Preference Queries in Large Multi-Cost Transportation Networks

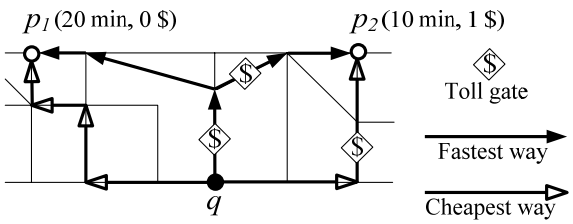
Kyriakos Mouratidis[1], Yimin Lin[1], Man Lung Yiu[2]
[1] School of Information Systems, Singapore Management University
[2] Department of Computing, Hong Kong Polytechnic University

Multi-cost Networks

- Most road networks involve multiple cost types, e.g., driving time, toll fee, Euclidean distance, etc. → ***multi-cost road networks*** (MCN)
- Proximity in MCN can be defined in different ways by different users
- Given a set of facilities P in MCN, it is unclear how to answer "Which facility is closest to location q?"
- To support proximity-based decisions, we propose ***skyline*** and ***top-k*** queries in MCNs
- Our skyline algorithms are *progressive*, and the top-k are *incremental*

Application Example

- q: port location
- p_1, p_2, ...: candidate warehouse locations
- Criteria (cost types):
  - c_1: driving time
  - c_2: toll fee
- Sensitive (e.g., dairy) products: c_1 more important
- Less sensitive goods: c_2 more important
- p_1 is cheaper to reach, while p_2 is faster

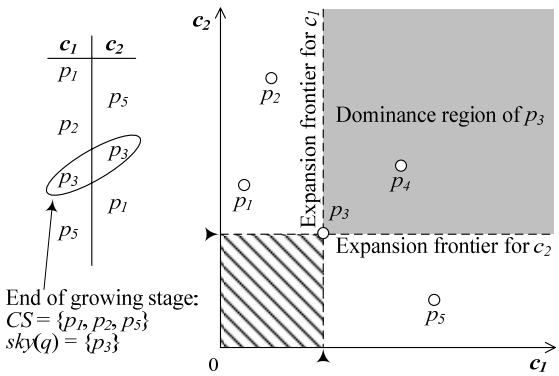


MCN Preference Queries

- **Skyline:**
  - if p is cheaper and faster to reach than p', p dominates p'
  - skyline includes all non-dominated facilities
- **Top-k:**
  - monotone function f over time and money to reach a facility p defines its aggregate cost
  - e.g., $f = 0.8 \cdot c_1 + 0.2 \cdot c_2$
  - top-k set: k facilities with smallest aggregate cost
- Naïve method:
  - compute all costs for all facilities
  - apply off-the-shelf skyline/top-k algorithm
  - problem: if there are d cost types, d complete network expansions are needed ⇒ very slow!

Skyline Processing

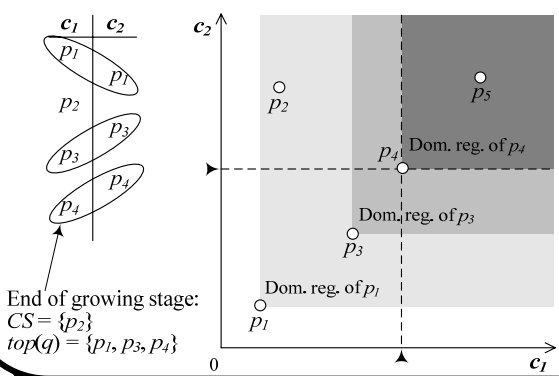
- **LSA**: limits expansions to neighborhood of query q
- Retrieves next NN for each cost (round-robin)
- ***Growing stage***: each NN is considered a ***candidate***
  - ends when 1st facility ***pinned*** (all costs computed)
- ***Shrinking stage***: d expansions continue, but...
  - no more candidates collected
  - when an existing candidate is pinned, it is included in skyline and eliminates dominated candidates



- **CEA**: shares information among d expansions to avoid multiple disk accesses for same nodes/facilities

Top-k Processing

- Growing stops when k^{th} facility is pinned



Experiments