Randomized Geometric Algorithms

Ken Clarkson
AT&T Bell Labs
Murray Hill, NJ

Outline

- Trapezoidal diagrams
- Randomized divide-and-conquer
- Convex hulls
- Randomized incremental algorithms

Trapezoidal Diagrams

Given a set S of
n line segments, with
A intersection points,
its TD $\mathcal{T}(S)$ has $\Theta(n + A)$ regions.

Results

Suppose S forms K known chains.
How much work is needed to find $\mathcal{T}(S)$,
and how quickly can the diagram be found?

work		time/ $\lg n$	
$\Omega(K \lg n + A$	$+n)$	$\Omega(1)$	
$K \lg n + A$	$+n \lg^* n$	$\lg \lg n \lg^* n$	CCT
A	$+n \lg n$	1	CCT
	n	$\lg \lg n \lg^* n$	CCT; simple
	n	•	Cha; simple
	$n \lg^* n$	•	CTVW, S
	n^2	1	G, HJW
$A \lg n$	$+n \lg^2 n$	1	G
A	$+n \lg n$	1	red/blue; GSG

Randomized divide-and-conquer [CS]:

- take $R \subset S$ random of size r;
- compute $\mathcal{T}(R)$;
- for $T \in \mathcal{T}(R)$,
  find segments S_T meeting it (insertion);
- compute $T \cap \mathcal{T}(S_T)$ for $T \in \mathcal{T}(R)$;
- merge pieces to find $\mathcal{T}(S)$;

We can use “slow” algorithms for $\mathcal{T}(R)$ and the $T \cap \mathcal{T}(S_T)$, since:

Each trapezoid meets
$O(n/r)$ segments, on average, and
$O(n/r)\log r$ with high probability.

Moreover, $\mathcal{T}(R)$ has expected $O(r + Ar^2/n^2)$ trapezoids.

For parallel work $O(A + n\log n)$,
choose $r = n/\log n$;
compute $\mathcal{T}(R)$ using $O(\log n)$ slack [G],
and do $O(n/r)^2$ work [G][HJW]
for subdiagrams.

Why $O(n/r)\log r$ segments/trapezoid?

$T \in \mathcal{T}(S)$ is in some $\mathcal{T}(A)$ for $|A| < 4$,
so $O(n^4)$ trapezoids to consider.

A trapezoid meeting αn segments
has $\binom{n-\alpha n-4}{r-4}/\binom{n}{r} \approx (r/n)^4(1-\alpha)^{r-4}$
chance to be in $\mathcal{T}(R)$.

So the probability that a trapezoid meeting
$K(n/r)\log r$ segments
is in $\mathcal{T}(R)$ is
$O(r^4)(1 - K(\log r)/r)^{r-4}$,
less than about $e^{K\log r - 4\log r} = 1/r^{K-4}$.

Using connectivity
(a.k.a., simple polygon triangulation)

To insert, walk through $\mathcal{T}(R)$ and S;

This gives $O(n \log\log n)$ expected time, with $r = n/\log n$ and average subproblem size $O(\log n)$.

For $O(n \log^* n)$ work:
For subsets
$S^1 \subset S^2 \subset \cdots \subset S^{\log^* n} = S$,
with $|S^1| = n/\log n$, $|S^2| = n/\log\log n$,
  $|S^i| = n/\log^{(i)} n$,
compute $\mathcal{T}(S^i)$ using $\mathcal{T}(S^{i-1})$.

In parallel, the insertion is done by
many parallel walks through subchains.

The main problem: while every trapezoid of $\mathcal{T}(R)$ meets few segments,

a segment may meet many trapezoids.

How to handle *bad* segments
that meet $\Omega(\log n)$ trapezoids?

There are $O(n/\log n)$ bad segments,
on average: to insert them,
compute their intersections
with the visibility edges using algorithm [GSG].

One way to show that
each $T \in \mathcal{T}(R)$ meets $< 4n/(r+1)$ segments
on average:

pick $x \in S \setminus R$ at random.
How many $T \in \mathcal{T}(R)$ does it meet?

That is, how big is
$\mathcal{T}(R) \setminus \mathcal{T}(R')$, where $R' = R \cup \{x\}$?

