Database Research Group

Efficient Computation of Reverse Skyline Queries

Evangelos Dellis (University of Marburg, Germany)

Bernhard Seeger (University of Marburg, Germany)

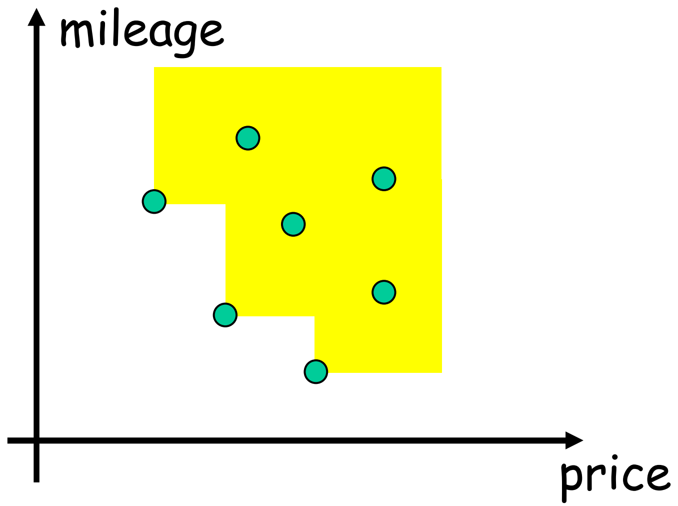
Database Research Group

Outline

- Skyline
- Dynamic Skyline Query
- Reversed Skyline Query
- Branch-and-Bound for Reversed Skylines
- Reversed Skylines with Approximations
- Experimental Results

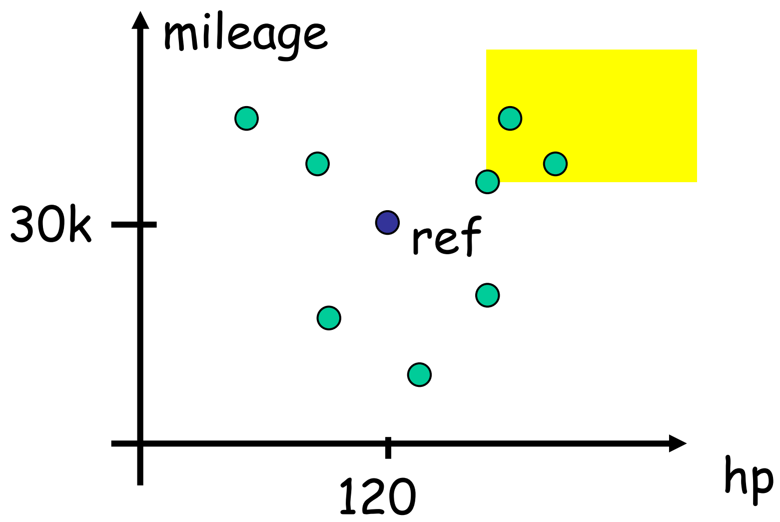
Skyline

- Important new class of queries
  - Given: a set of **d-dimensional** points
  - Result: points that are not dominated by others
    - x dominates y
      x is as good as y in all dimensions and better in at least one dimension
- Example (collection of used cars)
  - Goal: Cheapest car with lowest mileage



2. Dynamic Skyline Query

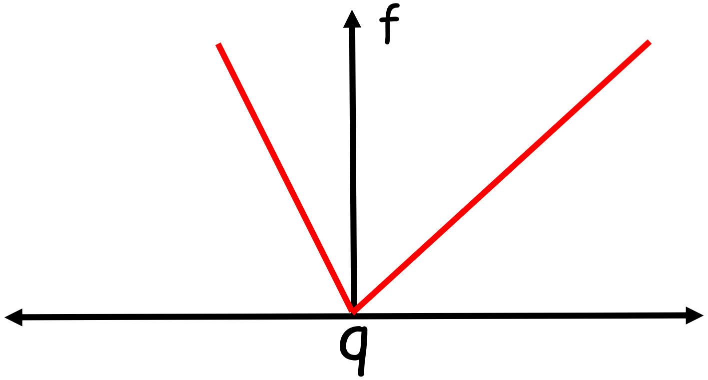
- Motivation (customer perspective)
  - ideal used car: 120 hp, 30000 km, build 2005, …
  - Find all cars that are close to customer's specification
- Skyline query relative to a **reference point ref**
  - x dominates y iff x is not farer from ref than y in in all dimensions and in at least one dimension closer to ref
- Example (Used Car Database)



Database Research Group

The distance function

- Distance function $f: R^d \rightarrow R^d$
  - $f(q) = (0,\dots,0)$
  - $f(c_1,\dots,c_{i-1},x_i,c_{i+1},\dots,c_d)$ linear decreasing in x_i, $x_i < q_i$
  - $f(c_1,\dots,c_{i-1},x_i,c_{i+1},\dots,c_d)$ linear increasing in x_i, $x_i > q_i$

