i.e., each user is located at only a single position.
- Most CLS literature overlooks peer competitors due to the complexity of modeling and evaluation, which significantly impairs the effectiveness in competitive markets.
- Unfortunately, in real markets, **users are mobile** and **competitive relationships exist** among service facilities of the same type.
- To this end, this paper proposes a novel and more practical CLS problem called Mobility-oriented Competitive-based CLS **(MC²LS)**, which **considers both mobility and competition factors.**



(a) for static users (b) for moving users (c) competitive environment

Challenges in MC²LS

- How to model facility influence overlap and competitive environments?
- How to prune massive user data with highly overlapping Minimum Bounding Rectangles (MBRs) for computational efficiency?



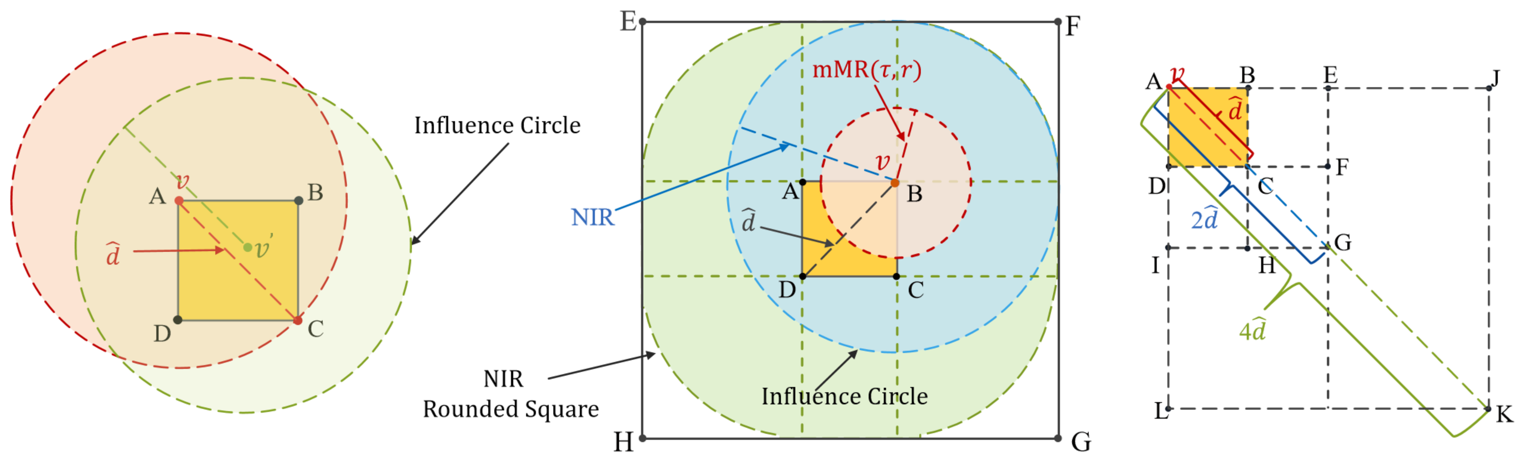
(a) C1 and C4 cover identical moving users without consideration of overlapped influence (b) Highly overlapping user MBRs

