Notes on Computational Geometry and Data Structures

2008

Geometric Data Structures and CGAL

Data Structure	CGAL
Interval Tree	no
Priority Search Tree	no
Segment Tree	up to 4 dimensions
Range tree	up to 4 dimensions no fractional cascading
k-d Tree	d dimensions

There is an introduction of these data structures in [4].

Interval Tree

Use: Report the k intervals out of n that contain a query point.

Performance:

construction	$O(n \log n)$
space	$O(n)$
time	$O(\log n + k)$

Applications:

- Interval intersections in one dimension
- Orthoganal intersections (used as y-structure)

Priority Search Tree

Use: Report k points out of n in a semi-unbounded query range of the form $(-\infty : x] \times [y : y']$.

Performance:

construction	$O(n \log n)$
space	$O(n)$
time	$O(\log n + k)$

Applications:

- Report all intervals which intersect to a query interval in one dimension.
- Orthoganal and interval intersections.
- Performance Analysis
- VLSI routing

Segment Tree

Use: Report the k segments out of n that intersect a vertical query segment.

Performance:

construction	$O(n \log n)$
space	$O(n \log n)$
time	$O(\log^2 n + k)$

Applications:

- Computation of the total volume of simple polygons in 2-dimensions.
- Hidden line/surface removal.
- Orthogonal intersection in d-dimensions.

Range Tree

Use: Report k points out of n in a d-dimension range query.

Performance:

construction	$O(n \log^{d-1} n)$
space	$O(n \log^{d-1} n)$
time	$O(\log^{d} n + k)$
time (fractional cascading)	$O(\log^{d-1} n + k)$

Applications:

- Multi-dimensional queries.

k-d Tree

Use: Report k points out of n in a d-dimension range query.

Performance:

construction	$O(dn \log n)$
space	$O(dn)$
time	$O(n^{1-\frac{1}{d}} + k)$

Applications:

- Multi-dimensional queries.
- Nearest neighbour queries.

Types of Multi-dimensional queries

Given n d−dimensional points

- Exact match queries
- Partial match queries
- Region queries

Output: k points

Region queries

The general case.

	k-d tree [1]	range tree	optimal [3, 2]
const	$O(n \log n)$	$O(n \log n)$	
space	$O(n)$	$O(n \log n)$	$O(n \log^{\epsilon} n)$
time	$O(\sqrt{n} + k)$	$O(\log^2 n + k)$	$O(\log n + k)$
const	$O(dn \log n)$	$O(n \log^{d-1} n)$	
space	$O(dn)$	$O(n \log^{d-1} n)$	$O(n(\log n / \log \log n)^{d-1})$
time	$O(n^{1-\frac{1}{d}} + k)$	$O(\log^{d-1} n + k)$	$O(\log^{c} n + k)$

for a fixed $\epsilon > 0$ and any constant c

Exact & Partial match queries

Exact match queries (search for a point)

- The region is a point.
- time: $(d + \log n)$

Partial match queries (t keys)

- The region is $k - t$ dimensional hyperplane.
- time: $(n^{1-t/d} + k)$

Nearest neighbour problem

k-d tree [1]

- worst case (point on a circle):
  # inspections = O(n)
- average (special case, hyperrectangles = hypercubic):
  # inspections = $O(\log n)$
  # distance calculations = exponential in dimension d

other techniques for nnp ...

References

- J. L. Bentley.
  Multidimensional binary search trees used for associative searching.
  Commun. ACM, 18(9):509–517, 1975.
- B. Chazelle.
  Lower bounds for orthogonal range searching: part i. the reporting case.
  J. ACM, 37(2):200–212, 1990.
- B. Chazelle.
  Lower bounds for orthogonal range searching: part ii. the arithmetic model.
  J. ACM, 37(3):439–463, 1990.
- M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf.
  Computational Geometry: Algorithms and Applications.
  Springer-Verlag, second edition, 2000.