An Optimal-Time Algorithm for Ham-Sandwich Cuts in the Plane (Abstract)

Chi-Yuan Lo
AT&T Bell Laboratories
and
Department of Computer Science
Rutgers University

William Steiger[1,2]
Department of Computer Science
Rutgers University

June 28, 1990

Abstract

Given sets $A = \{a_1, \dots, a_{n_A}\}$ and $B = \{b_1, \dots, b_{n_B}\}$ of points in the plane, $n = n_A + n_B$. A *ham-sandwich cut* is a line h with the property that at most half of the points in A and half of the points in B lie on the same side of h. For the case where the convex hulls of A and B do not intersect, Megiddo gave an algorithm to compute h that runs in time $O(n)$. Edelsbrunner and Waupotitsch modified Megiddo's algorithm for the general case so it can compute h in time $O(n \log n)$. Here we give a linear time algorithm and resolve the question regarding the complexity of computing h in two dimensions.

[1]The author expresses gratitude to the NSF DIMACS Center at Rutgers
[2]Research Supported in Part by NSF grant CCR-8902522

1 Introduction and Summary

Suppose that we are given two sets of points in the plane, $A = \{a_1, \ldots, a_{n_A}\}$ and $B = \{b_1, \ldots, b_{n_B}\}$, $n = n_A + n_B$. A line ℓ *bisects* a set, say A, if neither of the open halfspaces defined by ℓ contain more than $n_A/2$ points of A. A *ham-sandwich cut* is a line h that simultaneously bisects A and B. The ham sandwich theorem (see for example [3]) guarantees the existence of such a cut.

A special case of this problem arises in the question posed by Willard [7]. Given a set $S = \{p_1, \ldots, p_m\}$, he asked for two lines, h and h', which divide the plane into four quadrants, none of which contains more than $m/4$ points of S. It is trivial to obtain a line h' that bisects S (into A, points on the left of h' and $B = S \backslash A$, of points on the right). The other line h is then a ham-sandwich cut. Cole, Sharir and Yap [2] considered this *separated* case of the ham sandwich problem, where the convex hulls of A and B do not intersect. They gave an $O(n(\log n)^2)$ algorithm, and conjectured that $O(n \log n)$ was possible. In fact Megiddo [6] showed how to compute h in the separated case in linear time. Throughout we use a model of computation where any arithmetic operation or comparison is charged unit cost.

Edelsbrunner and Waupotitsch [4] modified Megiddo's algorithm for the general case. Their algorithm can compute h in time $O(n \log n)$. In this paper we give a linear time algorithm for the general case and thus settle the question regarding the complexity of two dimensional ham sandwich cuts.

It is convenient to look at the dual of our problem. We use the transformation T which maps the point (x, y) to the line ℓ whose equation is $v = xu + y$ and the line with equation $y = mx + b$ to the point $(-m, b)$. It is familiar that point-line incidence is preserved under this mapping. Moreover a point $P = (x, y)$, vertically above a line ℓ, maps to a line vertically above $T(\ell)$. In the dual, we have an arrangement of n lines, $\ell_1, \ldots, \ell_n$, n_A of them from set A (call them $a_1^*, \ldots, a_{n_A}^*$) and n_b of them from B (call then $b_1^*, \ldots, b_{n_B}^*$). The dual of a ham sandwich cut h is a point $h^* = T(h)$ which has half of the A lines and half of the B lines above it.

We make a general position assumption, namely that no line is vertical, no two are parallel, and no three intersect in a common point. Thus we have $N = \binom{n}{2}$ intersection points $\ell_i \bigcap \ell_j = (x_{ij}, y_{ij})$. We write $t_1 < \ldots < t_N$ for the x-coordinates of these points, in order. An important idea is the *k-level* in a line arrangement. This is defined as the continuous, piecewise linear function L_k whose segments always coincide with one of the lines in the arrangement. At a given x, L_k is ℓ_q if line q has $k-1$ lines above it at that x. When $j = \lfloor (n+1)/2 \rfloor$, L_j is called the *median-level* of the arrangement.