Problem Definition & Model Design

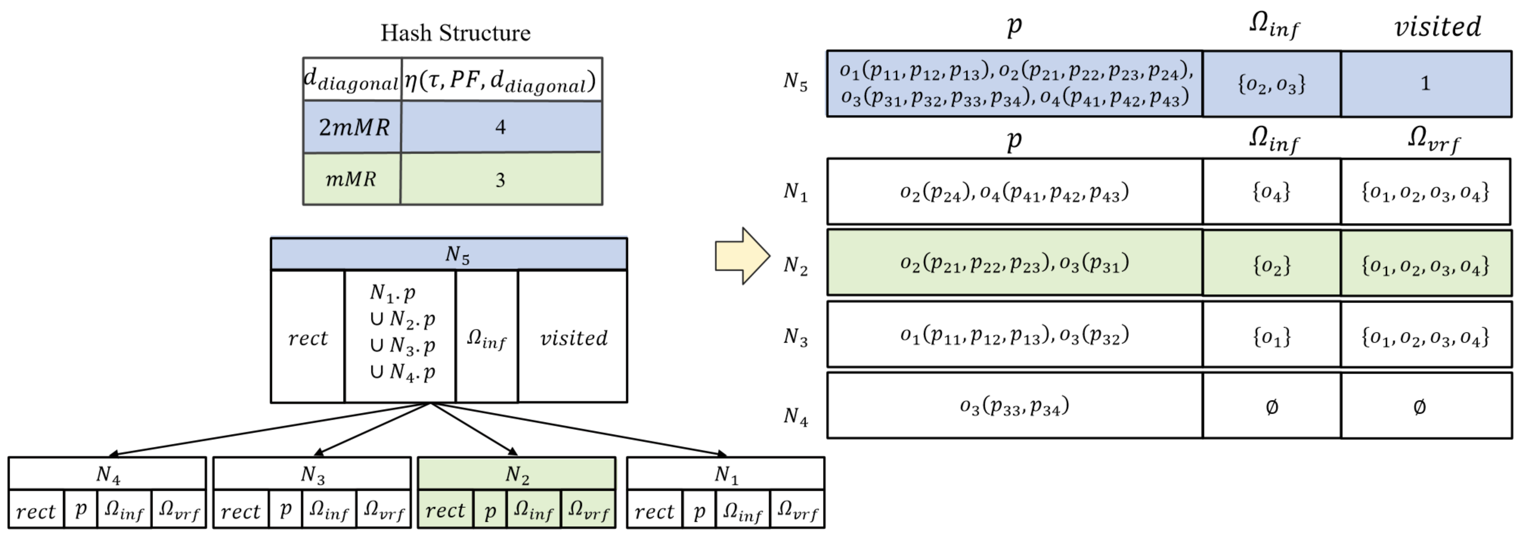
- **influence model:** For a moving user $o = \{p_1, p_2, \dots, p_r\}$, $Pr_v(p_i) = PF(d(v, p_i))$ denotes the probability that o is influenced by a facility v at position $p_i \in o$, where $PF(\cdot)$ is distance-based monotonically decreasing probability (utility) function, $d(v, p_i)$ is the distance. Given a probabilistic threshold τ for trade-offs between quality and quantity, if $Pr_v(o) = 1 - \prod_{i=1}^{r}(1 - Pr_v(p_i)) \geq \tau$, v can influence o.
- **competitive influence of a candidate site c on o:** F_o is the set of competitive facilities that influence o. $$cinf(c, o) = \frac{1}{|F_o| + 1}$$
- **competitive collective influence model:** For moving users Ω, a candidate site set G, $$cinf(G) = \sum_{o \in \Omega_G} \frac{1}{|F_o| + 1}, \quad \Omega_G = \{o | Pr_v(o) \geq \tau \land c \epsilon G \land o \epsilon \Omega\}$$
- **MC²LS problem:** To find an optimal candidate subset $G \subseteq C \land |G| = k$ to maximize its competitive collective influence.

IQuad-tree-Based Solution To MC²LS

- we propose a user-MBR-free strategy via reverse deduction of $mMR(\tau, r)$, where $mMR(\tau, r) = PF^{-1}\left(1 - (1 - \tau)^{1/r}\right)$. This allows us a new position count threshold of a distance $\hat{d}$ as $\eta(\tau, PF, \hat{d}) = 1/log_{1-\tau}\left(1 - PF(\hat{d})\right)$ to determine the influence relationship: If circle $\phi(v, \hat{d})$ (centered on v with radius $\hat{d}$) encloses $\lceil \eta(\tau, PF, \hat{d}) \rceil$ positions of user o, abstract facility v must influence o.
- Based on $\eta(\tau, PF, \hat{d})$, we construct two square-based pruning rules to filter out users for a batch of candidates: Influence Square (IS) and Non-Influence Radius (NIR).
- Even if a user o fails to meet the IS rule, o may have additional positions outside the IS region, indicating that o could possibly satisfy an enlarged $\eta(\tau, PF, \hat{d})$ with longer $\hat{d}$ (i.e., $2\hat{d}, 4\hat{d}, \dots$). This motivates us to integrate the pruning rules into Quad-tree to develop an IQuad-tree (Influence Quad-tree) for indexing users and their positions.



(a) Influence Square (b) None Influence Radius (c) The Enlargement of $\hat{d}$



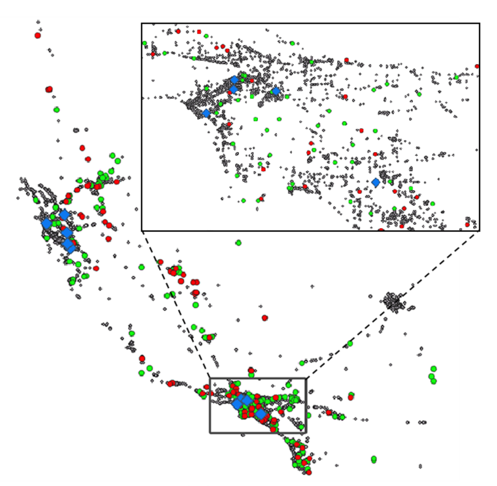
(d) The IQuad-tree index structure

Four Key Steps

- **Pruning phase:** First, index all users into IQuad-Tree. The pruning process of a facility v begins at leaf node in which it locates. IS and NIR pruning strategies are employed to identify users who are inevitably influenced by v and those not. The process is computed recursively to larger nodes until the root. Facilities in the same region as v can be batch-wise handled. The framework further incorporates the NIB pruning strategy (to prune candidates) for enhanced efficiency.
- **Verification phase:** The influence model $Pr_v(o) = 1 - \prod_{i=1}^{r}(1 - Pr_v(p_i)) \geq \tau$ is applied to determine the exact influence relationships between remaining candidate users and facilities.
- **Competitive influence value phase:** The competitive influence values for each candidate location are calculated using the competitive influence formula.
- **Updating phase:** Address the influence overlap issue greedily. When a facility is selected into the solution set, all users influenced by it are removed from the attraction sets of other candidate facilities.

