

# Decomposing a Simple Polygon into Pseudo-Triangles and Convex Polygons\*

Stefan Gerdjikov<sup>†</sup>      Alexander Wolff<sup>‡</sup>

Invited to the EWCG'06 special issue of CGTA  
Received: June 21, 2006. Revised: October 25, 2007

## Abstract

In this paper we consider the problem of decomposing a simple polygon into subpolygons that exclusively use vertices of the given polygon. We allow two types of subpolygons: pseudo-triangles and convex polygons. We call the resulting decomposition *PT-convex*. We are interested in *minimum* decompositions, i.e., in decomposing the input polygon into the least number of subpolygons. Allowing subpolygons of one of two types has the potential to reduce the complexity of the resulting decomposition considerably.

The problem of decomposing a simple polygon into the least number of *convex* polygons has been considered. We extend a dynamic-programming algorithm of Keil and Snoeyink for that problem to the case that both convex polygons and pseudo-triangles are allowed. Our algorithm determines such a decomposition in  $O(n^3)$  time and space, where  $n$  is the number of the vertices of the polygon.

## 1 Introduction

The problem of decomposing the convex hull of a set of points into subpolygons has a long history. We are interested in decompositions where the vertices of the subpolygons are restricted to the set of input points. Triangulations are an example of such decompositions. Every triangulation of a set of  $n$  points consists of  $2n - 2 - c$  triangles where  $c$  is the number of points on the convex hull. Thus one is usually interested in triangulations that optimize some additional parameter. For example, the Delaunay triangulation is known to maximize the smallest angle over all triangles in the triangulation. Another famous example is the minimum-weight triangulation. It minimizes the sum over the lengths of all edges in the triangulation. The complexity of computing this triangulation was open for a long time, until Mulzer and Rote [MR06] very recently managed to show that the problem is NP-hard. Krznaric and Levkopoulos [KL98] have shown that there is a constant-factor approximation for the problem, but the factor of their approximation is so large that they have not explicitly calculated it.

The concept of triangulations has been generalized by considering decompositions that consist of other, more complex and thus potentially fewer subpolygons. Two natural generalizations of triangles are (a) convex polygons and (b) pseudo-triangles. *Pseudo-triangles* are simple polygons with exactly three convex angles, i.e., interior angles of less than  $180^\circ$ . They have applications in visibility complexes [PV96], ray shooting [CEG<sup>+</sup>94, GT97], rigidity theory and robot arm motion planning [Str00], guarding polygons [ST05], and kinetic collision detection [KSS02].

Concerning the first generalization of triangles, Fevens et al. [FMR01] have investigated minimum convex decompositions, i.e., decompositions that consist of the least number of convex

---

\*Work supported by grant WO 758/4-2 of the German Research Foundation (DFG).

<sup>†</sup>Faculty of Mathematics and Informatics, Sofia University “Sv. Kliment Ohridski”. Email: [st\\_gerdjikov@abv.bg](mailto:st_gerdjikov@abv.bg)

<sup>‡</sup>Faculteit Wiskunde en Informatica, Technische Universiteit Eindhoven, Eindhoven, the Netherlands. WWW: <http://www.win.tue.nl/~awolff>

subpolygons. Their algorithm takes  $O(n^{3h+3})$  time, where  $h$  is the number of nested convex hulls of the given point set. Spillner [Spi05] has given a fixed-parameter algorithm for the problem, the number of points in the interior of the convex hull being the parameter. For the case that points are in general position, Knauer and Spillner [KS06] have given a simple 3-approximation that runs in  $O(n \log n)$  time and a more involved 30/11-approximation that runs in  $O(n^2)$  time.

Concerning the second generalization of triangles, Streinu [Str00] has shown strong links between minimally rigid graphs and minimum pseudo-triangulations. In particular, she proved that the minimum number of edges needed to obtain a pseudo-triangulation is  $2n - 3$  and thus, by Euler’s polyhedron theorem, the number of pseudo-triangles in a minimum pseudo-triangulation is  $n - 2$ , which does not depend on the structure of the point set (given general position) but only on its size. There has also been work on enumerating all minimum pseudo-triangulations [ARSS03, Ber05]. Gudmundsson and Levcopoulos [GL07] investigate the problem of computing minimum-*weight* pseudo-triangulations for sets of points, where the weight of a decomposition is the sum over the lengths of all edges in the decomposition. They approximate the problem in two ways. Given a set of  $n$  points, their first algorithm computes a pseudo-triangulation whose weight is at most  $O(\log n)$  times larger than that of a minimum spanning tree of the same point set. In contrast, they show there are point sets where every convex decomposition (and thus every triangulation) has weight  $\Omega(n)$  times that of a minimum spanning tree. Their second algorithm computes in cubic time a pseudo-triangulation whose weight is at most 15 times that of a minimum-weight pseudo-triangulation.