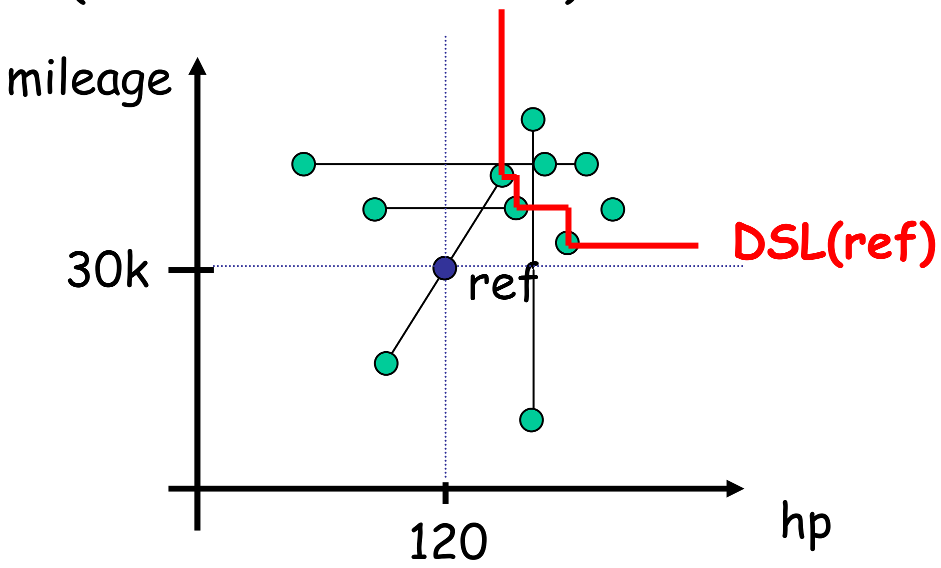


- Generalization to a more general class is possible

- Without loss of generality
  $f(x) = (|x_1-q_1|, |x_2-q_2|,\dots,|x_d-q_d|)$

Dynamic Skyline Query

- Motivation (customer perspective)
  - ideal used car: 120 hp, 30000 km, build 2005, …
  - Find all cars which are close to the customer's specification
- Skyline query relative to a **reference point ref**
  - x dominates y iff x is not farer from ref than y in in all dimensions and in at least one dimension closer to ref
- Example (Used Car Database)

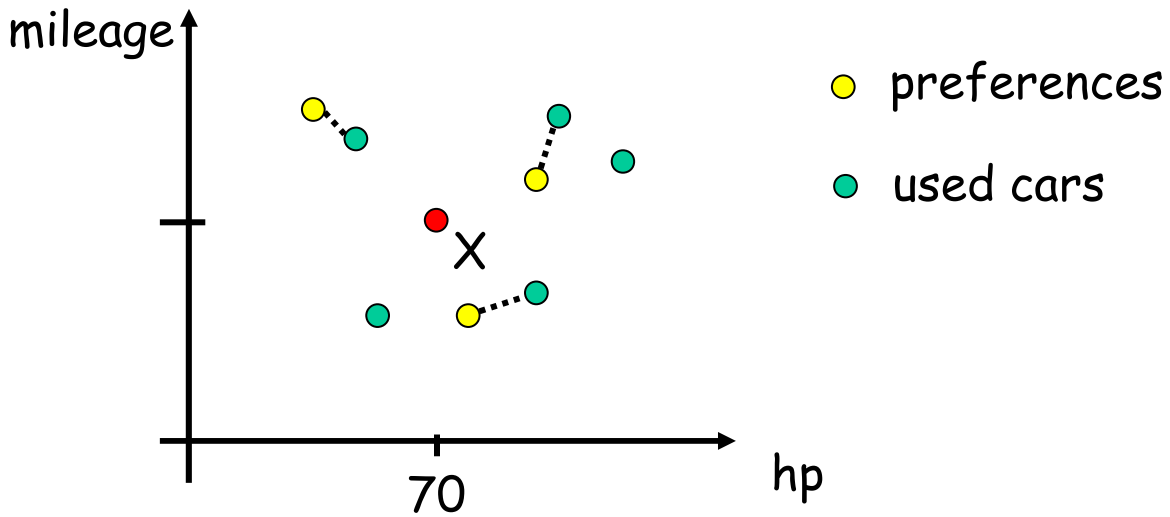


Database Research Group

3. Reverse Skyline Query

- Motivation (dealer perspective)
  - Given: the preferences of customers, the collection of used cars
  - Does it make sense to offer a car X to one of my customers?

    Car X is interesting, if it is in the skyline of a preference.