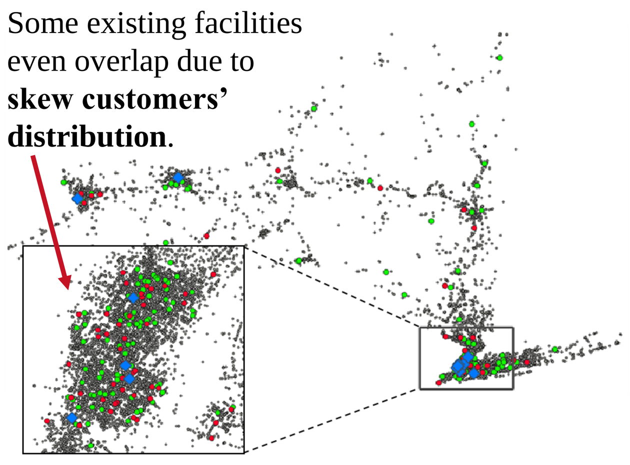
Experments

- Two real-world check-in datasets: New York (N) and California (C)
  - The distribution of users and abstract facilities is uniform in C, while it is highly skewed in N.
  - **Gray**, **green** and **red** dots indicate user positions, existing **facilities** and **candidates**. **Blue** diamonds denote the k selected candidate results.



(a) California

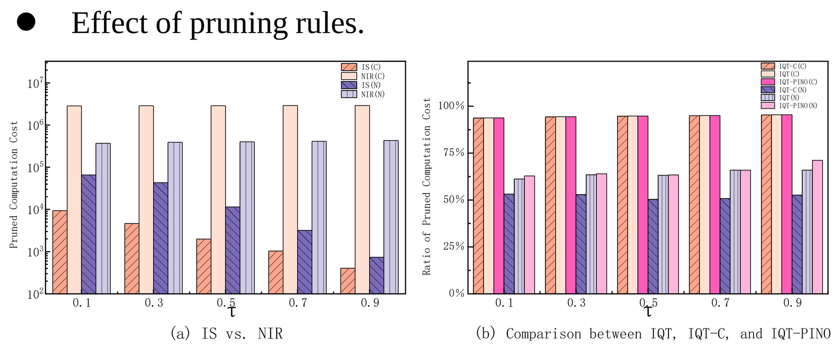
- Some existing facilities even overlap due to **skew customers' distribution.**



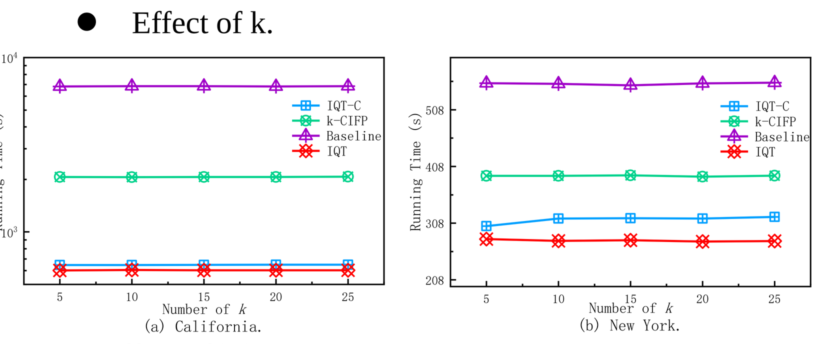
(b) New York

- We conducted sufficient experiments to explore the performance of the proposed method. The experimental results indicate that **the IQT method consistently outperforms competing approaches.**

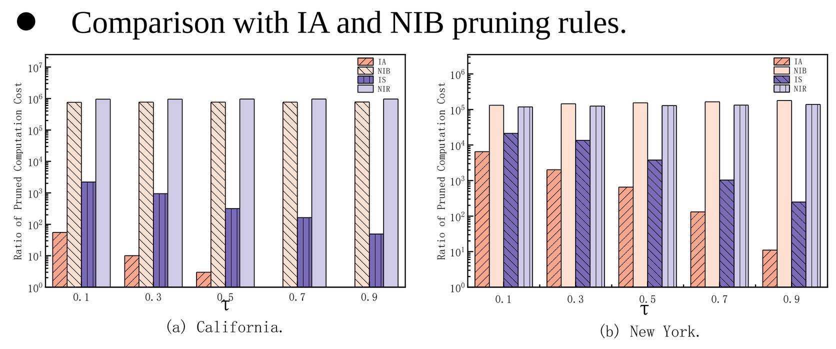
- Effect of pruning rules.



