AN EFFICIENT ALGORITHM FOR DETERMINING THE CONVEX HULL OF A FINITE PLANAR SET

R.L. GRAHAM

Bell Telephone Laboratories, Incorporated
Murray Hill, New Jersey, USA

Received 28 January 1972

Given a finite set \( S = \{ r_1, \ldots, r_n \} \) in the plane, it is frequently of interest to find the convex hull \( \text{CH}(S) \) of \( S \). In this note we describe an algorithm which determines \( \text{CH}(S) \) in no more than \( (n \log n)/(\log 2) + \text{cn} \) "operations" where \( c \) is a small positive constant which depends upon what is meant by an "operation".

The algorithm we give determines which points of \( S \) are the extreme points of \( \text{CH}(S) \). These, of course, define \( \text{CH}(S) \). The algorithm proceeds in five steps.

Step 1: Find a point \( P \) in the plane which is in the interior of \( \text{CH}(S) \). At worst, this can be done in \( c_1 n \) steps by testing 3 element subsets of \( S \) for collinearity, discarding middle points of collinear sets and stopping when the first noncollinear set (if there is one), say \( x, y, z \), is found. \( P \) can be chosen to be the centroid of the triangle formed by \( x, y, z \).

Step 2: Express each \( r_k \) in polar coordinates with origin \( P \) and \( \theta = 0 \) in the direction of an arbitrary fixed half-line \( L \) from \( P \). This conversion can be done in \( c_2 n \) operations for some fixed constant \( c_2 \).

Step 3: Order the elements \( r_k \exp(i\theta_k) \) of \( S \) in terms of increasing \( \theta_k \). This is well known to be possible in essentially \( (n \log n)/2 \) comparisons (cf. \( [1] \)). We now have \( S \) in the form

\[ S = \{ r_1 \exp(i\varphi_1), \ldots, r_n \exp(i\varphi_n) \} \]

with \( 0 < \varphi_1 < \ldots < \varphi_n < 2\pi \) and \( r_j > 0 \) (cf. fig. 1). Note that by the choice of \( P \), \( \varphi_{k-1} - \varphi_k < \pi \) where the index addition is modulo \( n \).

Step 4: If \( \varphi_j = \varphi_{j+1} \) then we may delete the point with the smaller amplitude since it clearly cannot be an extreme point of \( \text{CH}(S) \). Also any point with \( r_j = 0 \) can be deleted. We can eliminate all these points in less than \( n \) comparisons, and by relabelling the remaining points, we can set

\[ S' = \{ r_1 \exp(i\varphi_1), \ldots, r_n' \exp(i\varphi_n') \} \]

where \( n' < n \).

Step 5: Start with three consecutive points in \( S' \), say, \( r_k \exp(i\varphi_k), r_{k+1} \exp(i\varphi_{k+1}), r_{k+2} \exp(i\varphi_{k+2}) \) with \( \varphi_k < \varphi_{k+1} < \varphi_{k+2} \) (cf. fig. 2). There are two possibilities:

(i) \( \alpha + \beta = \pi \). Then we delete the point \( r_{k+1} \exp(i\varphi_{k+1}) \) from \( S' \) since it cannot be an extreme point of \( \text{CH}(S) \), and return to the beginning of step 5 with the points \( r_k \exp(i\varphi_k), r_{k+1} \exp(i\varphi_{k+1}), r_{k+2} \exp(i\varphi_{k+2}) \) replaced by \( r_{k-1} \exp(i\varphi_{k-1}) \), \( r_k \exp(i\varphi_k), r_{k+2} \exp(i\varphi_{k+2}) \) (where indices are reduced modulo \( n \)).
(ii) $\alpha + \beta < \pi$. Return to the beginning of step 5 with the points $r_k \exp(i\varphi_k), r_{k+1} \exp(i\varphi_{k+1}), r_{k+2} \exp(i\varphi_{k+2})$ replaced by $r_{k+1} \exp(i\varphi_{k+1}), r_{k+2} \exp(i\varphi_{k+2}), r_{k+3} \exp(i\varphi_{k+3})$.

By noting that each application of step 5 either reduces the number of possible points of $CH(S)$ by one or increases the current total number of points of $S'$ considered by one, an easy induction argument shows that with less than $2n'$ iterations of step 5, we must be left with exactly the subset of $S$ of all extreme points of $CH(S)$. This completes the algorithm.

The reader may find it instructive to consider a small example of ten points or so. Computer implementation of this algorithm makes it quite feasible to consider examples with $n = 50,000$.

---

**Reference**