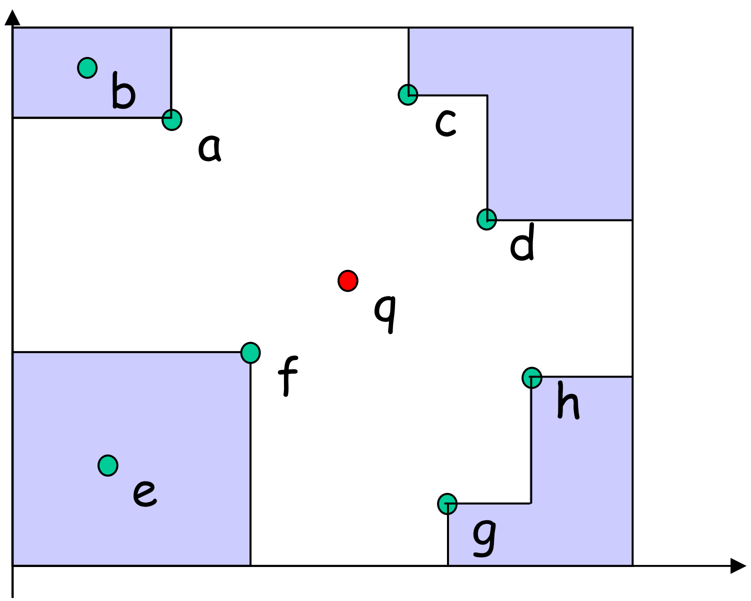
Reverse Skyline Query

- Monochromatic Problem
  - Given a set P of d-dimensional points and a query point q
- Reverse Skyline query of q
  - RSL(q) = points whose skyline contains q

- Two Algorithms
  - Assumption: R-tree on set P
  - Branch-and-bound algorithm (BBRS)
  - Reversed Skyline Search with Approximations (RSSA)

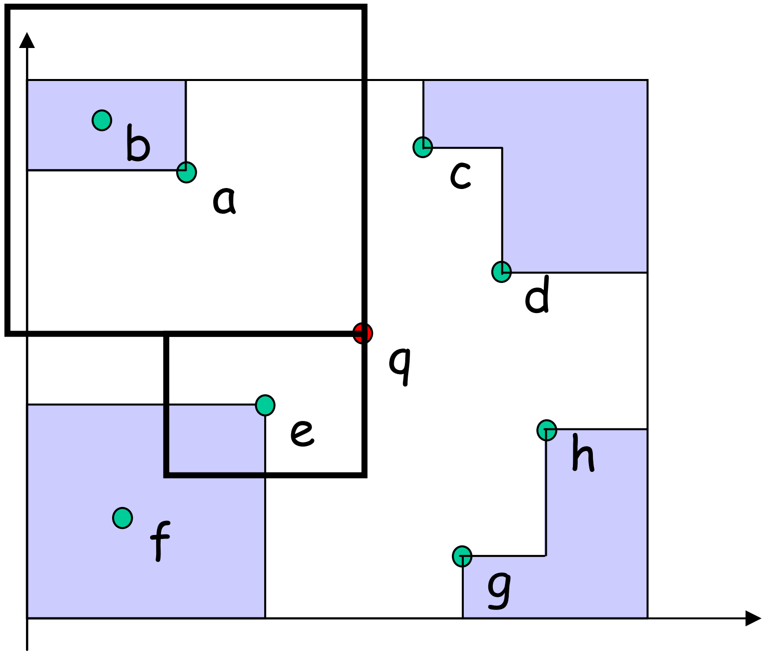
4. BBRS: Branch-and-Bound algorithm

- Assumption
  - Multidimensional index (e.g. R-tree) on point set P
- Goal
  - Processing reversed skyline of point q without transformation
- Global Skyline GSL(q)
  - points that are not globally dominated
  - **point x globaly dominates y**,
    if ε in $\{-1, 1\}^d$ exists such that for all i: $0 \le \varepsilon_i (x_i - q_i) \le \varepsilon_i (y_i - q_i)$



Important Properties

- $RSL(q) \subseteq GSL(q)$
- A point $a \in GSL(q)$ is not in $RSL(q)$ if
  there is a $b \in P$ such that for all i: $|b_i - a_i| < |a_i - q_i|$.



Algorithm BBRS

- Given: query point q, point set P
- Return the reversed skyline RSL(q)