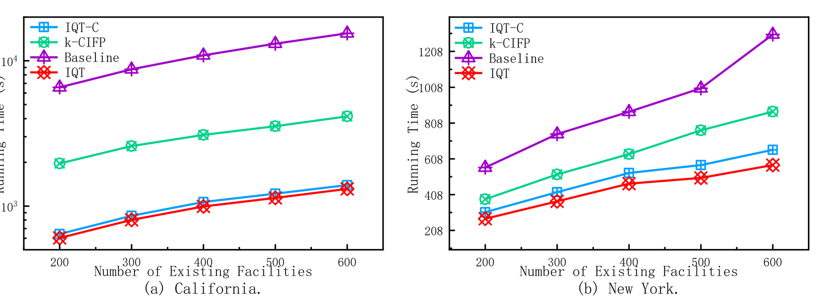
- Effect of k.



- Comparison with IA and NIB pruning rules.



- Effect of |F|.



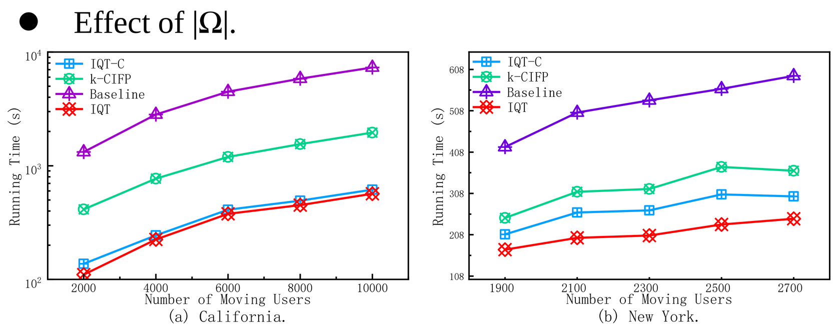
- Execution time comparison of index structures.

Dataset	IQuad-tree	IQT	R-tree	k-CIFP
C	47.70s	1,761.97s	0.04s	4,090.26s
N	3.88s	226.68s	0.04s	322.04s

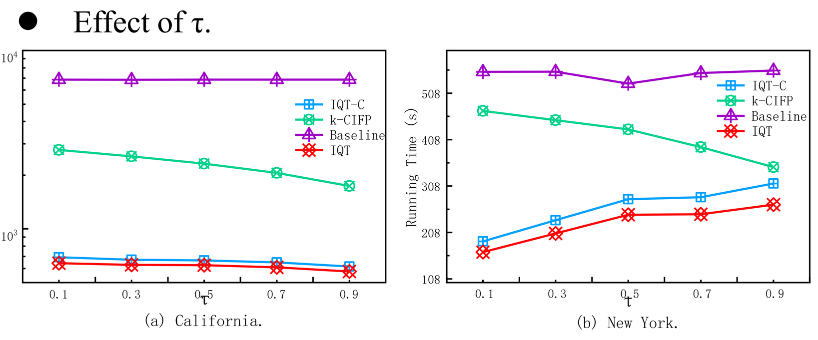
- Execution time comparison when considering IA.

# of abstract facilities	300	500	700	800	1.1K
IQT	276s	461s	646s	797s	976s
IQT-PINO	279s	469s	650s	809s	981s

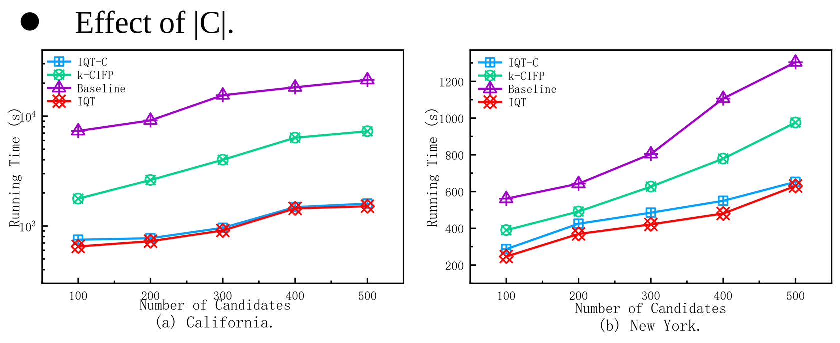
- Effect of |Ω|.



- Effect of τ.



- Effect of |C|.



- Effect of r (in C).