Aichholzer et al. [AHK<sup>+</sup>06] were the first to investigate decompositions where each subpolygon has the choice to be convex or a pseudo-triangle, i.e., one of the two generalizations of triangles mentioned above. We call such decompositions *PT-convex*. They show that each minimum PT-convex decomposition of a set of  $n$  points consists of less than  $7n/10$  polygons. In contrast, there are point sets where any minimum *convex* decomposition consists of at least  $12n/11 - 2$  subpolygons [GN06]. (On the other hand, Knauer and Spillner [KS06] showed that every point set can be decomposed into no more than  $15n/11 - 24/11$  convex polygons.)

A related problem is the decomposition of *simple polygons* into convex polygons or pseudo-triangles, e.g., for point location or ray shooting. When decomposing a simple polygon we also say that the decomposition is convex, a pseudo-triangulation or PT-convex if the decomposition uses exclusively the corresponding types of polygons. Again we are interested in minimum decompositions. Keil [Kei85] has given a general technique for decomposing a simple polygon into polygons of a certain type. The technique is based on optimally decomposing subpolygons each of which is obtained from the original by drawing a single diagonal. Keil’s technique yields an  $O(nr^2 \log r)$ -time algorithm for the convex decomposition problem, where  $r$  is the number of *reflex* vertices of the polygon, i.e., vertices whose inner angle is larger than  $180^\circ$ . Keil also showed that the convex decomposition problem becomes NP-hard if the input polygons can have holes. Keil and Snoeyink [KS02] improve Keil’s algorithm by giving a  $O(n + \min(nr^2, r^4))$ -time and -space solution. Interestingly, the problem can be solved faster, namely in  $O(n + r^3)$  time, when allowing Steiner points [CD85].

In the above-mentioned paper [GL07], Gudmundsson and Levcopoulos also give a cubic-time algorithm for computing a minimum-weight pseudo-triangulation of a simple polygon. They use this algorithm as a sub-routine for their 15-approximation of the minimum-weight pseudo-triangulation of a set of points.

In this paper we give an algorithm for computing a minimum PT-convex decomposition of a simple polygons. Our dynamic-programming algorithm is based on two main ingredients: Keil’s general decomposition technique [Kei85] and the way how Gudmundsson and Levcopoulos [GL07] determine all geodesics in the given polygon which form chains of reflex vertices and can thus potentially be sides of pseudo-triangles. Our algorithm takes  $O(n^3)$  time and space.

Our paper is structured as follows. We first briefly describe the approach of Keil [Kei85] and Keil and Snoeyink [KS02] in Section 2. Then we characterize pseudo-triangles in terms of chains of reflex vertices and vice versa, see Section 3. We present our algorithm in Section 4 and analyze its running time in Section 5. Finally, we give some open problems in Section 6.

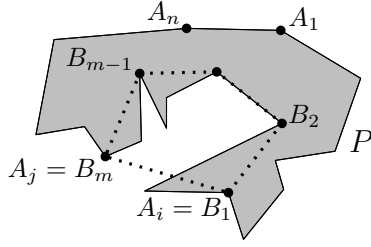


Fig. 1: The geodesic  $B_1B_2 \dots B_m$  from  $A_i$  to  $A_j$  is concave with respect to the simple polygon  $P$ .

## 2 Previous Work

Keil [Kei85] introduces a general technique for decomposing a simple polygon into polygons of a certain type. The technique is based on optimally decomposing subpolygons each of which is obtained from the original by drawing a single diagonal  $d$ . In each decomposition  $\mathcal{D}$  of a subpolygon there is a unique polygon  $P(\mathcal{D})$  that contains the diagonal.

Keil defines a relation  $D_1 \leq D_2$  between two minimum decompositions of a subpolygon if and only if the angles at  $d$  in the polygon  $P(D_1)$  are not greater than the corresponding angles in  $P(D_2)$  respectively. He argues that it suffices to consider decompositions that are minimal under this relation in order to find a minimum decomposition of  $P$ . Keil considers the equivalence classes which these minimal elements determine. Their representatives can be easily computed and can be used to check whether a given minimum decomposition can be extended without increasing the number of polygons. This idea yields an  $O(n^3 \log n)$ -time algorithm for the convex decomposition problem. Keil and Snoeyink [KS02] observe that (a) once a representative cannot be used to extend a decomposition it can be discarded and that (b) only diagonals incident on at least one reflex vertex need to be considered. This results in an  $O(nr^2)$ -time algorithm, where  $r$  is the number of reflex vertices of the polygon. Observation (a) helps us to obtain an  $O(n^3)$ -time algorithm for our problem.

## 3 Characterization of Pseudo-Triangles

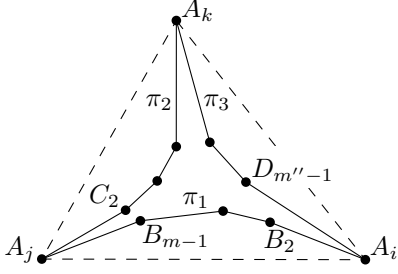
We use  $P^+(A_i, A_j)$  and  $P^-(A_i, A_j)$  to denote the paths on the boundary  $\partial P$  from a vertex  $A_i$  to a vertex  $A_j$  of  $P$  in clockwise and counter-clockwise direction, respectively. To simplify the notation we will always assume that the edge  $A_nA_1$  does not lie in the part of the polygon that we currently investigate; for three vertices  $A_i, A_j$ , and  $A_k$ , this means that we can write  $i < j < k$  if we mean that vertex  $A_j$  lies on the path  $P^+(A_i, A_k)$  (see Lemma 1, for an example).