Sketch

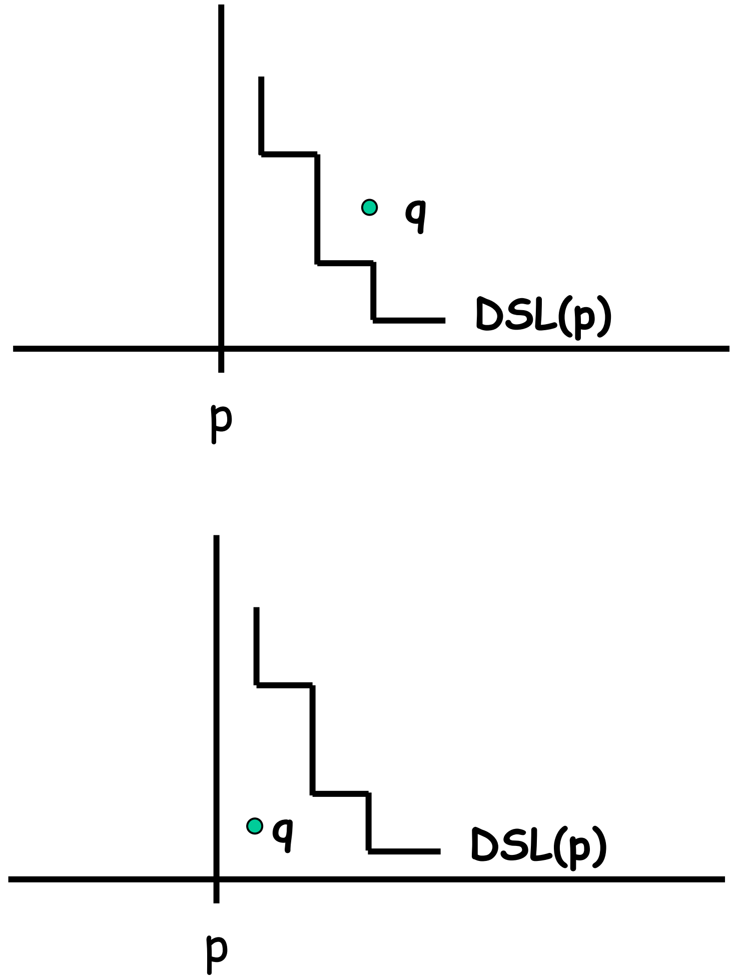
- **Candidate generation**:
  branch-and-bound computation of the global skyline GSL(q)
- For each candidate p in GSL(q) perform a **boolean window query**

Results

- ***Correctness***
- ***Minimum number of candidates***

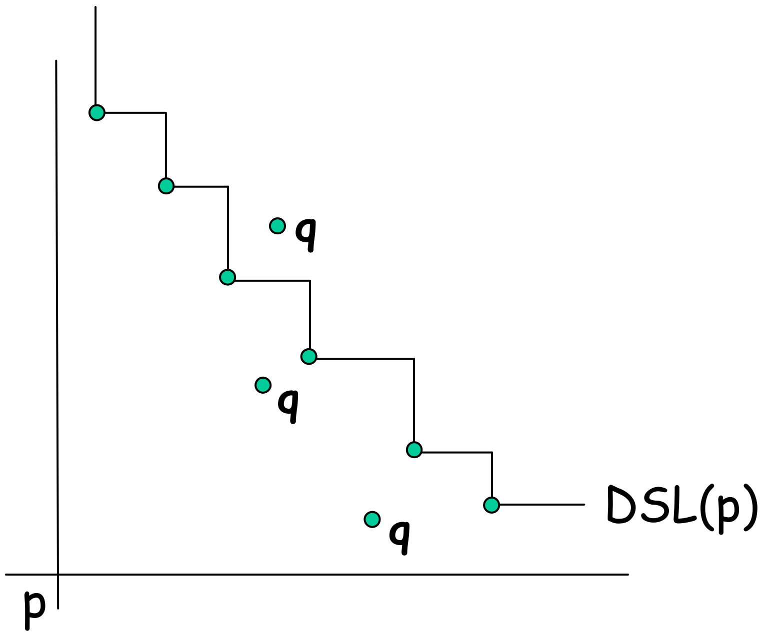
5.Reverse Skyline with Approximations

- Important property
  If any s from DSL(p) dominates q ←→ p is not in RSL(q)



