Cutting a Convex Polyhedron Out of a Sphere
(Extended Abstract)

Syed Ishtiaque Ahmed, Masud Hasan, and Md. Ariful Islam

Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology
Dhaka-1000, Bangladesh
ishtiaque@csebuet.org, masudhasan@cse.buet.ac.bd, arifulislam@csebuet.org
http://www.buet.ac.bd/cse

Abstract. Given a convex polyhedron P of n vertices inside a sphere Q, we give an $O(n^3)$-time algorithm that cuts P out of Q by using guillotine cuts and has cutting cost $O(\log^2 n)$ times the optimal.

Keywords: Approximation algorithm, guillotine cut, polyhedra cutting.

1 Introduction

The problem of cutting a convex polygon P out of a piece of planar material Q (P is already drawn on Q) with minimum total cutting length is a well studied problem in computational geometry. The problem was first introduced by Overmars and Welzl in 1985 [12] but has been extensively studied in the last eight years [1–4, 7, 8, 10, 12–14] with several variations, such as P and Q are convex or non-convex polygons, Q is a circle, and the cuts are line cuts or ray cuts.

This type of cutting problems have many industrial applications such as in metal sheet cutting, paper cutting, furniture manufacturing, ceramic industries, fabrication, ornaments, and leather industries. Some of their variations also fall under *stock cutting problems* [3].

If Q is another convex polygon with m edges, this problem with line cuts has been approached in various ways [2–5, 8, 9, 12, 13]. If the cuts are allowed only along the edges of P, Overmars and Welzl [12] proposed an $O(n^3 + m)$-time algorithm for this problem with optimal cutting length, where n is the number of edges of P. The problem is more difficult if the cuts are more general, i.e., they are not restricted to touch only the edges of P. In that case Bhadury and Chandrasekaran showed that the problem has optimal solutions that lie in the algebraic extension of the input data field [3] and due to this algebraic nature of this problem, an approximation scheme is the best that one can achieve [3]. They also gave an approximation scheme with pseudo-polynomial running time [3].

After the indication of Bhadury and Chandrasekaran [3] to the hardness of the problem, many have given polynomial time approximation algorithms. Dumitrescu proposed an $O(\log n)$-approximation algorithm with $O(mn + n \log n)$ running time [8, 9]. Then Daescu and Luo [5] gave the first constant factor approximation algorithm with ratio $2.5 + ||Q||/||P||$, where $||P||$ and $||Q||$ are the

Md.S. Rahman and S. Fujita (Eds.): WALCOM 2010, LNCS 5942, pp. 94–101, 2010.

perimeters of P and the minimum area bounding rectangle of P respectively. Their algorithm has a running time of $O(n^3 + (n + m) \log (n + m))$. The best known constant factor approximation algorithm is due to Tan [13] with an approximation ratio of 7.9 and running time of $O(n^3 + m)$. In the same paper [13], the author also proposed an $O(\log n)$-approximation algorithm with improved running time of $O(n + m)$. As the best known result so far, very recently, Bereg, Daescu and Jiang [2] gave a polynomial time approximation scheme (PTAS) for this problem with running time $O(m + \frac{n^6}{\epsilon^{12}})$.

For ray cuts, Demaine, Demaine and Kaplan [7] gave a linear time algorithm to decide whether a given polygon P is *ray-cuttable* or not. For optimally cutting P out of Q by ray cuts, if Q is convex and P is non-convex but ray-cuttable, then Daescu and Luo [5] gave an almost linear time $O(\log^2 n)$-approximation algorithm. If P is convex, then they gave a linear time 18-approximation algorithm. Tan [13] improved the approximation ratio for both cases as $O(\log n)$ and 6, respectively, but with much higher running time of $O(n^3 + m)$. See Table 1 for a summary of these results.

Our results. The generalization of this problem in 3D is very little known. To the best of our knowledge, the only result is to decide whether a polyhedral object can be cut out form a larger block using continuous hot wire cuts [10]. In this paper we attempt to generalize the problem in 3D. We consider the problem of cutting a convex polyhedron P which is fixed inside a sphere Q by using only guillotine cuts with minimum total cutting cost. A *guillotine cut*, or simply a *cut*, is a plane that does not pass through P and partitions Q into two smaller convex pieces. After a cut is applied, Q is updated to the piece that contains P. The *cutting cost* of a guillotine cut is the area of the newly created face of Q. We give an $O(n^3)$-time algorithm that cuts P out of Q by using only guillotine cuts and has cutting cost no more than $O(\log^2 n)$ times the optimal cutting cost. Also see Table 1.