We say that a point  $Q \in P$  is *visible* from a point  $Q' \in P$  if the relative interior of the line segment  $QQ'$  is contained in the interior of  $P$  or if  $Q$  and  $Q'$  are adjacent vertices of  $P$ ; note that the *relative interior* of a line segment is the set of all the points of the segment except for the endpoints. With  $\text{vis}(A_i)$  we denote the list of all vertices of  $P$  that are visible from  $A_i$  in clockwise order starting with  $A_{i+1}$ . Unless stated otherwise, we number the vertices of a polygon in clockwise order.

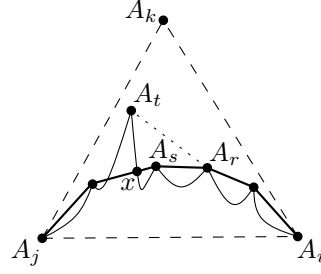
**Definition 1** Let  $P = A_1A_2 \dots A_n$  be a simple polygon. A path  $\Pi = B_1B_2 \dots B_m$  from  $A_i = B_1$  to  $A_j = B_m$  is a concave geodesic with respect to the polygon  $P$  if  $\Pi$  satisfies the following three conditions, see Figure 1:

(G1) for each  $k < m$  it holds that  $B_{k+1}$  is the last vertex on  $P^+(B_k, A_j)$  which is visible from  $B_k$ , and

(G2)  $B_1B_2 \dots B_m$  is a convex, counter-clockwise oriented polygon.



**Fig. 2:** Three consecutive concave geodesics  $\pi_1$ ,  $\pi_2$ , and  $\pi_3$  define a pseudo-triangle.



**Fig. 3:** Sketch for the proof of Lemma 1; arcs represent the boundary of the polygon  $P^+(A_i, A_j)$ , solid straight-line segments are the edges of  $\pi_1 = T^+(A_i, A_j)$ .

**Remark 1** If  $B_1B_2 \dots B_m$  is a concave geodesic from  $B_1$  to  $B_m$  with respect to a simple polygon  $P$  and  $m \geq 3$ , then  $B_2 \dots B_m$  is a concave geodesic from  $B_2$  to  $B_m$  with respect to  $P$ . On the other hand, given two vertices  $A_i$  and  $A_j$  of a simple polygon  $P$ , there is at most one concave geodesic that connects  $A_i$  and  $A_j$ . Thus there are  $O(n^2)$  concave geodesics in a simple polygon with  $n$  vertices.

For our further considerations we need the following observation.

**Observation 1** ([KS02]) Let  $A_i$  be a vertex of  $P = A_1A_2 \dots A_n$ . Consider the line segments  $A_iA_j$  with  $A_j \in \text{vis}(A_i)$ . Their cyclic order around  $A_i$  is the same as the order of their other endpoints along  $\partial P$ .

We now state the relationship between pseudo-triangles and concave geodesics in a simple polygon.

**Lemma 1** Let  $P = A_1A_2 \dots A_n$  be a simple polygon and let  $T \subseteq P$  be a pseudo-triangle whose vertices are vertices of  $P$  and whose convex vertices are  $A_i$ ,  $A_j$  and  $A_k$  with  $i < j < k$ . Then the paths  $\pi_1 = T^+(A_i, A_j)$ ,  $\pi_2 = T^+(A_j, A_k)$ , and  $\pi_3 = T^+(A_k, A_i)$  are concave geodesics with respect to  $P$ .

*Proof.* First note that the vertices of  $\pi_1$  lie on  $P^+(A_i, A_j)$ , those of  $\pi_2$  lie on  $P^+(A_j, A_k)$ , and those of  $\pi_3$  lie on  $P^+(A_k, A_i)$  otherwise  $T$  would not be simple. To avoid double indices, let  $B_1B_2 \dots B_m = \pi_1$ ,  $C_1C_2 \dots, C_{m'} = \pi_2$ , and  $D_1D_2 \dots, D_{m''} = \pi_3$ , see Figure 2. We now check the three properties of concave geodesics for  $\pi_1$ . Again, to simplify the notation, we assume that  $\pi_1$  does not contain the edge  $A_nA_1$ . The proofs for  $\pi_2$  and  $\pi_3$  are symmetric.