Approximations

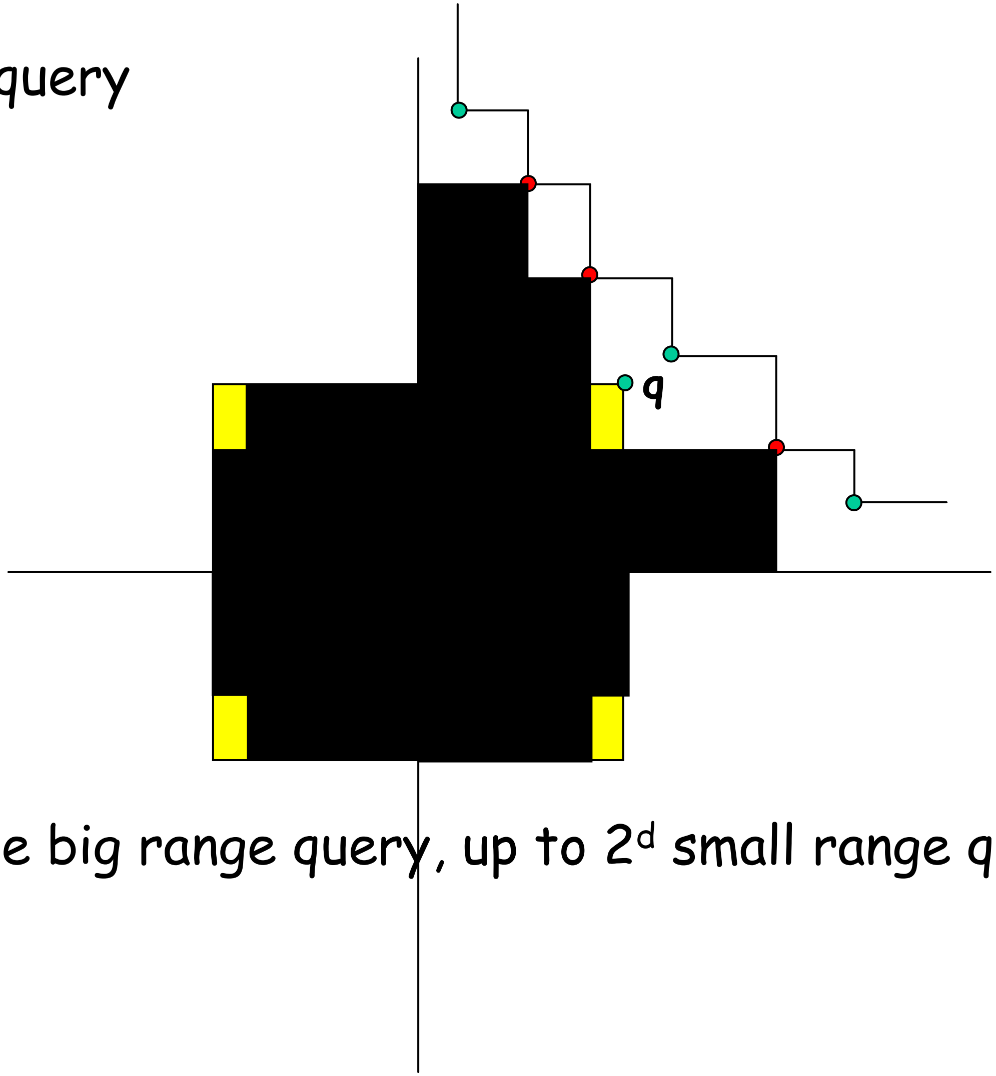
- For each p we keep a subset of DSL(p) of constant size
  - Parameter k



- Filter Step
  - If q dominates one of the samples → p is in RSL(q)
  - If a sample dominates q → p is not in RSL(q)
  - Otherwise, call the refinement step