Since $|A| + |B \setminus A| = |B| + |A \setminus B|$,
$E|\mathcal{T}(R) \setminus \mathcal{T}(R')|$
$= E|\mathcal{T}(R)| - E|\mathcal{T}(R')| + E|\mathcal{T}(R') \setminus \mathcal{T}(R)|$

That is, it's enough to know the number of trapezoids created when x is added
$=$ number deleted when x deleted from R'
$< 4E|\mathcal{T}(R')|/(r+1)$.
since T incident to < 4 segs,
each with prob $1/(r+1)$ of being x.

So the number of seg/trap intersections is expected $< 4(n-r)E|\mathcal{T}(R')|/(r+1)$.

Convex hulls

Given a set
S of
n points in
d dimensions,
maintain the convex hull of $R \subset S$.

We'll analyze under assumptions
implying R is random:
e.g., add points in random order.

When $x \in S \setminus R$ is added to R,
edges *visible* to x
are no longer in the hull.

Visibility testing requires
a line-point orientation test.

The algorithm:
maintain a *triangulation* of $\operatorname{conv} R$.

To **update**
when adding x, and
edge $\overline{ab}$ is visible to x,
include $\triangle abx$.

The (asymptotically) hard problem is
search: find the edges visible to x.

One technique:
walk through the triangulation
from a known point (the origin).

Another search scheme:
starting at origin, look at all triangles
whose *base edges* are visible to x.

Base edge:
When x is added and Δabx created,
$\overline{ab}$ is a base edge.

Visibility now means:
edge visible to y when edge created.
(If y added instead of x, get Δyab.

Analysis

...of second scheme,
under random insertions $x_1, \ldots, x_i$

space: what is $E|\mathcal{T}|$,
the expected number of triangles in
triangulation $\mathcal{T}$?

time: what is the expected number
of triangles visited for x_i?

Note $R_i = \{x_1 \ldots, x_i\}$
is a random subset of S,
x_i is a random element of R_i and S.

Let
$f_i =$ expected number of current hull edges of
R_i.

Space:

(Look at general dimension d,
since $d = 2$ trivial.)

= expected number of
(current and old convex hull) facets.

Count the expected number of facets
created when x_j is inserted, for $j \le i$,
and sum over j.

facets created when x_j added are
facets incident to x_j in $\operatorname{conv} R_j$.

d vertices/facet, implies
df_j expected vertex-facet incidences, implies
df_j/j facets incident to x_j, expected.

So $E|\mathcal{T}| = \sum_{j \le i} df_j/j$.

Time:

Let $H = H_{i-1} = H(x_1, \ldots, x_{i-1})$
denote the set of hull facets for insertions $x_1, \ldots, x_i$.
(So $|H| = |\mathcal{T}|$.)

Let $H' = H(x_i, x_1, \ldots, x_{i-1})$.

then $H \setminus H'$ is
the set of facets of H visible to x_i, and search time is $O(d)E|H \setminus H'|$.

Since $|H| + |H' \setminus H| = |H'| + |H \setminus H'|$,
$E|H| + E|H \setminus H| = E|H'| + E|H \setminus H'|$.

the space analysis gives
$E|H| = \sum_{j \leq i-1} df_j/j,\ E|H'| = \sum_{j \leq i} df_j/j$,
so $E|H \setminus H'| = E|H' \setminus H| - df_i/i$.

We want $E|H' \setminus H|$,
the expected number of facets incident to x_i in the set $H' = H(x_i, x_1, \ldots, x_{i-1})$.

As in the space bound, count the number expected when x_j is inserted, and sum over j.

The facets are incident to x_i and x_j, and facets of $\operatorname{conv} R'_i$, $R'_i = \{x_i, x_1, \ldots, x_j\}$.

$\binom{d}{2}$ vertex pairs/facet implies
$\binom{d}{2} f_{j+1}$ pair-facet incidences, implies
$\binom{d}{2} f_{j+1} / \binom{j+1}{2}$ facets/pair.

So

$$E|H' \setminus H| = \sum_{j \le i-1} \frac{d(d-1)}{(j+1)j} f_{j+1}$$

or

$$\sum_{j \le i} \frac{d(d-1)}{j(j-1)} f_j$$

The expected number of location tests for x_i is less than twice this.

Conclusions

We've seen randomization for

- parallel algorithms, divide-and-conquer;
- dynamic algorithms, incremental;
- data structures
- TD, CH/VD, LP, MSTs, BSPTs, NN, HSE,. . .

What about

- determinism; [M,M,M,M,. . .]

- simple $O(n)$ triangulation;

- realistic machine models for parallel algorithms;

- tail estimates;

- self-adjustment;