Now we establish property (G1). We consider the path  $\pi_1$  and show that the construction proposed in property (G1) does not fail. Assume to the contrary that there is an index  $r \in \{i, \dots, j-1\}$  such that  $A_r$  violates the construction proposed in property (G1). See Figure 3. Let  $A_s$  be the vertex on  $\pi_1$  immediately following  $A_r$ . Then it is clear that  $s > r$ . Now let  $t \in \{s+1, \dots, j\}$  be such that  $A_t$  is visible from  $A_r$ . Due to Observation 1 we know that the edges  $A_rA_{r+1}$ ,  $A_rA_s$  and  $A_rA_t$  appear in clockwise order around  $A_r$ . In particular, because of the convexity of  $T^+(A_r, A_j)A_r$ , the edge  $A_rA_t$  intersects  $T^+(A_r, A_j)$  only in  $A_r$  and the edge  $A_rA_s$  is contained in the polygon  $P^+(A_r, A_t)A_r$ . However,  $A_j$  lies outside this polygon and thus  $T^+(A_r, A_j)$  leaves  $P^+(A_r, A_t)A_r$  in some point  $x$  which does not belong to  $A_rA_t$ , see Figure 3. Hence  $T^+(A_r, A_j)$  leaves  $P$ . This contradicts the fact that  $T$  is contained in  $P$ . Thus the assumption that property (G1) fails at some point is wrong. This shows that  $\pi_1$  actually does satisfy property (G1).

Finally we show that property (G2), i.e., convexity, holds for the polygon  $B = B_1B_2 \dots B_{m'}$ . Consider the ray that emanates from  $A_i$  and goes through  $B_2$ . Note that the line segment  $A_iB_2$  lies in  $P$  since  $T$  is contained in  $P$ . Due to Observation 1 we know that if we turn the ray in clockwise direction, it will hit  $D_{m''-1}$ . During this movement the part of the ray in a small neighborhood of  $A_i$  will remain in  $P$ . Denote by  $r$  the ray in an arbitrary position during this movement. Define

an analogous ray  $r'$  emanating from  $A_j$ . Since the chain  $\pi_1$  has reflex angles at  $B_2, \dots, B_{m-1}$  in the interior of  $T$  (which correspond to convex angles in the interior of  $B$ ), the chain cannot leave the triangle  $\Delta$  formed by the line segment  $A_i A_j$  and the two rays  $r$  and  $r'$ . Since  $\Delta$  has convex angles in  $A_i = B_1$  and  $A_j = B_{m'}$ , this also holds for  $B$ . Thus all angles in  $B$  are convex, and so is  $B$ .  $\square$

Next we establish the converse relation: three concave geodesics determine a pseudo-triangle.

**Lemma 2** *Let  $P = A_1 A_2 \dots A_n$  be a simple polygon. Further let  $i < j < k$  and  $\pi_1 = A_i \dots A_j$ ,  $\pi_2 = A_j \dots A_k$  and  $\pi_3 = A_k \dots A_i$  be concave geodesics with respect to  $P$ . Then  $\pi_1 \pi_2 \pi_3$  is a pseudo-triangle.*

*Proof.* To avoid double indices let  $\pi_1 = A_i \dots A_j = B_1 B_2 \dots B_m$  and  $\pi_2 = A_j \dots A_k = C_1 C_2 \dots C_{m'}$ , see Figure 2. We first consider the geodesics  $\pi_1$  and  $\pi_2$  and show that they are disjoint except for the vertex  $A_j$  where  $\pi_1$  meets  $\pi_2$  in a convex angle.

Rotate a ray counter-clockwise around  $A_j$  starting at  $A_i$ . Due to property (G2) in Definition 1 the ray sweeps over the vertices  $A_i = B_1, \dots, B_{m-1}$  of  $\pi_1$  in this order. Due to Observation 1 the ray then hits all vertices of  $P^-(A_i, A_k)$  visible from  $A_j$ . Due to property (G1) the next vertex hit by the ray is  $C_2$ . Again due to (G2) the ray then hits the vertices  $C_3, \dots, C_{m'} = A_k$  of  $\pi_2$  in this order. This shows that  $\pi_1$  and  $\pi_2$  do not intersect and that the angle  $\angle B_{m-1} A_j C_2$  is convex. Symmetric arguments show that analogous statements hold for the other two pairs  $(\pi_2, \pi_3)$  and  $(\pi_3, \pi_1)$  of geodesics. Due to property (G2) all vertices of  $\pi_1 \pi_2 \pi_3$  other than  $A_i, A_j$ , and  $A_k$  are reflex. Thus  $\pi_1 \pi_2 \pi_3$  is a simple polygon with exactly three convex vertices, i.e., a pseudo-triangle.  $\square$

## 4 Algorithm

We use the same approach for finding a minimum PT-convex decomposition of a simple polygon as Keil and Snoeyink [KS02] use for finding the minimum *convex* decomposition of a polygon, i.e., we consider subpolygons which are obtained from the original polygon by drawing a single diagonal. As Keil and Snoeyink we use dynamic programming, treating subpolygons in order of increasing number of vertices. For each subpolygon  $P'$  of the given polygon  $P$  we consider the diagonal  $d$  that separates  $P'$  from  $P$ . We compute two decompositions of  $P'$ , namely one that is minimum under the constraint that  $d$  bounds a convex face of  $P'$  and one that is minimum given that  $d$  bounds a pseudo-triangle in  $P'$ .