Refinement Step

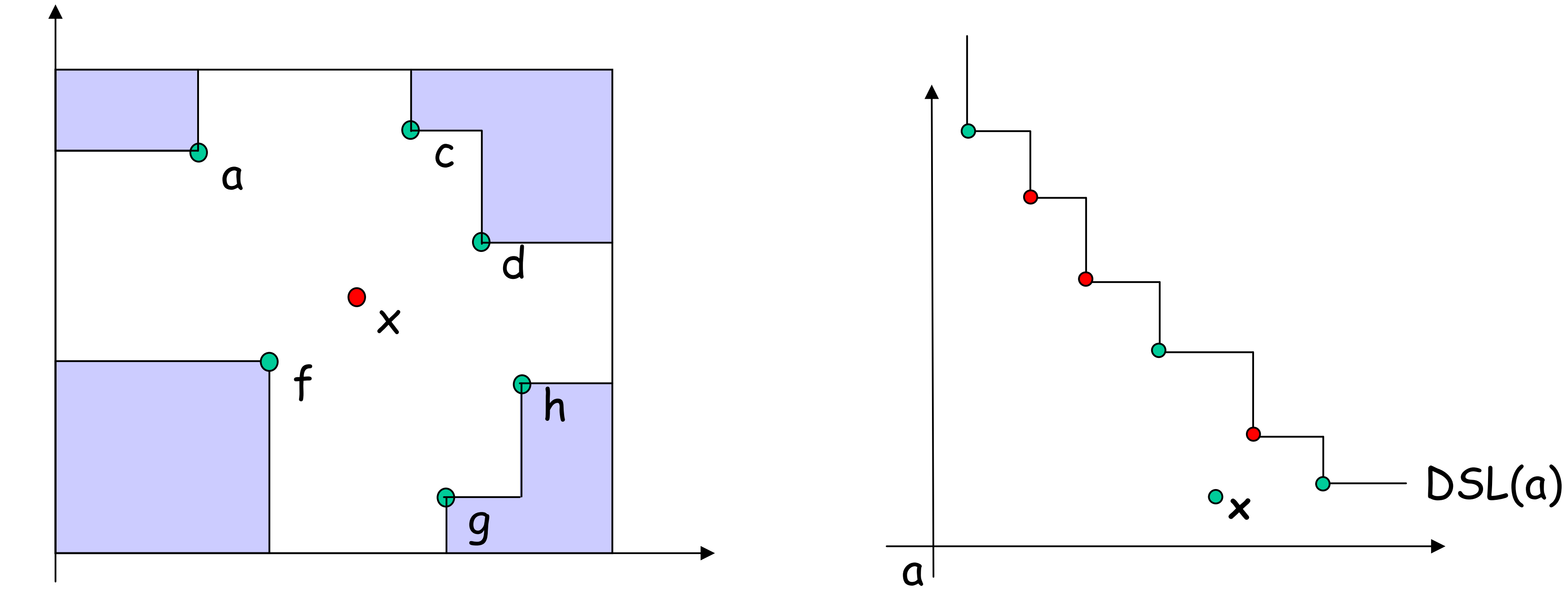
- Empty range query



- Instead of one big range query, up to 2^d small range queries

Dynamic Maintenance

- Insertion of a new point x
- Algorithm
  - Compute the global skyline GSL(x)
  - For every a ∈ GSL(x) examine the approximation of DSL(p).
    If x dominates at least one sample ➔ Update the approximation



Computing Approximations

- d=2
  - An algorithm based on the dynamic programming paradigm produces an optimal approximation.
- d>2
  - Greedy-algorithm

    Iteratively add the point with the maximum approximation gain

Related literature

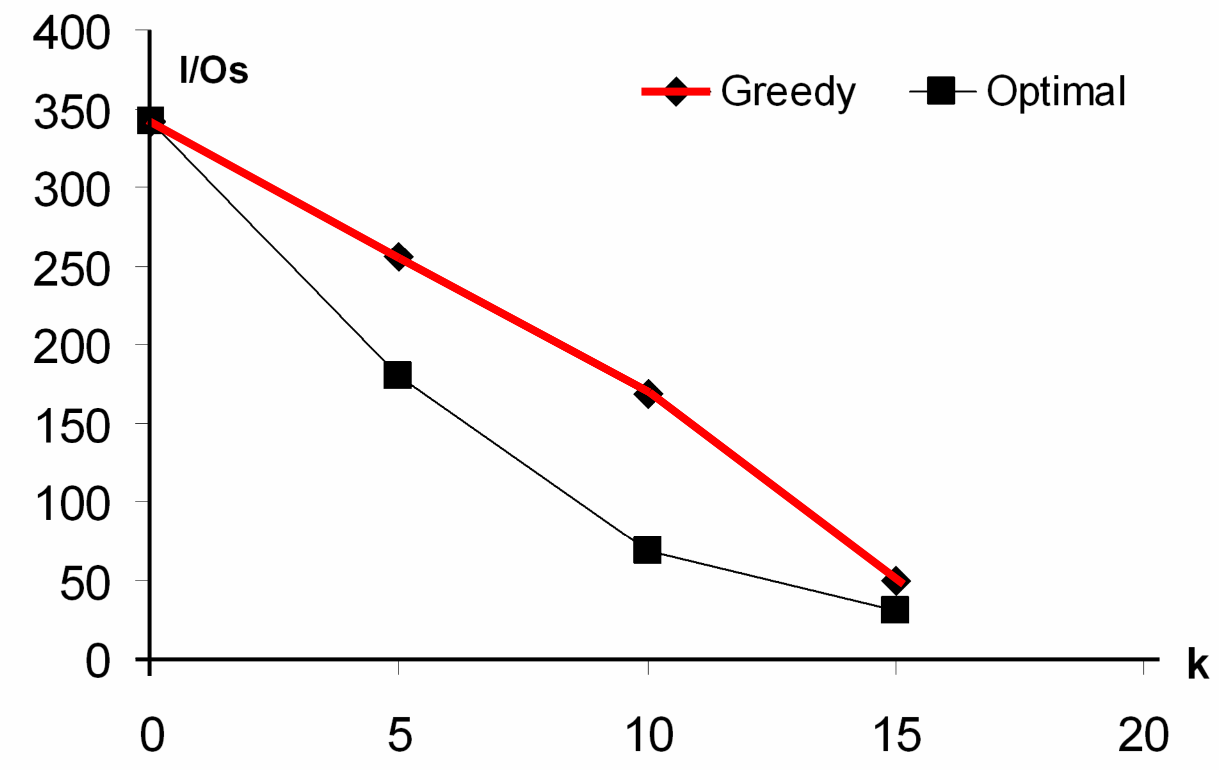
- Jagadish et al.: Optimal Histograms with Quality Guarantees, VLDB 1998
- Xuemin Lin, Yidong Yuan, Qing Zhang, Ying Zhang : Selecting Stars: The k Most Representative Skyline Operator, ICDE 2007

6. Experiments

- Data sets
  - Real Data
    - CarDB: d = 2; N = 50000
    - NBA: d = 4; N = 17000
  - Synthetic Data
    - Uniform distribution: d=2,…,4; N = 80000
    - Cluster distribution: d = 2,…,4; N = 80000
- Queries
  - 100 reversed skyline queries
- Implementation
  - XXL library (newest version on request)