Table 1. Comparison of the results

Dim.	Cut Type	Q	P	Approx. Ratio	Running Time	Reference
2D	Ray	-	Non-convex	Ray-cuttable?	$O(n)$	[7]
		Convex	Convex	18	$O(n)$	[5]
		Convex	Non-convex	$O(\log^2 n)$	$O(n)$	[5]
		Convex	Convex	6	$O(n^3 + m)$	[13]
		Convex	Non-convex	$O(\log n)$	$O(n^3 + m)$	[13]
	Line	Convex	Convex	$O(\log n)$	$O(mn + n \log n)$	[8, 9]
		Convex	Convex	$2.5 + \|\|Q\|\|/\|\|P\|\|$	$O(n^3 + (n + m) \log (n + m))$	[5]
		Convex	Convex	7.9	$O(n^3 + m)$	[13]
		Convex	Convex	$(1 + \epsilon)$	$O(m + \frac{n^6}{\epsilon^{12}})$	[2]
		Circle	Cornered con.	$O(\log n)$	$O(n)$	[1]
		Circle	Cornered con.	6.48	$O(n^3)$	[1]
3D	Hot-wire	-	Non-convex	Cuttable?	$O(n^5)$	[10]
	Guillotine	**Sphere**	**Convex**	$\boldsymbol{O(\log^2 n)}$	$\boldsymbol{O(n^3)}$	**This paper**

2 The Algorithm

The overall idea is as follows. Let C^* be the optimal cutting cost. We shall have two phases in our algorithm: *box cutting phase* and *carving phase*. In the box cutting phase, we shall cut a minimum volume rectangular box B containing P out of Q with cutting cost no more than a constant factor of C^*. Then in the carving phase we shall cut P out of B with cutting cost bounded by $O(\log^2 n)$ times of C^*.

A cut is *vertex/edge/face cut* if it is tangent to P at a single vertex/a single edge/a face respectively. We call P to be *cornered* if it does not contain the center o of Q, otherwise it is called *centered*. For cornered P, the *D-separation* of P is the minimum-cost (single) cut that separates P from o. A point p of P is *visible* from o if the line segment $\overline{op}$ does not intersect any other point of P.

2.1 Box Cutting Phase

If P is cornered, we first apply a D-separation to Q.

Lemma 1. *The D-separation must be either a vertex, an edge or a face cut. Moreover, if $\overline{oo'}$ is the line segment perpendicular to the D-separation at o', then o' must be the corresponding vertex or a point of the corresponding edge or face.*

Proof. Let x be the closest point of P from o. Clearly, x is visible from o. A D-separation must be the plane that can separate o from x and is furthest from o. This plane is none but the plane perpendicular to $\overline{ox}$ at x. This plane is also tangent to P, since otherwise x would not be closest to o. □

Observe that since P is convex, the D-separation is unique.

Lemma 2. *The D-separation can be found in $O(n)$ time.*

Proof. By Lemma 1 we need to find the closest point x. To check whether x is a vertex of P, for each vertex v we draw a plane π_v perpendicular to $\overline{ov}$ at v. If π_v is tangent to P, then π_v is the D-separation. Checking π_v to be a tangent of P can be done in $O(d_v)$, where d_v is the degree of v. Over all v, it is $O(n)$.

To check whether x is a point of an edge e (similarly, a face f) of P, for each edge e (face f) we draw the line segment $\overline{oo'}$ perpendicular to the line l_e passing through e (to the supporting plane π_f of f). If o' is a point of l_e (π_f), then x is o'. □

For cornered P, after the D-separation is applied, Q is a spherical segment and let r be the radius of its base circle.

Lemma 3. *For cornered P, cost of the D-separation, which is πr^2, is at most C^*.*

Proof. [Sketch only] The proof depends upon the fact that the cuts in an optimal cutting sequence must be tangents to P. Overmars and Welzl [12] proved this fact for 2D, whose 3D generalization also holds. The idea is that if c is the first cut that does not touch P, then the cost of c and the subsequent cuts behaves, while moving c parallelly, as a concave function in the distance of c from P. Therefore, the minimum cost is achieved when it touches P or is infinitely away from P. With the above fact, the authors in [1] proved in 2D that to separate P from o an optimal cutting sequence must use the D-separation. The 3D generalization of this fact also holds. The main idea is that, to separate o from P if a single cut is used that is not a D-separation, then it must have cost more than the D-separation, since D-separation is the minimum such cut. If more than one cut are used, then their total cost would be even higher. □

A similar lemma for centered P is the following.

