CPS234 Computational Geometry **08 September 2005**

Lecture 4: Convex Hulls in Higher Dimensions

Lecturer: Pankaj K. Agarwal *Scribe: Amber Stillings*

4.1 Lecture Summary

This lecture will introduce an incremental construction algorithm for convex hulls in $\Re^d$ with constant d, as well as any necessary mathematical definitions and observations. A worst-case time analysis is shown for this algorithm. Randomization is then added, and a time analysis is done to show that randomization improves the expected running time. Additional information and links are provided at the end of these notes.

4.2 Incremental Construction Algorithm for Convex Hulls

4.2.1 Visible/Invisible Facets and Horizon Ridges

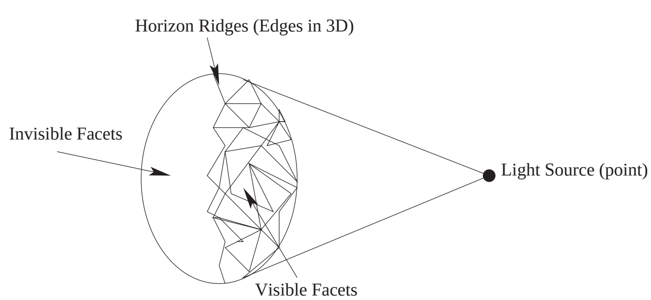

Figure 4.1: Visible Facets, Invisible Facets, and Horizon Ridges

When are facets visible?

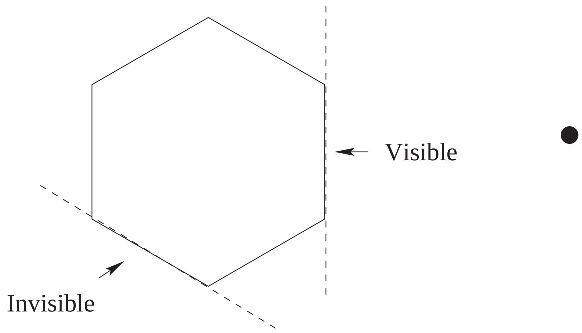

Figure 4.2: Visible and Invisible Facets in 2D

Use Orient Test to determine whether a facet is visible from point q.

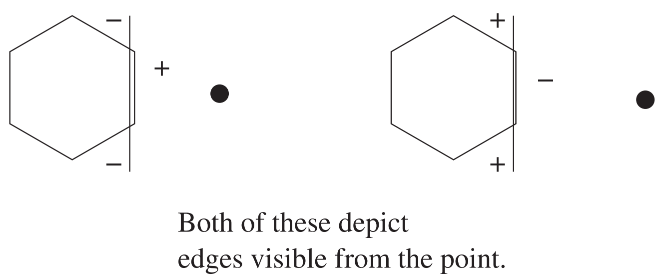

Figure 4.3: Orient Test to Show Visibility

If the signs of the orient test did not alternate as in the picture above, the facet would not be visible from the point.

R: set of points
$q \in \Re^d$

Definition 1 $V(q, R)$*: set of facets of* $conv(R)$ *visible from* q

Definition 2 $hor(q, R)$*: set of horizon ridges, where a horizon edge is an edge of both a visible and invisible facet.*

Observation 1) The set of visible facets are connected. $V(q, R)$ is a connected set.

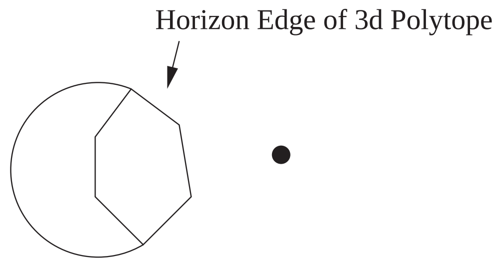

Figure 4.4: Intuition for Connect Visible Facets

The horizon edges form a cycle in the above 3d convex polytope. It follows from this that the visible facets are connected.

Observation 2)

$$conv(R \cup q) = (conv(R) - V(q, R)) \bigcup_{g \in hor(q,R)} conv(q, g)$$

4.2.2 Algorithm

$S_r = p_1, \ldots, p_r$
$P_r = conv(S_r)$
$P_{d+1} = conv(p_1, \ldots, p_{d+1})$ (no points are degenerate)
for $i = d + 2$ to n do
  Compute $V(p_i, S_{i-1})$
  Delete $f \in V(p_i, S_{i-1})$ from P_{i-1}
  Add $conv(q \cup p_i)$ to $P_{i-1} \forall q \in hor(p_i, S_{i-1})$
endfor

Computing Visible Facets (Step 1 of Algorithm)

Compute $V(p_i, S_{i-1})$

Given a facet $f \in V(p_i, S_{i-1})$, perform a Depth-First-Search on a face lattice starting from f and report all facets in $V(p_i, S_{i-1})$ and ridges in $hor(p_i, S_{i-1})$. (Assume for now that you already have a facet $f \in V(p_i, S_{i-1})$ The update function discussed later demonstrates how this would be obtained.) A ridge is a horizon ridge if it belongs to both a visible and invisible facet.

