A modification of Graham's algorithm for determining the convex hull of a finite planar set

P. T. An
Institute of Mathematics, Vietnam

Annales Mathematicae et Informaticae
34 (2007) pp. 3–8
http://www.ektf.hu/tanszek/matematika/ami

A modification of Graham's algorithm for determining the convex hull of a finite planar set

Phan Thanh An

Institute of Mathematics
Vietnamese Academy of Science and Technology
e-mail: thanhan@math.ac.vn

Submitted 4 September 2007; Accepted 11 October 2007

Abstract

In this paper, in our modification of Graham scan for determining the convex hull of a finite planar set, we show a restricted area of the examination of points and its advantage. The actual run times of our scan and Graham scan on the set of random points shows that our modified algorithm runs significantly faster than Graham's one.

Keywords: Algorithm, computational complexity, convex hull, extreme point, Graham scan

MSC: 52B55, 52C45, 65D18

1. Introduction

The determination of the convex hull of a point set has successfully been applied in application domains such as pattern recognition [2], data mining [3], stock cutting and allocation [4], or image processing [10].

Graham's algorithm [5] is an important sequential algorithm used for determining the convex hull of the set of n points in the plane ($n \geqslant 3$). This algorithm has a complexity of $O(n \log n)$. Take an interior point x of the convex hull and assume without loss of generality that no three points of the given set (including x) are collinear. We will use the phrase "convex hull" to mean "the set of extreme points of the convex hull". The first step of Graham's algorithm is to construct a sequence $\mathcal{P} = \{p_1, \dots, p_n\}$ of the points in polar coordinates ordered about x in terms of increasing angle (see Fig. 1) (note that point p_1 is adjacent to p_n). In this sequence, call a point *reflex* if the interior angle made by it and its adjacent points is greater than π. In Fig. 1, p_1 is nonreflex and p_2 is reflex. Then, a reflex point does not belong to the convex hull. Graham scan in the algorithm examines the points of the sequence in counterclockwise order and deletes those that are reflex; upon termination, only nonreflex points remain, so the rest is the convex hull of $\mathcal{P}$.

Several modifications of Graham's algorithm have been proposed, all having to do with the following. If the first point in $\mathcal{P}$ is guaranteed to be on the convex hull, then it is never reflex (see [1, 6, 9, 10, 11] etc).

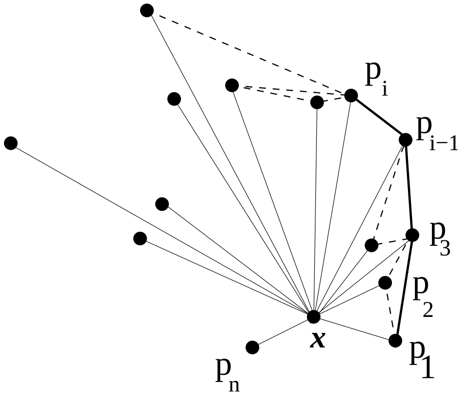

Figure 1: The first step of Graham's algorithm constructs a sequence $\mathcal{P} = \{p_1, \dots, p_n\}$ of the points in polar coordinates ordered about x.

Determining when the counterclockwise examination of points can stop seems to be the major difficulty, because deleting a reflax point can change its neighbors from nonreflex to reflex. That is one of the reasons why some of modifications of Graham's algorithm contain errors (see [7]). In this note, in our modification of Graham scan, we show a restricted area of the examination of points and its advantage. The actual run times of our scan and Graham scan on the set of random points are given in Table 1, which shows that our modified algorithm runs significantly faster than Graham's one.

2. A modification of the Graham scan

We shall shortly describe a restricted area of the examination of points in Graham scan. Suppose that α is some compact convex set containing $\mathcal{P}$ (see Fig. 2). The first step of Graham's algorithm constructs a sequence $\mathcal{P} = \{p_1, \dots, p_n\}$ of the points in polar coordinates ordered about the interior point x in terms of increasing angle. After that, let p_{i-1} be nonreflex (i.e., the interior angle made by it and p_i and p_{i-2} is less than π). Let the rays xp_i and $p_{i-1}p_i$ intersect the boundary of α at u_i and v_i, respectively (see Fig. 2). Denote $\widehat{u_i x v_i}$ and $[\widehat{u_i x v_i}]$ the angle at point x and the area, respectively, formed by rays xu_i and xv_i.