Lemma 4. *For centered P, $C^* \geq \pi R^2$, where R is the radius of Q.*

Proof. [Sketch only] Since P contains the center o of Q, any cutting sequence, starting from the boundary of Q, must wrap P and finally get out of Q by different location in the boundary of Q. That means the wrapping must enclose the center o. In the best case when P is the center o, the sequence must traverse at least $\frac{1}{2}\pi R^2$ area to reach P and need to traverse another $\frac{1}{2}\pi R^2$ area to finish the cutting. In the worst case when P is almost the sphere Q, the sequence must traverse the whole area of Q, which is $4\pi R^2$. □

We next find a minimum volume rectangular bounding box B of P in $O(n^3)$ time by the algorithm of O'Rourke [11]. Then we cut out this box from Q by applying six cuts along the six faces of B.

Lemma 5. *Cost of cutting B out of Q is at most $3C^*$ for cornered P and at most $4C^*$ for centered P.*

Proof. Let S be the surface of Q. For cornered P, area $|S| \leq 3\pi r^2 \leq 3C^*$ (by Lemma 3) and for centered P, $|S| = 4\pi R^2 \leq 4C^*$ (by Lemma 4). While cutting along the faces of B, for each cut c let Q' be the portion of Q that does not contain P. Let q' be the portion of the surface of Q' that is "inherited" from S, i.e., that was a part of the surface of S. One important observation is that the cost of c is no more than the area of q'. Moreover, over all six cuts, sum of these inherited surface area is $|S|$. Therefore, the lemma holds. □

Once the minimum area bounding box B has been cut, the lower bound on cutting cost can be given in terms of the area of B.

Lemma 6. *$C^* \geq \frac{1}{6}|B|$, where $|B|$ is the area of B.*

Proof. Let h be a maximum area face of B. Project P orthogonally from the direction perpendicular to h. P projects to a convex polygon X. In this projection, h is the minimum area bounding rectangle of X, since otherwise we could rotate the four faces of B that are not perpendicular to h and would get a bounding rectangle smaller than h, which in turn would give a bounding box smaller than B, but that is a contradiction that B is the smallest bounding box. It implies that the area of X is at least $\frac{1}{2}|h|$. Now, C^* is at least twice the area of X, and $|B| \leq 6|h|$. Therefore, $C^* \geq 2|X| \geq 2 \cdot \frac{1}{2}|h| \geq \frac{1}{6}|B|$. □

2.2 Carving Phase

Let $T = B - P$ be the portion of B that is "trapped" between the boundaries of P and B. T is a polyhedral object, convex or non-convex and possibly disconnected. The *inner* (*outer*) surface of T is the surface that touches (does not touch) the faces of P. Our idea is to apply an edge cut through each edge of P, and we shall do that in two types of rounds: *face rounds* and *edge rounds*. Face rounds will find polygonal chains that will partition the faces of P into smaller connected components and edge rounds will apply edge cuts through the edges of those polygonal chains. There will be $O(\log n)$ face rounds. Within each face round there will be a number of edge rounds but their total cost will be $O(C^* \log n)$. Once we have applied edge cuts through all the edges of P, each face f of P will have a small "cap"-like portion of T over it, which we shall cut at a cost of the area of f to get P, giving a cost of $O(C^*)$ for all faces.

Face Rounds. Let F be a *connected face set* of l faces of P. At the very first face round $i = 0$, F consists of all the faces of P. We find a chain of edges P' that will partition F into two smaller connected face sets F_1 and F_2 by the following lemma.

