A Taxonomy of Range Queries

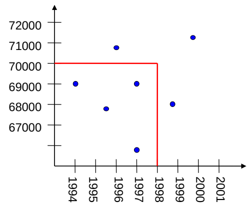
Mihai Pătraşcu

at&t

MADALGO Summer School 2010

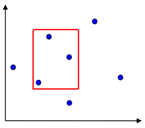
Monday, Morning 1

```
SELECT * FROM employees
WHERE salary <= 70000
  AND startdate <= 1998
```



Query = ?

Database = *n* points in the plane
Query = axis-parallel rectangle

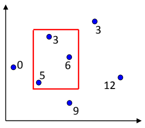


- *existential queries:* is there any point inside?
- *counting queries*: how many points inside?
- *reporting queries*: list the points inside.

NB: The query time is $f(n) + k \cdot g(n)$, where k = # output points.

Weights

Database = *n* points in the plane ***with weights***
Query = axis-parallel rectangle

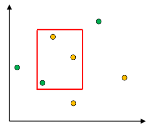


- *weighted counting*: total weight inside rectangle
- *range min/max query* (RMQ)
- *range median query*
- *top-k reporting:* report the heaviest *k* guys

- *k* given, output in any order
- *k* given, output sorted
- output one at a time, sorted, until stopped

Colors

Database = n points in the plane ***with colors***
Query = axis-parallel rectangle



- *colored counting:* how many *distinct* colors?
- *colored reporting*: list the distinct colors
- *top-k colored reporting*

🟡 = Carlsberg

🟢 = TUBORG

Dynamism

The problem can be:

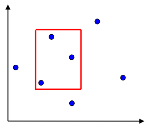
- static: space vs. query time [vs. preprocessing time]
- dynamic: update time vs. query time [vs. space]
  - incremental only
  - decremental only
- offline
- parametric/kinetic

Orthogonal Range Queries

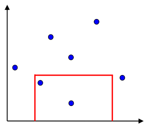
Database = *n* points in *d*-dimensional space
Query = box $[a_1, b_1] \times \cdots \times [a_d, b_d]$

- *d*≥1
- *d*=constant, so 2^d=O(1)

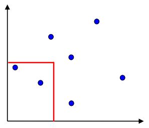
k-sided queries = exactly 2*d*-*k* values in $(a_1, a_2, \ldots, a_d)$ are zero.



general = 4-sided



3-sided



dominance = 2-sided

The Universe

The universe (space of coordinates) can be:

- {0, .., *u*} where *u* = MAX_INT
  = the largest number that fits in a machine word
- {1, .., *n*} *“rank space”*

NB: general universe ↦ rank space
using just 2*d* binary searches

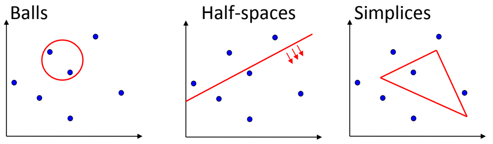
- $\{1, .., n^{1/d}\}$

12	3	5	91	1
5	7	2	0	43
22	13	1	77	2
8	4	4	3	17

Non-Orthogonal

Database = *n* points in *d*-dimensional space

Query ranges:

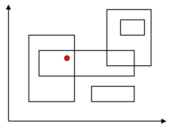


… etc

Stabbing

Database = set of ranges

Query = one point



Internet routing:

- "packets going to IP range [*a*,*b*] are sent to link X"
- rules for source/destination ranges
- rules might have priorities

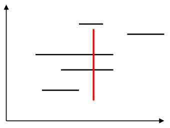
Range vs. Range

Database = set of ranges

Query = one range

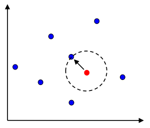
- *intersection queries:* who intersects the query?
- *containment queries:* who contains/is contained by query?

E.g. orthogonal
segment intersection



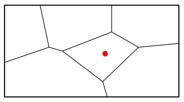
Other Geometric Problems

- nearest neighbor search



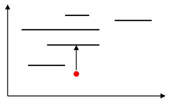
Related to:
ranges = balls

- point location



stabbing with
disjoint ranges

- ray shooting



segment intersection

Models of Computation

Serious models:

- the standard model: Word RAM
  word size = Ω(lg *n*) to store indices, pointers
- external memory:
  access memory in pages of *B* words at a time

NB: reporting time O(lg *n* + *k*)
very different from O(lg *n* + *k*/*B*)

- cache oblivious:
  same as external memory, but you don't know *B*

Models of Computation

Serious models: standard RAM, external memory

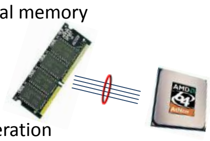
Semi-serious model: Real RAM

- machine words store values from ℝ
- irrelevant for orthogonal stuff (see: rank space)
- convenient for non-orthogonal stuff (forget precision!)
- sometimes gives Computational Geometry a bad reputation…

Models of Computation

Serious models: standard RAM, external memory

Semi-serious model: Real RAM



Lower bound models:

- cell probe: RAM with any exotic operation
- weak models:
  - for reporting: pointer machine
  - for counting: arithmetic models (group, semigroup)
  - etc