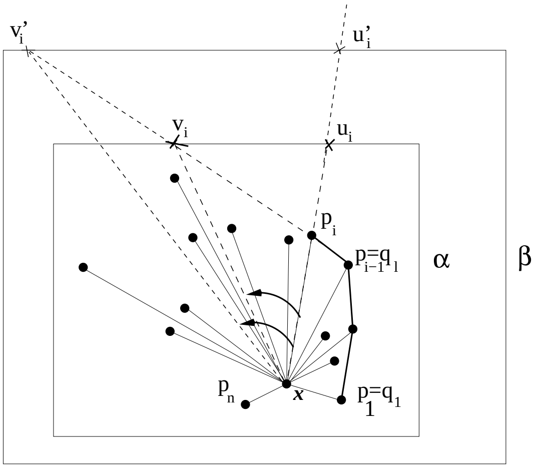

Figure 2: α contains $\mathcal{P}$ and the restricted area at point p_i is $[\widehat{u_ixv_i}]$.
If $\alpha \subset \beta$ then $[\widehat{u_ixv_i}] \subset [\widehat{u_i'xv_i'}]$.

Proposition 2.1. *Let the rays xp_i and $p_{i-1}p_i$ intersect the boundary of α at u_i and v_i, respectively. If p_{i-1} is nonrelfex and all points of $\mathcal{P} \cap [\widehat{u_ixv_i}]$ are nonreflex, then p_i is nonrelfex, too.*

Proof. Assume that $p_{i+1}, \ldots, p_k \in [\widehat{u_ixv_i}]$ and $p_j \notin [\widehat{u_ixv_i}]$ for $k+1 \leqslant j \leqslant n$. Since α is convex, the intersection of α and the closed half-plane bounded by the line $p_{i-1}p_i$ and containing x is convex. It follows that $\widehat{p_{i-1}p_ip_j} < \pi$ for $k+1 \leqslant j \leqslant n$. Therefore, $\widehat{p_{i-1}p_ip_j} < \pi$ for $i+1 \leqslant j \leqslant n$. Since p_{i-1} is nonreflex, p_i is nonreflex, too. □

By Proposition 2.1, to examine if p_i is nonreflex or not, we only need to examine if p_i is nonreflex or not with the points of $\mathcal{P}$ in counterclockwise order beginning from p_{i+1} and belonging to $[\widehat{u_ixv_i}]$. We now present our modification for Graham's algorithm.

Algorithm:

First, find interior point x; label it p_0. Then sort all other points angularly about x; label $p_1, \ldots, p_n$. Set $\mathcal{P} = \{p_1, \ldots, p_n\}$. Take a compact convex set α containing these points. We now determine the convex hull $\mathcal{Q} = \{q_1, \ldots, q_{l+1}\}$.

1. Begin at p_1. Set $l = 1$ and $i = 2$. Because p_1 is on the convex hull, we have $q_1 = p_1$.

2. Consider q_l. If $i = n$, go to **3**. Else, let the rays xq_l and q_lp_i intersect the boundary of α at u_i and v_i, respectively.

2.1 Set $m = 1$.

2.2 If $\widehat{p_i x p_{i+m}} \leqslant \widehat{p_i x v_i}$ (i.e., $p_{i+m} \in \mathcal{P} \cap [\widehat{u_i x v_i}]$) and $\widehat{q_l p_i p_{i+m}} < \pi$, then set $m = m + 1$ and go to **2.2**. Else either

$\widehat{p_i x p_{i+m}} > \widehat{p_i x v_i}$, then by Proposition 2.1, p_i is nonreflex, set $q_{l+1} = p_i$, $i = i + 1$ and $l = l + 1$ go to **2**, or

$\widehat{q_l p_i p_{i+m}} > \pi$, then $\widehat{q_l p_i p_k} < \pi$ for all $p_k \in \mathcal{P}, i < k < i + m$. Set $i = i + m$, go to **2**.

3. Set $q_{l+1} = p_n$. Then, $\mathcal{Q} = \{q_1, \ldots, q_{l+1}\}$ is the convex hull. STOP.

Note that x can be chosen to be a point on the convex hull (see [1, 9]).

Proposition 2.2. *The algorithm computes the convex hull in $n(\log n)$ time.*

Proof. By Proposition 2.1, points of $\mathcal{Q}$ are nonreflex. Hence, the algorithm computes the convex hull.

After sorting points that requires $n(\log n)$ time, the algorithm can only take linear time, since it only advances, never backs up, and the number of steps is therefore limited by the number of points of $\mathcal{P}$. Therefore, the algorithm runs in $n(\log n)$ time. □

Proposition 2.3. *Suppose that α and β are compact convex sets containing $\mathcal{P}$. Let the rays xp_i and $p_{i-1}p_i$ intersect the boundary of α (β, respectively) at u_i and v_i (at u'_i and v'_i, respectively). If $\alpha \subset \beta$ then $[\widehat{u_i x v_i}] \subset [\widehat{u'_i x v'_i}]$.*

Proof. Since α, β are convex and $\alpha \subset \beta$, v_i belongs to the segment $[v'_i, p_i]$. It follows that $[\widehat{u_i x v_i}] \subset [\widehat{u'_i x v'_i}]$. □