4.3 Worst Case Running Time of Incremental Algorithm

Definition 3 D_i: *number of facets deleted in step* i

Definition 4 *C_i: number of facets created in step i*

Definition 5 *deg(q, R): number of facets incident upon q $\in$ conv(R)*

Running Time:

$$\binom{d+1}{d} + \sum_{i=d+2}^{n} C_i + D_i \leq 2\sum_{i=d+2}^{n} C_i$$

(Ignore $\binom{d+1}{d}$ because it is a constant, and D_i is bounded by C_i).

$$\begin{aligned} 2\sum_{i=d+2}^{n} C_i &= 2\sum_{i=d+2}^{n} deg(p_i, S_{i-1}) \\ &= 2\sum_{i=d+2}^{n} O(i^{\lfloor \frac{d-1}{2} \rfloor}) \\ &\leq n^{\lfloor \frac{d-1}{2} \rfloor} + 1 \\ &= O(n^{\lceil \frac{d}{2} \rceil}) \end{aligned}$$

If you don't have any control of the order you add points, in 3D the algorithm is quadratic in the worst case. (Can you change the running time from ceiling to floor? Yes, with a little smartness.)

4.4 Randomized Incremental Construction (RIC)

Theorem 1 *If $p_1, \cdots, p_n$ is a random permutation, then the expected value*

$$\begin{aligned} E[deg(p_i, S_{i-1})] &= O(i^{\lfloor \frac{d}{2} \rfloor - 1}) \\ \sum_{i \geq d+2}^{n} i^{\lfloor \frac{d}{2} \rfloor - 1} &= O(n^{\lfloor \frac{d}{2} \rfloor}) \;\; for \;\; d \geq 4 \end{aligned}$$

Proof: Focusing on the i^{th} step, already having added $i-1$ points to the convex polytope.

$$E[deg(p_i, S_{i-1})] = \frac{1/i}{\sum_{p \in S_i}} deg(p, S_i)$$

Count points on each facet $\rightarrow$ d points

$$\begin{aligned} &= \frac{d}{i} O(i^{\frac{d}{2}}) \\ &= O(d \cdot i^{\frac{d}{2} - 1}) \\ &= O(i^{\frac{d}{2} - 1}) \end{aligned}$$

4.4.1 Modification to Algorithm

For each point $q \in S \setminus S_i, (q \notin conv(S_i))$, maintain a facet $F = v(q) \in V(q, S_i)$.

For each $F \in conv(S_i)$ maintain backward pointers $v(F) = q \mid F = v(q)$.

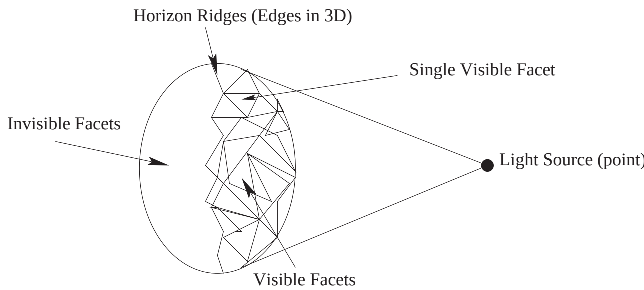

Figure 4.5: Start DFS from a single Visible Facet

Update

$\forall F \in V(p_i, S_{i-1})$
$\forall q \in v(F)$
Perform a DFS among $V(p_i, S_{i-1}) \cap V(q, S_{i-1})$
Collect all horizon ridges, G_q, visible from q.
$\forall g \in G_q$, if $conv(g, p_i)$ is visible from q then $v(q) = conv(g \cup p_i)$

Incremental Construction Algorithm with added update

for $i = d + 2$ to n do
Compute $V(p_i, S_{i-1})$
Delete $f \in V(p_i, S_{i-1})$ from P_{i-1}
Add $conv(q \cup p_i)$ to $P_{i-1} \forall q \in hor(p_i, S_{i-1})$
Update $v(q)$ if $v(q) \in V(p_i, S_{i-1})$
endfor

4.5 Analysis of RIC – Pankaj's Notes

Definition 6 *deg(p,P): number of facets in P incident upon p.*

4.5.1 Running Time

Running Time:

$$\sum_{r=d+2}^{n} \text{deg}(p_r, P_r)$$

Expected Value:

$$E[\sum_{r=d+2}^{n} \text{deg}(p_r, P_r)] = \sum_{r=d+2}^{n} E[\text{deg}(p_r, P_r)]$$

Each facet contains d vertices:

$$\sum_{p \in P_r} \text{deg}(p, P_r) = d \cdot f(P_r) = d \cdot r^{\lfloor \frac{d}{2} \rfloor}$$

So:

$$\begin{aligned} \sum_{r=d+2}^{n} E[\text{deg}(p_r, P_r)] &= \sum_{r=d+2}^{n} \frac{d}{r} r^{\lfloor \frac{d}{2} \rfloor} \\ &= \begin{cases} O(r^{\lfloor \frac{d}{2} \rfloor}) & \text{d} \geq 4 \\ O(r \log n) & \text{d} \leq 3 \end{cases} \end{aligned}$$