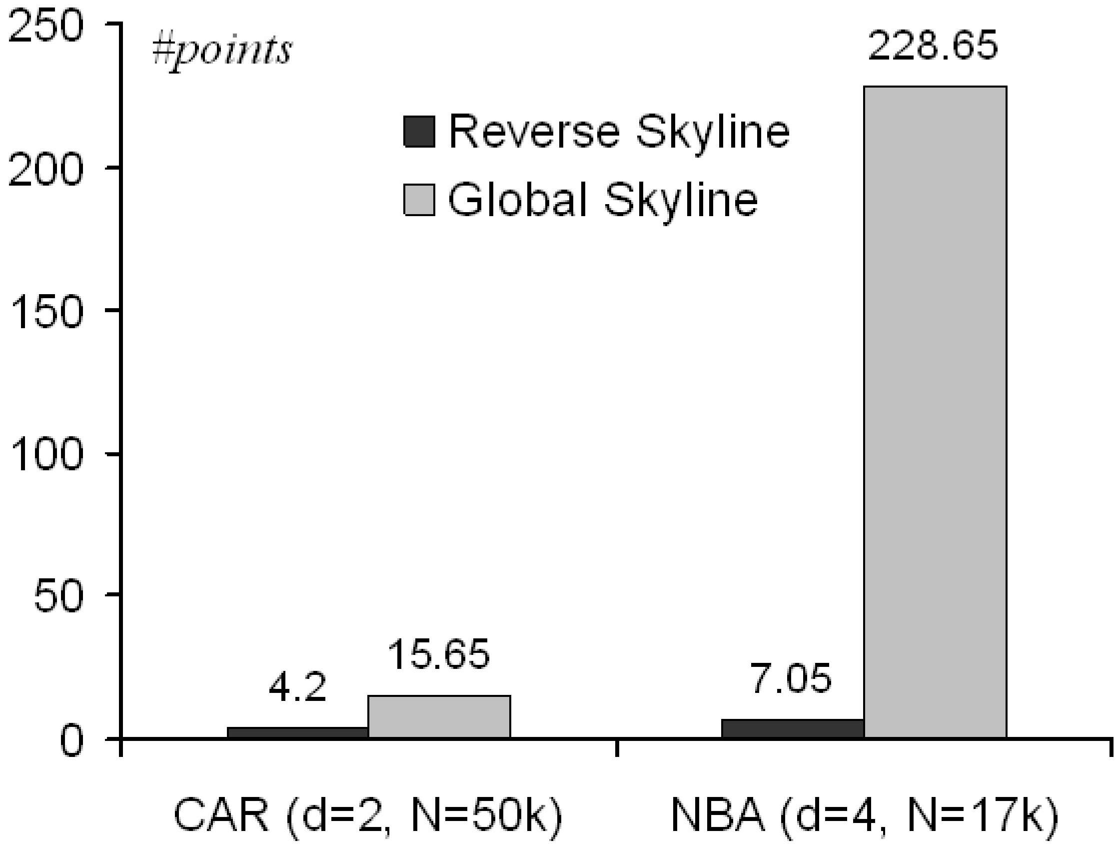
RSSA algorithm

- Performance as a function of k



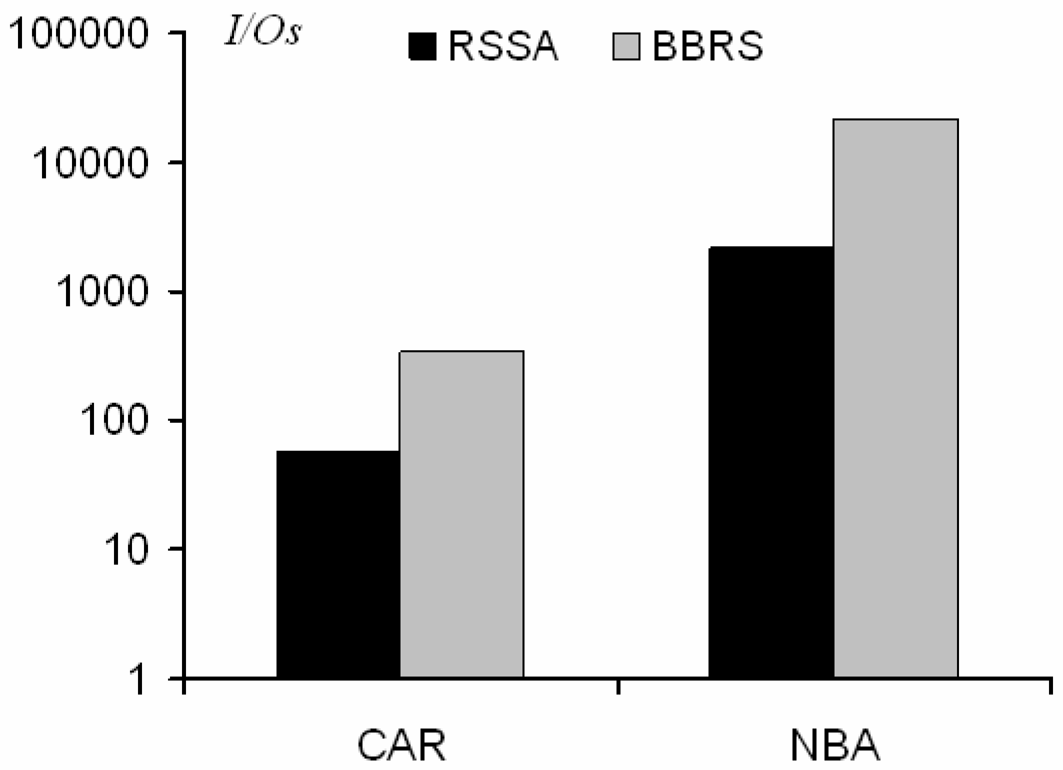
Size of the reversed skyline

- in comparison to the size of the global skyline



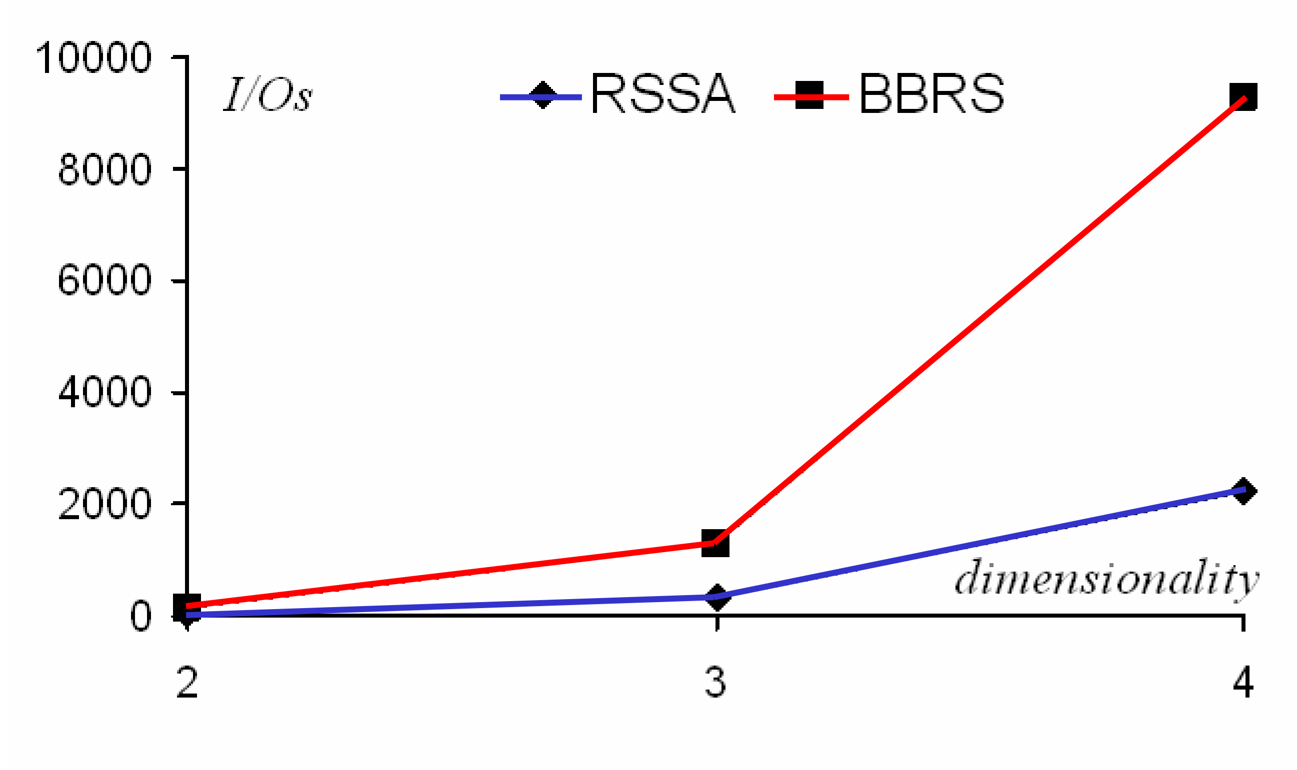
Comparison RSSA vs. BBRS

- Average number of I/Os (logarithmic scale)



Comparison RSSA vs. BBRS

- Performance as a function of dimensionality



Database Research Group

Conclusions

- Reverse Skylines are important for finding interesting points
  - Dealer perspective:
    What kind of items are interesting to my customers?
- Two Algorithms
  - BBRS
    - Adaptation of the original BBS algorithm
  - RSSA
    - Filter-and-refinement paradigm
    - Preprocessing approximations of skylines
    - Updates are expensive
- Future Work
  - Accurate Approximation of skylines for d > 2
  - Bichromatic Reversed Skylines