We compute the smallest decomposition of the first type analogously to Keil and Snoeyink. For the second type of decomposition where the diagonal  $d$  bounds a pseudo-triangle we proceed as follows. Assume we have a precomputed list  $L$  of all concave geodesics with respect to  $P$ . Then we can filter  $L$  to find all pseudo-triangles that contain the diagonal  $d$  as an edge. For each such pseudo-triangle  $T$  we compute the size of a minimum decomposition that contains  $T$ . Among all these decompositions we keep the smallest.

Now we describe our algorithm in detail. Let  $P = A_1 A_2 \dots A_n$  be a simple polygon. We use definitions similar to those of Keil and Snoeyink. If  $i < j$  and  $A_j$  is visible from  $A_i$  in  $P$  then we denote the line segment  $A_i A_j$  by  $d_{ij}$  and call it a *diagonal* of  $P$ . Note that by our definition of visibility, each edge of  $P$  is a diagonal. Each diagonal defines a (simple) subpolygon  $P_{ij} = A_i A_{i+1} \dots A_j$  contained in  $P$ . If the diagonal is an edge of  $P$ , the subpolygon is empty.

**Definition 2** *Let  $\mathcal{D}_{ij}$  denote the set of all PT-convex decompositions of a polygon  $P_{ij}$  and let*

$$\begin{aligned} w_{ij} &= \min\{|\mathcal{D}| : \mathcal{D} \in \mathcal{D}_{ij}\}, \\ cw_{ij} &= \min\{|\mathcal{D}| : \mathcal{D} \in \mathcal{D}_{ij} \text{ and the diagonal } d_{ij} \text{ is an edge of a convex polygon of } \mathcal{D}\}, \\ pw_{ij} &= \min\{|\mathcal{D}| : \mathcal{D} \in \mathcal{D}_{ij} \text{ and the diagonal } d_{ij} \text{ is an edge of a pseudo-triangle of } \mathcal{D}\}. \end{aligned}$$

Since the polygons  $P_{i,i+1}$  ( $i = 1, \dots, n$ ) are degenerate we set  $w_{i,i+1} = cw_{i,i+1} = pw_{i,i+1} = 0$ . Note also that  $w_{ij} = \min(cw_{ij}, pw_{ij})$ .

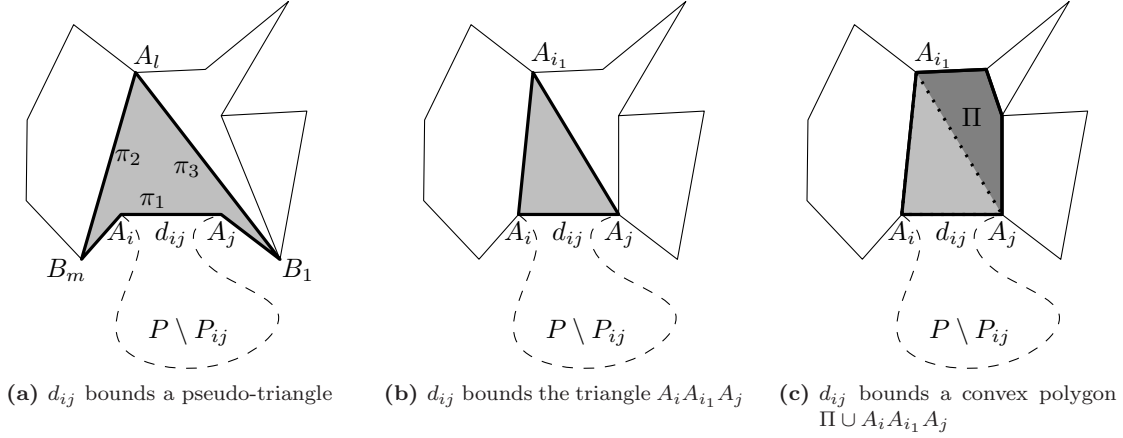


Fig. 4: Three types of minimum PT-convex decompositions of  $P_{ij}$ .

#### 4.1 Computation of $pw_{ij}$

We describe how to find  $pw_{ij}$  given the values  $w_{kl}$  for each pair  $(k, l)$  with  $(l - k) \bmod n < (j - i) \bmod n$  and a list  $L$  of all concave geodesics for the polygon  $P$ . We consider all concave geodesics which contain the edge  $A_i A_j$  and lie completely in  $P_{ij}$ . For each such geodesic  $\pi_1 = B_1 B_2 \dots B_m$  we go along  $P^-(B_1, B_m)$  and for each vertex  $A_l \in P^-(B_1, B_m)$  we check whether there exist concave geodesics  $\pi_2 = B_m \dots A_l$  and  $\pi_3 = A_l \dots B_1$ . If  $\pi_2$  and  $\pi_3$  exist, then  $\pi_1 \pi_2 \pi_3$  is a pseudo-triangle according to Lemma 2, see Figure 4a. A minimum decomposition of  $P_{ij}$  containing this pseudo-triangle can be obtained if and only if for each pair  $(k, l) \neq (i, j)$  such that  $A_k A_l$  is an edge of  $\pi_1 \pi_2 \pi_3$  the polygon  $P_{kl}$  is optimally decomposed.