Suppose that both n_A and n_B are odd. For $x < t_1$ (to the left of all intersections) and for $x > t_N$ (to the right of all intersections) the median level of the A's is the same A line, say a_p^*; similarly the B median level outside $[t_1, t_n]$ is, say b_q^*. Assume that the slope of a_p^* is less than that of b_q^* (the opposite case is similar). Then the A's median level is above that of the B's at the left of the arrangement, and below it at the right. By continuity the median levels must intersect at some point h^*, a ham sandwich cut. A brute force, $O(n^3)$ algorithm would test each intersection $a_i^* \cap b_j^*$ of an A line with a B line to see whether this point is a ham sandwich cut. In the next section we present some of the ideas involved in quickly finding a pair of lines a_r^* and b_s^* whose intersection point is h^*.

2 The Method

We have n lines, n_A of them from A and n_B of them from B. We seek a certain pair a_r^* and b_s^* whose intersection, h^*, is an intersection point of the median levels of the A's and B's. We describe a linear time algorithm for this task. The main idea is a familiar one. In time $O(n)$ we discard a fixed fraction, α, of the lines from further consideration. This leaves $(1-\alpha)n$ lines among which h^* is the intersection point of (say) the p and q levels of the remaining A and B lines (p and q no longer necessarily corrrespond to medians).

A key ingredient is the following result on the complexity of approximating t_k, the k^{th} smallest intersection point (x-coordinate) in an arrangement of n lines. An intersection point t_j is called an *ε-approximation of t_k* if $|j-k| < \varepsilon N$. We have

Lemma 1 *(Matoušek) Given n lines in general position, a rank, k, $1 \le k \le N$, and $\varepsilon > 0$, an ε approximation of the k^{th} intersection point, t_k, may be found in time $O(n \log \varepsilon^{-1})$.*

This statement was first proposed in the important paper of Matoušek [5]. The special case where $\varepsilon = c/n$ was proved in [1] and Lemma 1 is actually implicit from the methods of that paper.

Inductively we are trying to find an intersection of the p level of the A-lines and the q level of the B-lines, knowing in advance that the levels do intersect. Start by considering only the A lines. We will approximate $t_{j_1} < t_{j_2} \ldots$, where

$$j_i = \lfloor i\frac{\varepsilon^2}{2}N_A \rfloor ,$$

$i = 1, \ldots, \lfloor 2/\varepsilon^2 \rfloor$, and ε a constant to be chosen later (N_A denotes the number of pairs of A lines). We do this by computing the sequence $t'_{j_1}, t'_{j_2} \ldots$, where t'_{j_i} is an $\varepsilon^2/4$ approximation to t_{j_i}. By Lemma 1, each approximation may be computed in time $O(n_A \log \varepsilon^{-1})$, and therefore the entire sequence in $O((n_A/\varepsilon^2)\log \varepsilon^{-1})$. By definition of ε–approximations, there is at least one, and at most $\varepsilon^2 N_A$ A-intersections between successive t'_{j_i}; thus $t'_{j_1} < t'_{j_2} \ldots$.

At each vertical line $x = t'_{j_i}$ there are n_A intercepts (intersections of A-lines in the arrangement with the vertical line). We select three intercepts at each vertical line, the

p^{th} largest intercept, the $(p - \varepsilon n_A)^{th}$ largest, and the $(p + \varepsilon n_A)^{th}$ largest. This may be done in time $O(n_A/\varepsilon^2)$ overall.