Lemma 7. *It is always possible to find in $O(l \log l)$ time an orthogonal projection of P which is non-degenerate w.r.t the faces of F such that the sets of visible and invisible faces of F contain at least $\lfloor \frac{l}{2} \rfloor$ faces each.*

Proof. For this proof we shall move on to the surface of an origin-centered sphere s. For each face $f \in F$, its outward normal is uniquely represented by a point of s, which we call the *normal point* of f. Each point of s also represents an orthogonal projection direction of P. So, an orthogonal projection of P which is non-degenerate w.r.t the faces of F is represented by a great circle of s that does not pass through the normal points of the faces of F. We need one such great circle satisfying an additional criterion that its two hemispheres contain at least $\lfloor \frac{l}{2} \rfloor$ normal points each. There exists infinitely many such great circles and one of them can be found in $O(l \log l)$ time as follows. Take any two antipodal points that are normal points as poles. Take a great circle g through these two poles and rotate it around these poles until the number of normal points in its two hemisphere differ by at most one. If it happens that some normal points fall on the great circle, then slightly change the poles and the great circle to distribute those normal points into two hemispheres as necessary. For running time, all we need to do is to sort the normal points according to their angular distance with the plane of g at the origin. □

We call the projection direction to achieve g by the above lemma the *zone direction* of of F. P' is the chain of edges in the boundary of the above projection whose each edge has both adjacent faces (one is visible and another is invisible) in F. We call P' a *separating chain* of F. We shall apply edge cuts through the edges of P' by the edge rounds as described in the next paragraph. In the next face round $i+1$, we shall apply Lemma 7 for each of F_1 and F_2 and shall thus get two separating chains and four connected face sets. We shall repeat the same procedure for each of these four face sets. We shall continue like this until each face set has only one face. Clearly, we need $O(\log n)$ face rounds.

Edge Rounds. Let $P' = e_1, e_2, \ldots, e_k$ with its two ends from e_1 and e_k touching the outer surface of T. We shall apply edge cuts through the edges of P' such that all of them are parallel to a particular direction. Such a direction can be the corresponding projection direction. We call this set of k edge cuts a *zone* of cuts and its direction the *zone cut direction.* We shall apply these cuts in $\log k$ edge rounds. At the very first edge round $j = 0$, we apply an edge cut through $e_{k/2}$ in the zone cut direction. This cut will partition the edges of P' into two subchains of size at most $\lfloor \frac{k}{2} \rfloor$. In the next round, we apply two edge cuts through the two middle edges of these two subchains, which will result into four subchains. Then in the next round we apply four similar cuts to the four subchains. We continue like this until each subchain has only one edge. Clearly, we need $O(\log k)$ edge rounds for P'.

Lemma 8. *After all the face rounds and the corresponding edge rounds are completed, all edges of P get an edge cut.*

Proof. Let e be an edge that does not get an edge cut. Then the two adjacent faces of e are in the same face set. But that is a contradiction that each face set has only one face. □

Analysis. We define the *box area* of a face set F as follows. When F contains all faces of P, its box area is B —the whole surface area of B. Zone of cuts through the separating chain of F partitions F into F_1 and F_2 and T into two components, say T_1 and T_2, respectively. Then the *box area* of F_1 (F_2) is the outer surface area of T_1 (T_2), which we denote by by B_1 (B_2). Observe that $|B_1| + |B_2| \leq |B|$. Box area of any subsequent face set is similarly defined. Moreover, two face sets from the same face round have their box areas disjoint and in any face round sum of all box area is at most $|B|$.

Lemma 9. *Let ${P'}_m$ be the separating chain with k edges of an arbitrary face set F_m to which we apply $O(\log k)$ edge rounds. Let B_m be the box area of F_m. At each edge round j, total cost of 2^j cuts is $O(|B_m|)$. Over all $\log k$ edge rounds, total cost is $O(|B_m| \log n)$.*

Proof. This proof is similar to that of Lemma 5. Consider a particular edge round j. For each cut c the cost of c is no more than the portion of B_m that is thrown away by c. Moreover, these cuts are pairwise disjoint. Indeed, they can at best intersect the cut which is in between them and was applied in $(j-1)$-th round. It implies that the total cost of 2^j cuts is at most $|B_m|$. Since $k \leq n$, the second part of the lemma follows. □

Lemma 10. *Let F be the face set consisting of all faces of P to which we shall apply $O(\log n)$ face rounds. At each face round i, total cost of 2^i zones of cuts is $O(|B| \log n)$. Over all $O(\log n)$ face rounds, the total cost is $O(C^* \log^2 n)$.*

Proof. At each face round i, we apply 2^i zones of cuts to 2^i face sets. By the previous lemma, for a particular face set F_m, $0 \leq m \leq 2^i$, cost of the zone of cuts applied to it is at most $O(|B_m| \log n)$. Since $\sum_1^{2^i} |B_m| \leq |B|$, cost of all zone cuts is $\sum_1^{2^i} O(|B_m| \log n) = O(|B| \log n)$. Over all $O(\log n)$ face rounds, the total cost is $O(|B| \log^2 n)$, which by Lemma 6 is $O(C^* \log^2 n)$. □