Thus if  $w(\pi)$  denotes the sum of all  $w_{kl}$  where  $A_k A_l$  lies on a geodesic  $\pi$ , then it is clear that the minimum decomposition of  $P_{ij}$  using the pseudo-triangle  $\pi_1 \pi_2 \pi_3$  consists of

$$s(\pi_1, A_l) = 1 + w(\pi_2) + w(\pi_3) + \sum_{A_k A_l \in \pi_1 \setminus \{A_i A_j\}} w_{kl}$$

polygons. Now  $pw_{ij}$  is the minimum of  $s(\pi_1, A_l)$  over all pairs  $(\pi_1, A_l)$  that fulfill the above requirements.

#### 4.2 Computation of $cw_{ij}$

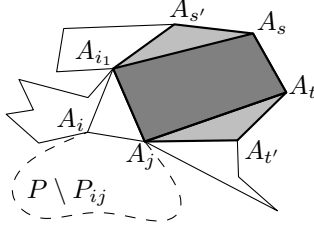
In this section we describe how to compute  $cw_{ij}$  if we have the values  $cw_{kl}$  and  $w_{kl}$  whenever  $(l - k) \bmod n < (j - i) \bmod n$ . Our approach is based on the algorithm of Keil and Snoeyink [KS02] for computing a minimum convex decomposition of a simple polygon. We start with the following definition.

**Definition 3** A PT-convex decomposition  $\mathcal{D}$  of  $P_{ij}$  is called diagonal-convex if the diagonal  $d_{ij}$  is an edge of a convex polygon  $A_i A_{i_1} \dots A_{i_m} A_j$  in  $\mathcal{D}$ , where  $m > 1$ .

Given a diagonal-convex decomposition  $\mathcal{D}$  of  $P_{ij}$  that contains a convex polygon  $A_i A_{i_1} \dots A_{i_m} A_j$  it is clear that the triangle  $A_i A_{i_1} A_j$  is contained in  $P$ . Moreover, if  $m > 1$ , then  $A_{i_1} A_{i_2} \dots A_j$  is also a (non-degenerate) convex polygon and thus  $\mathcal{D}$  induces a diagonal-convex decomposition of  $P_{i_1 j}$ .

Now let us change the point of view. Consider a triangle  $A_i A_{i_1} A_j$  with  $i < i_1 < j$  that is contained in  $P$ . Then decomposing  $P_{i i_1}$  and  $P_{i_1 j}$  optimally and adding the triangle  $A_i A_{i_1} A_j$  yields a diagonal-convex decomposition of  $P_{ij}$ , see Figure 4b. This decomposition  $\mathcal{D}^*$  consists of

$$w_{i i_1} + w_{i_1 j} + 1$$



**Fig. 5:** The pair of indices  $(s, t)$  is a representative of  $P_{i_1 j}$ , while  $(s', t')$ ,  $(s, s)$ , and  $(t, t)$  are not.

polygons. Now the question is whether we can do better. Can we extend the triangle  $A_i A_{i_1} A_j$  into a larger convex polygon  $A_i A_{i_1} A_{i_2} \dots A_j$ ? This is possible if and only if there is a diagonal-convex decomposition  $\mathcal{D}'$  of  $P_{i_1 j}$  containing a convex polygon  $\Pi = A_{i_1} A_{i_2} \dots A_{i_m} A_j$  with the additional property that

$$\angle A_{i_2} A_{i_1} A_i < 180^\circ \quad \text{and} \quad \angle A_i A_j A_{i_m} < 180^\circ. \quad (1)$$

Then we can merge the triangle  $A_i A_{i_1} A_j$  with  $\Pi$ . This yields a diagonal-convex decomposition of  $P_{ij}$  consisting of  $w_{i_{i_1}} + |\mathcal{D}'|$  polygons. Note that if  $\mathcal{D}'$  is not minimum, we have  $w_{i_{i_1}} + |\mathcal{D}'| \geq w_{i_{i_1}} + cw_{i_1 j} + 1 \geq w_{i_{i_1}} + w_{i_1 j} + 1 = |\mathcal{D}^*|$ , see Figure 4c. Since we consider  $\mathcal{D}^*$ , we can ignore non-minimum decompositions  $\mathcal{D}'$ . (Observe that there is no need to check whether the triangle  $A_i A_{i_1} A_j$  can be extended by a convex polygon  $A_i A_k \dots A_{i_1}$  that is adjacent to edge  $A_i A_{i_1}$ ; such a decomposition will be considered when we process the triangle  $A_i A_k A_j$ .)

