

A Characterization Theorem and an Algorithm for a Convex Hull Problem

Bahman Kalantari

Extended Abstract. Given a set $S = \{v_1, \dots, v_n\} \subset \mathbb{R}^m$ and a point $p \in \mathbb{R}^m$, testing if $p \in \text{conv}(S)$, the convex hull of S , is a fundamental problem in computational geometry and linear programming. Denoting the Euclidean distance between $u, w \in \mathbb{R}^m$ by $d(u, w) = \sqrt{\sum_{i=1}^m (u_i - w_i)^2}$, first we prove a *distance duality*:

Distance Duality

Precisely one of the two conditions is satisfied:

- (i): For each $p' \in \text{conv}(S) \setminus \{p\}$, there exists $v_j \in S$ such that $d(p', v_j) \geq d(p, v_j)$;
- (ii): There exists $p' \in \text{conv}(S)$ such that $d(p', v_i) < d(p, v_i)$, for all $i = 1, \dots, n$.

Condition (i) is valid if and only if $p \in \text{conv}(S)$, and condition (ii) if and only if $p \notin \text{conv}(S)$. Utilizing this duality, we describe a simple fully polynomial time approximation scheme, called the *Triangle Algorithm*:

Triangle Algorithm ($S = \{v_1, \dots, v_n\}, p$)

- **Step 1.** Given $p' \in \text{conv}(S) \setminus \{p\}$, check if there exists $v_j \in S$ such that $d(p', v_j) \geq d(p, v_j)$. If no such v_j exists, stop, $p \notin \text{conv}(S)$.
- **Step 2.** Otherwise, on the line segment joining p' to v_j compute the point nearest to p . Denote this by p'' . Replace p' with p'' , go to Step 1.

We refer to p' in Step 1 as *iterate* and v_j as *pivot point*. Given $\epsilon \in (0, 1)$, the Triangle Algorithm in at most $48mn\epsilon^{-2} = O(mn\epsilon^{-2})$ arithmetic operations computes a point $p' \in \text{conv}(S)$ such that either

$$d(p', p) \leq \epsilon d(p, v_j), \quad \text{for some } j; \text{ or} \quad (1)$$

$$d(p', v_i) < d(p, v_i), \quad \forall i = 1, \dots, n. \quad (2)$$

We refer to the point p' satisfying (1) as an ϵ -*approximate solution*. Clearly, approximation to a prescribed absolute error is also possible. We refer to a point p' satisfying (2) as *witness*. This condition holds if and only if $p \notin \text{conv}(S)$. This is because in this case we can prove the Voronoi cell of p' with respect to the two point set $\{p, p'\}$ contains $\text{conv}(S)$ (see Figure 1). Equivalently, the orthogonal bisector of the line segment pp' separates p from $\text{conv}(S)$.

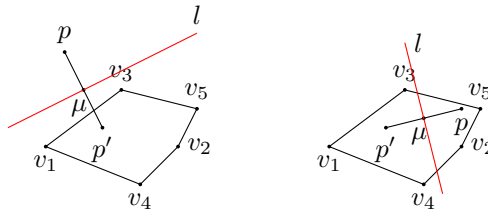


Figure 1: Example of cases where orthogonal bisector of pp' does and does not separate p from $\text{conv}(S)$.

The set W_p of all such witnesses is the intersection of $\text{conv}(S)$ and the open balls, $B_i = \{x \in \mathbb{R}^m : d(x, v_i) < d(p, v_i)\}$, $i = 1, \dots, n$. W_p is a convex open set in the relative interior of $\text{conv}(S)$ (see Figure 2).

By squaring the distances, $d(p', v_j) \geq d(p, v_j) \iff d(p', 0)^2 - d(p, 0)^2 \geq 2v_j^T(p' - p)$. Thus Step 1 does not require taking square-roots. Also, the computation of p'' in Step 2 requires no square-root operations.