Running time in face round i involves finding 2^i separating chains, each of size $\frac{n}{2^i}$, plus applying a zone of cuts to each of them. Each separating chain can be found in $O(\frac{n}{2^i} \log \frac{n}{2^i})$ time by Lemma 7. Each cut needs to update Q, which "can be done" in $O(n)$ time assuming that Q is represented by suitable data structure [6]. It gives that a zone of cuts needs $O(\frac{n^2}{2^i})$ time. So, in round i total time is $O(2^i(\frac{n^2}{2^i} + \frac{n}{2^i} \log \frac{n}{2^i})) = O(n^2)$. Over all $O(\log n)$ rounds, it becomes $O(n^2 \log n)$.

Theorem 1. *Given a convex polyhedron P fixed inside a sphere Q, P can be cut out of Q by using only guillotine cuts in $O(n^3)$ time with cutting cost $O(\log^2 n)$ times the optimal, where n is the number of vertices of P.*

3 Conclusion

In this paper, we have given an $O(n^3)$-time algorithm that cuts a convex polyhedron P with n vertices from a sphere Q, where P is fixed inside Q, by using guillotine cuts with cutting cost $O(\log^2 n)$ times the optimal.

This problem is well studied in 2D, where the series of results include several $O(\log n)$ and constant factor approximation algorithms and a PTAS. The key ingredients of the 2D algorithms involve three major steps: (1) take some approximate vertex cuts through the vertices of P, (2) use dynamic programming to find an optimal cutting sequence among the edge cuts and the vertex cuts taken in step (1), and (3) show that the cutting cost of the sequence obtained in step (2) is within the desired factor of the optimal. Using the idea of 2D algorithms may be a way to improve the approximation ratio of our algorithm. Among the above three steps, it may not be difficult to generalize steps (1) and (3) for 3D, but the most difficult part we find is the applying a dynamic programming.

References

1. Ahmed, S.I., Hasan, M., Islam, M.A.: Cutting a cornered convex polygon out of a circle. Journal of Computers (to appear), `http://203.208.166.84/masudhasan/cut.pdf`
2. Bereg, S., Daescu, O., Jiang, M.: A PTAS for cutting out polygons with lines. In: Chen, D.Z., Lee, D.T. (eds.) COCOON 2006. LNCS, vol. 4112, pp. 176–185. Springer, Heidelberg (2006)
3. Bhadury, J., Chandrasekaran, R.: Stock cutting to minimize cutting length. Euro. J. Oper. Res. 88, 69–87 (1996)
4. Chandrasekaran, R., Daescu, O., Luo, J.: Cutting out polygons. In: CCCG 2005, pp. 183–186 (2005)
5. Daescu, O., Luo, J.: Cutting out polygons with lines and rays. International Journal of Computational Geometry and Applications 16, 227–248 (2006)
6. de Berg, M., van Kreveld, M., Overmars, M., Schwarzkopf, O.: Computational Geometry: Algorithms and Applications, 2nd edn. Springer, Heidelberg (2000)
7. Demaine, E.D., Demaine, M.L., Kaplan, C.S.: Polygons cuttable by a circular saw. Computational Geometry: Theory and Algorithms 20, 69–84 (2001)
8. Dumitrescu, A.: An approximation algorithm for cutting out convex polygons. Computational Geometry: Theory and Algorithms 29, 223–231 (2004)
9. Dumitrescu, A.: The cost of cutting out convex n-gons. Discrete Applied Mathematics 143, 353–358 (2004)
10. Jaromczyk, J.W., Kowaluk, M.: Sets of lines and cutting out polyhedral objects. Computational Geometry: Theory and Algorithms 25, 67–95 (2003)
11. O'Rourke, J.: Finding minimal enclosing boxes. International Journal of Computer and Information Sciences 14(3), 183–199 (1985)
12. Overmars, M.H., Welzl, E.: The complexity of cutting paper. In: SoCG 1985, pp. 316–321 (1985)
13. Tan, X.: Approximation algorithms for cutting out polygons with lines and rays. In: Wang, L. (ed.) COCOON 2005. LNCS, vol. 3595, pp. 534–543. Springer, Heidelberg (2005)
14. Toussaint, G.: Solving geometric problems with the rotating calipers. In: MELECON 1983, Athens, Greece (1983)