It only remains to show how to test condition (1) efficiently. We do this similarly to Keil and Snoeyink [KS02]. First observe that in order to check condition (1) one only needs access to vertices  $A_{i_2}$  and  $A_{i_m}$  rather than to the whole polygon  $\Pi$ . Next consider two minimum diagonal-convex decompositions  $\mathcal{D}$  and  $\mathcal{D}'$  of  $P_{i_1 j}$  with  $\mathcal{D} \neq \mathcal{D}'$ . Let  $\Pi = A_{i_1} A_s \dots A_t A_j$  and  $\Pi' = A_{i_1} A_{s'} \dots A_{t'} A_j$  be the corresponding convex polygons which contain the edge  $A_{i_1} A_j$ . Suppose that  $s' \leq s \leq t \leq t'$ . Then according to Observation 1, we have that

$$\angle A_i A_{i_1} A_s \leq \angle A_i A_{i_1} A_{s'} \quad \text{and} \quad \angle A_i A_j A_t \leq \angle A_i A_j A_{t'}.$$

Hence either  $\mathcal{D}'$  violates condition (1), or if  $\mathcal{D}'$  satisfies it, then  $\mathcal{D}$  also satisfies it. For an example, see Figure 5. In either case we can ignore  $\mathcal{D}'$  without risking to lose the optimum solution. This is the motivation for defining a pair  $(s, t)$  of vertex indices with  $i < s \leq t < j$  to be a *representative* for a polygon  $P_{ij}$  if the following three conditions hold.

(R1) The polygon  $P_{ij}$  has a minimum diagonal-convex decomposition that contains a convex polygon of the form  $A_i A_s \dots A_t A_j$ .

(R2) For each other pair  $(s', t')$  satisfying condition (R1) it holds that  $s' \leq s$  or  $t \leq t'$ .

We again refer to the example in Figure 5. Condition (R2) yields the following observation.

**Observation 2** For each value of  $s \in S = \{i + 1, \dots, j - 1\}$  the subpolygon  $P_{ij}$  has at most one representative  $(s, t_s)$ . Thus  $P_{ij}$  has at most  $|S| \in O(n)$  representatives.

Given the representatives of all  $P_{lj}$  with  $i < l < j$  we can compute  $cw_{ij}$  as

$$cw_{ij} = \min \left\{ \min_{\substack{i < l < j: \\ A_i A_l A_j \subset P}} (w_{il} + w_{lj} + 1), \min_l (w_{il} + cw_{lj}) \right\}$$

where the last minimum is over all  $i < l < j$  such that  $A_i A_l A_j \subset P$  and additionally  $P_{lj}$  possesses a representative satisfying condition (1). The information about the representatives of all  $P_{i_1 j}$  with  $i < i_1 < j$  suffices to determine the representatives of  $P_{ij}$ . As Keil and Snoeyink [KS02] we maintain a list of representatives for each  $P_{ij}$  sorted with respect to the first component which allows us to compute all values of type  $cw_{ij}$  in amortized  $O(n)$  time per pair  $(i, j)$ .

## 5 Analysis

We now investigate the time and space complexity of our algorithm. In order to compute the values of type  $pw_{i,j}$  we need access to all concave geodesics. The following slight modification of Theorem 2 in [GL07] yields a data structure that lets us compute and efficiently access the concave geodesics in a simple polygon. In order to motivate property (P3), observe that  $A_{\min(\pi)}$  and  $A_{\max(\pi)}$  are adjacent on a concave geodesic  $\pi$  if and only if  $\pi$  lies in the subpolygon  $P_{\min(\pi)\max(\pi)}$  and contains the diagonal  $d_{\min(\pi)\max(\pi)}$ .

**Proposition 1** *Given a simple polygon  $P = A_1A_2 \dots A_n$  we can construct in  $O(n^2)$  time and space a data structure  $DS$  with the following properties:*

(P1) *Given a pair  $(i, j)$ ,  $DS$  decides in  $O(1)$  time whether there is a concave geodesic  $\pi$  from  $A_i$  to  $A_j$ .*

(P2) *If  $\pi$  is a concave geodesic,  $DS$  provides the minimum index  $\min(\pi) = \{l \mid A_l \in \pi\}$  and the analogously defined maximum index  $\max(\pi)$  of  $\pi$  in  $O(1)$  time.*

(P3) *If  $\pi$  is a concave geodesic,  $DS$  decides in  $O(1)$  time whether  $A_{\min(\pi)}$  and  $A_{\max(\pi)}$  are adjacent on  $\pi$ .*

(P4) *If  $\pi$  is a concave geodesic of length  $l$ ,  $DS$  provides in  $O(l)$  time a walk along  $\pi$ .*

*Proof.* We first compute all lists  $\text{vis}(A_i)$  in  $O(n^2)$  total time. Then we use dynamic programming to check whether there is a concave geodesic  $\pi$  from  $A_i$  to  $A_j$ . If  $\pi$  exists, we also compute the second and the second last vertex on  $\pi$ . Having these vertices on each shorter geodesic, we can walk along  $\pi$  by repeatedly jumping to the second vertex of the remaining path, which by Remark 1 is also a geodesic.