Given a point $p' \in \text{conv}(S)$ that is not a witness, having $d(p, p')$ as the current *gap*, the Triangle Algorithm moves to a new point $p'' \in \text{conv}(S)$ where the new gap $d(p, p'')$ is reduced. We will prove that when $p \in \text{conv}(S)$, the number of iterations K_ϵ , needed to get an approximate solution p' satisfying (1) is bounded above by $48\epsilon^{-2} = O(\epsilon^{-2})$. In the worst-case each iteration of Step 1 requires $O(mn)$ arithmetic operations.

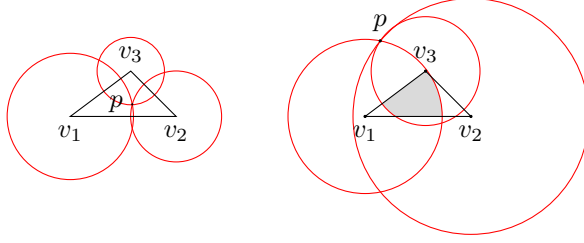


Figure 2: Examples of empty W_p ($p \in \text{conv}(S)$) and nonempty W_p ($p \notin \text{conv}(S)$), gray area.

However, it may also take only $O(m)$ operations. The number of arithmetic operations in each iteration of Step 2 is only $O(m)$. Thus the complexity for computing an ϵ -approximate solution is $O(mn\epsilon^{-2})$ arithmetic operation. In particular, for fixed ϵ the complexity of the algorithm is only $O(mn)$.

When $p \notin \text{conv}(S)$, the Triangle Algorithm does not attempt to compute the closest point to p , say $p_* \in \text{conv}(S)$, rather a separating hyperplane. However, by virtue of the fact that it finds a hyperplane orthogonally bisecting the line pp' , it in the process computes an approximation to $d(p, p_*)$ to within a factor of two. More precisely, any witness p' satisfies the inequality

$$.5d(p, p') \leq d(p, p_*) \leq d(p, p'). \quad (3)$$

Not only this approximation is useful for the convex hull problem, but for computing the distance between two convex hulls, the *polytope distance* problem. As is well known the Minkowski difference of two convex hulls is a polytope whose shortest vector has norm equal to the distance between the two polytopes.

The justification in the name of the algorithm lies in the fact that in each iteration the algorithm searches for a triangle $\triangle pp'v_j$ where $v_j \in S$, $p' \in \text{conv}(S) \setminus \{p\}$, such that $d(p', v_j) \geq d(p, v_j)$. Given that such triangle exists, it uses v_j as a pivot point to “pull” the current iterate p' closer to p to get a new iterate $p'' \in \text{conv}(S)$.

We also show how to solve general LP via the Triangle Algorithm and give a corresponding complexity analysis. In particular, we prove a sensitivity theorem that converts LP feasibility with bounded domain into a convex hull problem, then gives the necessary accuracy for computing an ϵ -approximate solution. We also contrast the theoretical performance of the Triangle Algorithm with the *sparse greedy approximation* (equivalent to Frank-Wolfe and Gilbert algorithms) for the minimization of a convex quadratic over a simplex, a problem arising in machine learning, approximation theory, and statistics. The bibliography contains sample references from the main article.

References

- [1] K. L. Clarkson. Coresets, Sparse Greedy Approximation, and the Frank-Wolfe algorithm. In SODA '08: Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms, 922 - 931. Society for Industrial and Applied Mathematics, 2008.
- [2] M. Frank and P. Wolfe, An algorithm for quadratic programming, *Naval Res. Logist. Quart.*, 3 (1956), 95 - 110.
- [3] B. Gärtner and M. Jaggi, Coresets for polytope distance, Symposium on Computational Geometry (2009), 33 - 42.
- [4] E. G. Gilbert, An iterative procedure for computing the minimum of a quadratic form on a convex set, *SIAM Journal on Control* Volume 4 (1966), 61 - 80.
- [5] B. Kalantari, Finding a lost treasure in convex hull of points from known distances. In the Proceedings of the 24th Canadian Journal of Computational Geometry (2012), 271 - 276.
- [6] T. Zhang, Sequential greedy approximation for certain convex optimization problems, *IEEE Trans. Information Theory*, 49 (2003), 682 - 691.

Department of Computer Science, Rutgers University, Piscataway, NJ 08854
kalantari@cs.rutgers.edu