Consider adjacent vertical lines $x = t'_{j_i}$ and $x = t'_{j_{i+1}}$ and the intercepts we have selected. Connect the two $(p - \varepsilon n_A)^{th}$ intercepts by a straight line ℓ_1 and the two $(p + \varepsilon n_A)^{th}$ intercepts by another straight line, ℓ_2. This gives the trapezoidal region $Trap_i$ (the first and last regions are unbounded). It is easy to prove

Lemma 2 *Each region $Trap_i$ contains the p level of the A-lines.*

Here is a sketch showing that in the bounded case, the p level of the A-lines is below ℓ_1 in $Trap_i$. A similar argument shows it is above ℓ_2. Both arguments can be modified for the unbounded regions. At both $x = t'_{j_i}$ and $x = t'_{j_{i+1}}$ there are exactly $p - \varepsilon n_A - 1$ A-lines above ℓ_1. Between $x = t'_{j_i}$ and $x = t'_{j_{i+1}}$ A-lines can cross ℓ_1 from below (greater slope) or from above (smaller slope). Call these collections U and D, respectively. When a line in U crosses ℓ_1, one additional A-line is above ℓ_1; when a line in D crosses, one fewer A-line is above. Therefore $|U| = |D|$. Each line in U intersects each line in D between t'_{j_i} and $t'_{j_{i+1}}$. This gives

$$|U||D| = |U|^2 \leq \varepsilon^2 N_A,$$

the right hand side being a bound on the total number of intersections of A-lines in $(t'_{j_i}, t'_{j_{i+1}})$. Therefore $|U| \leq \varepsilon n_A$ and the p level of the A's remains below ℓ_1.

This argument also shows that at most $|U| + |D| \leq 2\varepsilon n_A$ A-lines meet ℓ_1. The same is true for ℓ_2. Clearly no more than $2\varepsilon n_A$ A-lines can meet either of the vertical segments of $Trap_i$. This proves

Lemma 3 *At most $4\varepsilon n_A$ A-lines meet any region $Trap_i$.*

From now on $\varepsilon = 1/8$. One of the regions must contain the intersection of the p level of the A lines with the q level of the B lines. The first (left most) such region may be found in time $O(n/\varepsilon^2)$ by selecting the intercepts of these levels at successive vertical lines $x = t'_{j_i}$ and noticing whether or not they reverse their relative order. When the appropriate trapezoid has been found the (at least) $n_A/2$ A-lines which avoid this region may be discarded. Assuming r of them were above $Trap_i$, the value of p is decreased by r for the next iteration. The next iteration is similar except that it processes the remaining B lines, and we continue in this way, alternating between the A and B sets and eliminating at least half of the remaining lines in that set. Finally, these ideas combine to prove

Proposition 1 *A planar ham sandwich cut may be found in linear time.*

The ham sandwich theorem guarantees the existence of a $d-1$ dimensional hyperplane in R^d that simultaneously bisects d sets of points, n points in all. When $d = 3$ we can generalize Megiddo's result and find a cut in the separated case in quadratic time. It would be consistent with the present result if $O(n^{d-1})$ turned out to be the complexity in the general case. We can only establish this bound up to logarithmic factors.

References

[1] R. Cole, J.S. Salowe, W.L. Steiger and E. Szemerédi. An Optimal-time Algorithm for Slope Selection. *SIAM J. Comp.* 1989.

[2] R. Cole, M. Sharir, and C. Yap. On k- Hulls and Related Topics *SIAM J. Comp. 16*, 61-77, 1987.

[3] H. Edelsbrunner. *Algorithms in Combinatorial Geometry*. Springer-Verlag, Berlin, 1987.

[4] H. Edelsbrunner and R. Waupotitisch. Computing a Ham Sandwich Cut in Two Dimensions. *J. Symbolic Comp. 2*, 171–178, 1986.

[5] J. Matoušek Construction of ϵ Nets. *Proc.Fifth Ann. ACM Symp. on Comp. Geom.*, 1-10, 1989.

[6] N. Megiddo. Partitioning with Two Lines in the Plane. *J. Algorithms 6*, 430–433, 1985.

[7] D. Willard Polygon Retrieval. *SIAM J. Comp. 11*, 149–165, 1982.