

Three Problems
about
Dynamic Convex Hulls

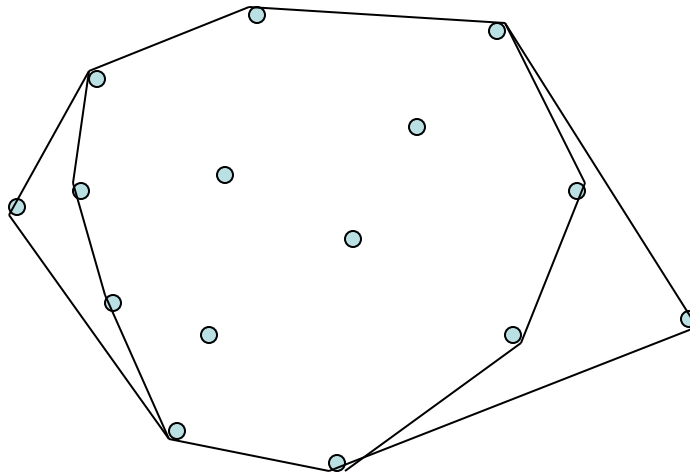
Timothy Chan

School of CS

U of Waterloo

I. Dynamic 2D Convex Hulls

- Maintain point set in 2D under insertions & deletions s.t. we can answer queries about the convex hull (CH)



History

- Overmars, van Leeuwen '80 ("Hull Tree")
 $O(\log^2 n)$ update time, $O(\log n)$ query time
- C. [FOCS'99]
 $O(\log^{1+\epsilon} n)$ update (amort.), $O(\log n)$ query
- Brodal, Jacob [SWAT'00]
 $O(\log n \log \log n)$ update (amort.), $O(\log n)$ query
- Brodal, Jacob [FOCS'02]
 $O(\log n)$ update, $O(\log n)$ query
[current full page > 100 pages!]

Problem Solved... NOT!

- C. & Brodal, Jacob apply to "Type-A" Queries:
 - decide whether CH intersects a query line
 - extreme pt along a query direction
 - neighbor of a query vertex
 - tangent to a query pt
- But not "Type-B" Queries:
 - decide whether CH contains a query pt
 - intersect CH with a query line
 - outer/separating tangents between 2 disjoint CHs

What's Known for Type-B Queries

- C. [FOCS'99]

$O(\log^{3/2} n)$ update (amort.), $O(\log^{3/2} n)$ query

- Open: better??

- New Result

$O(\log^{1+\varepsilon} n)$ update (amort.), $O(\log^{1+\varepsilon} n)$ query (rand.)

[or $O(\log n \cdot 2^{O(\log \log n \log \log \log n)^{1/2}})$ update, query!]

Technique

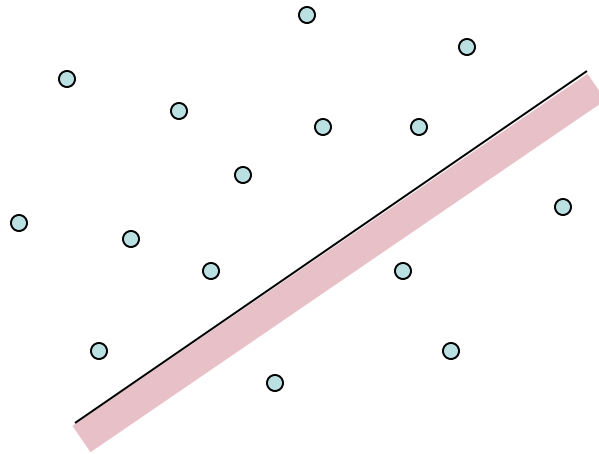
- Work in dual \Rightarrow LP queries
- Based on C. [FOCS'99]:
 - Logarithmic Method + Delete-Only DS
 - + Interval Tree
 - + Bootstrapping

NEW: + Randomized LP-type Alg'm

(where "basis evaluation" oracle is implemented by bootstrapping)

II. Dynamic 2D Halfplane Range Reporting

- Maintain point set S in 2D under insertions & deletions s.t. we can report all k pts inside a query halfplane



[generalizes dynamic CH ($k=0$)...]

What's Known

- **Overmars, van Leeuwen**
⇒ $O(\log^2 n)$ update,
 $O(\log n + k \log^2 n)$ query by k repeated deletions
or $O(\log n + k \log n)$ query by direct recursion
- **Brodal, Jacob**
⇒ $O(\log n)$ update, $O(\log n + k \log n)$ query
- **Static case: Chazelle, Guibas, Lee '85**
 $O(n)$ space, $O(\log n + k)$ query
- **Agarwal, Matoušek '92:**
 $O(n^\epsilon)$ update, $O(\log n + k)$ query

- **Open:** $O(\text{polylog})$ update, $O(\text{polylog } n + k)$ query??