4.5.2 Finding a Visible Face

Definition 7 (p, F)*: Visible point-facet pair*

Definition 8 $V(p, R)$*: number of facets of conv(R) visible from* p*.*

Definition 9 $v(p, R) =\mid V(p, R) \mid$

Definition 10 C_r*: number visibility pairs* (q, F) *created in the* r^{th} *step, where* F *is one of the vertices incident to the facet added while inserting the* r^{th} *point.*

$$\begin{aligned} E[C_r]_{givenR} &= \frac{d}{r} \sum_{q \in S \setminus R} v(q, R) \\ E[C_r] &= \frac{1}{\binom{n}{r}} \sum_{R \subset S} \frac{d}{r} \sum_{q \in S \setminus R} v(q, R) \end{aligned}$$

Definition 11 $\Phi\ (R)$*: number of facets in conv*(R)

Definition 12 Φ_r*: expected number of facets in a convex hull of a* r*- subset of* s

$$\Phi_r = \frac{1}{\binom{n}{r}} \sum_{R \subseteq S, |R|=r} \Phi(R)$$

$$v(q,R) = \Phi(R) - \Phi(R \cup q) + \text{deg}(q, R \cup q)$$

So:

$$E[C_r] = \frac{1}{\binom{n}{r}} \sum_{R \subseteq S, |R|=r} \frac{d}{r} \sum_{q \in S \setminus R} (\Phi(R) - \Phi(R \cup q) + deg(q, R \cup q)$$

$$\frac{1}{\binom{n}{r}} \sum_{R \subseteq \binom{S}{r}} \frac{d}{r} \sum_{q \in S-R} \Phi(R) = \frac{d}{r}(n-r)\Phi_r$$

$$\frac{1}{\binom{n}{r}} \sum_{R \subseteq \binom{S}{r}} \frac{d}{r} \sum_{q \in S-R} \Phi(R \cup q) = \frac{d}{r} \cdot \frac{n-r}{r+1} \cdot \frac{1}{\binom{n}{r+1}} \sum_{R \subseteq \binom{S}{r+1}} \sum_{q \in R'} \Phi(R')$$

$$= \frac{d}{r}(n-r)\Phi_{r+1}$$

$$\frac{1}{\binom{n}{r}} \sum_{R \subseteq \binom{S}{r}} \frac{d}{r} \sum_{q \in S-R} deg(q, R \cup q) = \frac{d}{r} \cdot \frac{n-r}{r+1} \cdot \frac{1}{\binom{n}{r+1}} \sum_{R' \subseteq \binom{S}{r+1}} \sum_{q \in R'} deg(q, R')$$

$$= \frac{d}{r} \cdot \frac{n-r}{r+1} \cdot \frac{1}{\binom{n}{r+1}} \sum_{R' \subseteq \binom{S}{r+1}} d * \Phi(R')$$

$$= \frac{d^2}{r} \cdot \frac{n-r}{r+1} \cdot \Phi_{r+1}$$

$$E[C_r] = \frac{d}{r}(n-r)\Phi_r - \frac{d}{r}(n-r)\Phi_{r+1} + \frac{d^2(n-r)}{r(r+1)}\Phi_{r+1}$$

$$\leq d(\frac{n}{r} - 1)\Phi_r - d(\frac{n}{r+1} - 1)\Phi_{r+1} + \frac{d^2(n-r)}{r(r+1)}\Phi_{r+1}$$

$$\sum_{r=d+1}^{n} E[C_r] \leq d(\frac{n}{d+1} - 1)\Phi_{d+1} + \sum_{r=d+1}^{n} \frac{d^2(n-r)}{r(r+1)}\Phi_{r+1}$$

$$\leq d(\frac{n}{d+1} - 1)\Phi_{d+1} + \sum_{r=d+1}^{n} \frac{d^2 n}{r^2} O(r^{\lfloor d/w \rfloor})$$

$$= \begin{cases} O(r^{\lfloor \frac{d}{2} \rfloor}) & \text{d} \geq 4 \\ O(r \log n) & \text{d} \leq 3 \end{cases}$$

4.6 Additional Information/Links

Seidel discusses another construction algorithm for convex hull of higher dimension [1]. This algorithm runs in O($m^2 + F \log m$). algorithm implements a method called ”straight line shelling of a polytope” to achieve this time. This paper also summarizes the ”beneath-beyond method” and gift-wrapping methods for construction.

Clarkson, Mehlhorn, and Seidel discuss RIC and results [2]. The paper discusses space, data structures, and time for insertion/deletion, as well as a game related to RICs.

Bronnimann discusses an algorithm that can handle degenerate points with the same expected running time as RIC[3].

References

[1] Raimund Seidel Constructing higher-dimensional convex hulls at logarithmic cost per face In *Proceedings of the eighteenth annual ACM symposium on Theory of computing*, 1986

[2] Kenneth L Clarkson, Kurt Mehlhorn, and Raimund Seidel, Four results on randomized incremental constructions In *Computational Geometry: Theory and Applications*, September 1993

[3] Herve Bronnimann Degenerate convex hulls on-line in any fixed dimension In *Proceedings of the fourteenth annual symposium on Computational geometry*, 1998