Our modification only need to examine the points of $\mathcal{P}$ in counterclockwise order beginning from p_i and belonging to $[\widehat{u_i x v_i}]$ while Jarvis's algorithm [8] and variations of Graham's convex hull algorithm like Akl-Toussaint's algorithm [1], Graham-Yao's algorithm [6], Toussaint-Avis's algorithm [11], etc require that for many points. By Proposition 2.3, the execution time is reduced if the set α is enough small such that it still contains $\mathcal{P}$. So we can choose α to be the smallest rectangle $\mathcal{U}$ enclosing $\mathcal{P}$ and having sides parallel to the coordinate lines.

The algorithm requires to check the condition $\widehat{p_i x p_{i+m}} \leqslant \widehat{p_i x v_i}$. This is implemented in our code as follows: Let xp_{i+m} intersect $u_i v_i$ at $\bar{p}_{i+m}$. Then $\widehat{p_i x p_{i+m}} \leqslant \widehat{p_i x v_i}$ iff x-coordinate of $\bar{p}_{i+m}$ is between x-coordinates of u_i and v_i.

For a given set $\mathcal{P}$ of points randomly positioned in some rectangle $\mathcal{V}$ having sides parallel to the coordinate lines, we can take this rectangle to be α. Based on the "throw-away" principle [1], we can assume that $\mathcal{P}$ includes a finite number of points randomly positioned in the interior of the right-angled triangle abc having sides parallel to the coordinate lines and two points b and c (which form the hypotenuse of the triangle).

Our modified algorithm is implemented in C code. To compare it with Graham's algorithm we use an implementation of Graham's algorithm written by O'Rourke [9]. Codes are compiled by the GNU C Compiler under SuSe Linux 10.0 and are executed on a Pentium IV processor. For the comparison to be meaningful, both implementations use the same code for file reading and rotary sort. The actual run times of the scans in our algorithm and Graham's algorithm on such set $\mathcal{P}$ are given in Table 1, which shows that our modified algorithm runs significantly faster than Graham's one (with integer coordinates). In this case, $\alpha = \mathcal{U} = \mathcal{V}$.

Input size	Number of extreme points	Graham Scan	Our Modified Scan
20000	159	0.0905	0.0638
30000	189	0.1500	0.0946
60000	225	0.3190	0.2068
100000	236	0.5520	0.3611
200000	272	1.2446	0.8503
300000	302	2.0292	1.4025
1000000	376	8.2442	5.7995

Table 1: The actual run times of scans in our algorithm and Graham's algorithm (time in sec) on a finite number of points randomly positioned in the interior of the right-angled triangle *abc* of size 40000 having sides parallel to the coordinate lines and two points *b* and *c*.

References

[1] Akl, S.G. and Toussaint, G.T., A fast convex hull algorithm, *Information Processing Letters*, 7 (1978) 219–222.

[2] Akl, S.G. and Toussaint, G.T., Efficient convex hull algorithms for pattern recognition applications, *Int. Joint Conf. on Pattern Recognition*, Kyoto, Japan, (1978) 483–487.

[3] Böhm, C. and Kriegel, H., Determing the convex hull in large multidimensional databases, *Proceedings of the Third International Conference on Data Warehousing and Knowledge Discovery, Lecture Notes in Computer Science, Springer-Verlag,* 2114 (2001) 294–306.

[4] Freeman, H. and Shapira, R., Determining the minimum-area encasing rectangle for an arbitrary closed curve, *Comm. ACM*, 18(7) (1975).

[5] Graham, R.L., An efficient algorithm for determining the convex hull of a finite planar set, *Information Processing Letters*, 26 (1972) 132–133.

[6] Graham, R.L. and Yao, F.F., Finding the convex hull of a simple polygon, *Journal of Algorithms*, 4 (1983) 324–331.

[7] Gries, D. and Stojmenovic', I., A note on Graham's convex hull algorithm, *Information Processing Letters*, 25 (1987) 323–327.

[8] Jarvis, R.A., On the identication of the convex hull of a finite set of points in the plane, *Information Processing Letters*, 2 (1973) 18–21.

[9] O'Rourke, J., Computational Geometry in C, *Cambridge University Press, Second Edition*, 1998.

[10] Rosenfeld, A., Picture Processing by Computers, *Academic Press, New York*, 1969.

[11] Toussaint, G.T. and Avis, D., On convex hull algorithm for polygons and its application to triangulation problems, *Pattern Recognition*, 15, No. 1 (1982) 23–29.

Phan Thanh An
Institute of Mathematics
Vietnamese Academy of Science and Technology
18 Hoang Quoc Viet Road, 10307 Hanoi, Vietnam