- **New Result**

$O(\log^{6+\varepsilon} n)$ update (amort., rand.), $O(\log n + k)$ query

Technique

- Based on C. [SODA'06] on dynamic 3D convex hull:
 - Logarithmic Method
 - + Deletion by Re-insertion
 - + Hierarchy of Shallow Cuttings
 - NEW:** + Auxiliary DSs for Conflict Lists
 - (+ bootstrap twice)

Applications

- Dynamic 3D halfspace range reporting:

$O(\log^{6+\varepsilon} n)$ update (amort., rand.),
 $O(\log^2 n / \log \log n + k)$ query

- Dynamic 3D dominance range reporting:

$O(\log^{6+\varepsilon} n)$ update (amort., rand.), $O(\log n + k)$ query

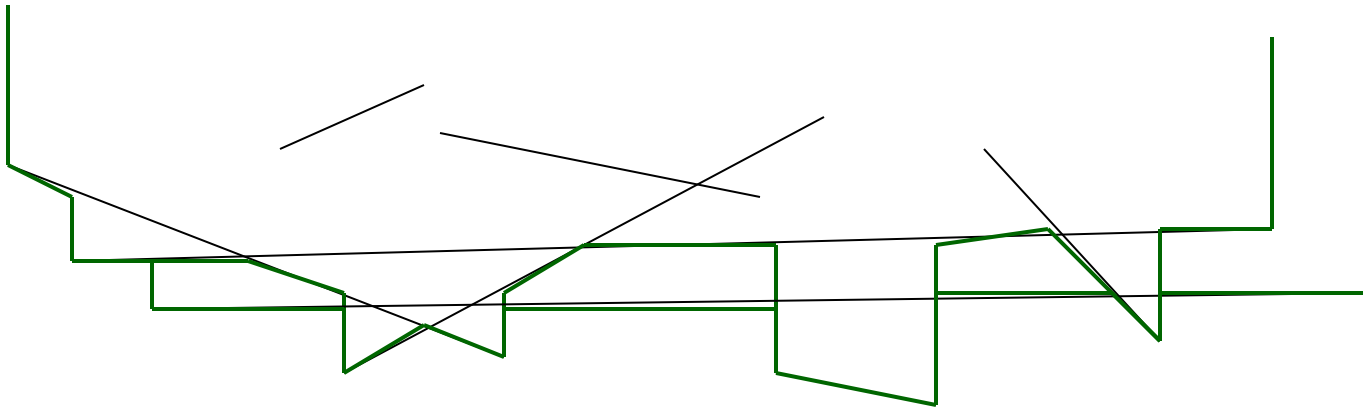
- Dynamic 3D orthogonal range reporting:

$O(\log^{9+\varepsilon} n)$ update (amort., rand.), $O(\log n + k)$ query

III. (Semi-)Dynamic 2D Lower Envelopes of Line Segments

- Maintain set S of line segments in 2D under insertions & deletions s.t.

we can answer queries about the lower envelope (LE)



[generalizes dynamic CH \Leftrightarrow LE of lines]

What's Known

Decomposable search problem!

- $O(\text{polylog})$ update, query is straightforward for "Type-A" queries:
 - decide whether a query pt is below LE
 - intersect LE with vertical line
 - decide whether a query line segment is **completely below** LE (or **partially above** LE)
 - ray shooting from a pt below LE
- But not "Type-B" queries:
 - decide whether a query line segment is **completely above** LE (or **partially below** LE)
 - ray shooting from a pt above LE

NOT decomposable!!

What's Known for Type-B Queries

- $O^*(n^{1/2})$ update, query for insert-only
by a method of C. [SODA'02]
- $O^*(n^{1/2})$ update, query for fully dynamic
by dividing into $n^{1/2}$ slabs [noted by Agarwal]
- Open: better?? (say, for insert-only)
- New Result:

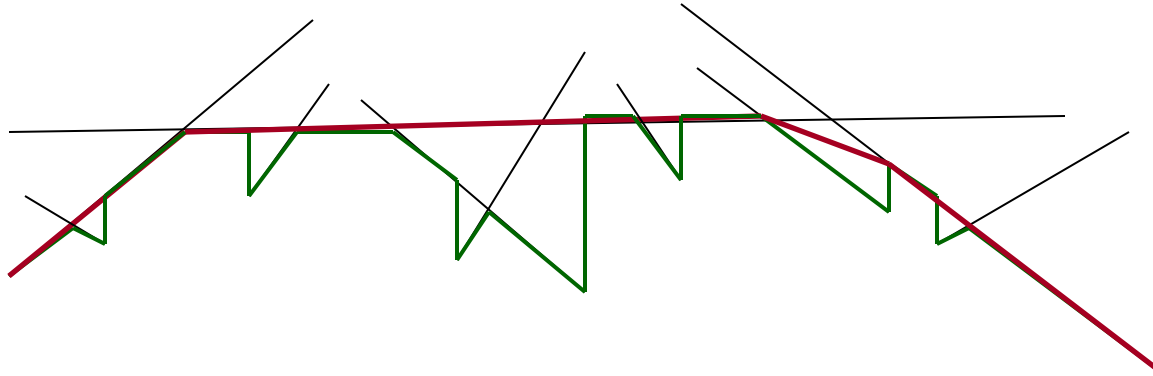
$O(n^\epsilon)$ update (amort.), $O(\log n)$ query

for insert-only (or "semi-online")

[or $O(2^{O(\log n)^{1/2}})$ update, query]

Technique

- Maintain upper hull of lower envelope



- Overmars, van Leeuwen's Hull Tree
+ An Unusual Variant of Segment Tree

An Application

- Given triangulated terrain in 3D & a viewpt, can find all faces that are **partially visible** (or **completely hidden**) in

$O(n^{1+\epsilon})$ time

Open Problems

- **Dynamic CH:**
Type-B queries with $O(\log n)$ update, query??
- **Dynamic halfplane range reporting:**
 $O(\log n + k)$ query with better than $O(\log^6 n)$ update??
- **Dynamic LE of line segments:**
 $O(\text{polylog})$ update, query for insert-only??
fully dynamic??