We consider the pairs  $(i, j)$  in increasing order of the number of vertices on the path  $P^+(A_i, A_j)$ . The edges  $A_iA_{i+1}$  obviously correspond to concave geodesics and it is easy to determine the second and second last vertex of these paths.

When  $P^+(A_i, A_j)$  consists of more than one edge, we use the list  $\text{vis}(A_i)$  to find the last vertex  $A_l$  visible from  $A_i$  on  $P^+(A_i, A_j)$ . Observation 1 allows us to extract the desired information from  $\text{vis}(A_i)$  in  $O(1)$  amortized time. Then we query  $DS$  to see whether there is a concave geodesic  $\pi$  from  $A_l$  to  $A_j$ . (We can query  $DS$  with  $(l, j)$  since  $(l - j) \bmod n < (i - j) \bmod n$ .) If this is the case, we use the second and the second last vertex on  $\pi$  to check whether  $A_i$  can be added to  $\pi$  without violating property (G1). According to Remark 1 this is the only way for obtaining a concave geodesic  $\pi'$  from  $A_i$  to  $A_j$ . If  $A_j$  is visible from  $A_i$ , we can easily set  $\min(\pi')$  and  $\max(\pi')$ , and determine the second and the second last vertex on  $\pi'$ . Moreover  $A_{\min(\pi')}$  and  $A_{\max(\pi')}$  are adjacent in this case.

Consider the more interesting case that  $A_j$  is not visible from  $A_i$ . Then the second vertex on  $\pi'$  is  $A_l$  and the second last vertex on  $\pi'$  is the second last vertex on  $\pi$ . It is also clear that  $\min(\pi') = \min(i, \min(\pi))$  and  $\max(\pi') = \max(i, \max(\pi))$ . If  $\min(\pi') \neq i$  and  $\max(\pi') \neq i$ , then  $A_{\min(\pi')}$  and  $A_{\max(\pi')}$  are adjacent on  $\pi'$  if and only if  $A_{\min(\pi)}$  and  $A_{\max(\pi)}$  are adjacent on  $\pi$ , a piece of information that we have already computed. Otherwise, say if  $i = \min(\pi')$ , then  $A_{\min(\pi')}$  and  $A_{\max(\pi')}$  are adjacent on  $\pi'$  if and only if  $\max(\pi') = l$ . Thus the information we earlier computed for  $\pi$  yields the desired information for  $\pi'$  in  $O(1)$  time. It follows that we spend  $O(1)$  (amortized) time and  $O(1)$  space per pair  $(i, j)$ , and  $O(n^2)$  time and space in total.  $\square$

Now we are ready to prove our main theorem.

**Theorem 1** *A minimum PT-convex decomposition of a simple polygon  $P = A_1A_2 \dots A_n$  can be computed in  $O(n^3)$  time and space.*

*Proof.* Our algorithm for computing a minimum PT-convex decomposition is based on dynamic programming. We first detail how to compute the *number* of polygons in a minimum PT-convex decomposition rather than the decomposition itself.



We first set up the data structure  $DS$  of Proposition 1. Then for each subpolygon  $P_{ij}$  we compute a list  $L_{ij}$  consisting of all concave geodesics that are contained in  $P_{ij}$  and contain the diagonal  $d_{ij}$  as an edge. Recall that this is exactly the set of concave geodesics  $\pi$  with  $\min(\pi) = i$  and  $\max(\pi) = j$ . We store  $\pi = A_h, \dots, A_i, A_j, \dots, A_k$  simply as the pair  $(h, k)$  in  $L_{ij}$ . Note that each concave geodesic is contained in at most one list  $L_{ij}$ . We generate the lists of type  $L_{ij}$  as follows. We consider all pairs  $(i, j)$  in order of increasing value of  $j - i$  and query  $DS$  to see whether there is a concave geodesic  $\pi$  from  $A_i$  to  $A_j$ . We then use  $DS$  to determine whether  $A_{\min(\pi)}$  and  $A_{\max(\pi)}$  are adjacent on  $\pi$ . If yes, we insert the information provided by  $DS$  for the concave geodesic  $\pi$  in the list  $L_{\min(\pi)\max(\pi)}$ . Since all the queries to  $DS$  so far require constant time each, the construction of the lists  $L_{ij}$  takes  $O(n^2)$  time in total.

Then we implement the algorithm of Section 4. Using the technique of Keil and Snoeyink [KS02], we can compute all values of type  $cw_{ij}$  in  $O(n^3)$  time in total. It remains to bound the time needed for computing the values of type  $pw_{ij}$ . We check each concave geodesic  $\pi$  in  $L_{ij}$ . (Recall that the lists of type  $L_{ij}$  are pairwise disjoint.) We walk along  $\pi$  to determine the sum of the values  $w_{kl}$  over all  $(k, l) \neq (i, j)$  with  $A_k A_l \subseteq \pi$ . For each point  $A_l$  on  $P^+(B_m, B_1)$  we check whether there is a concave geodesic  $\pi_1$  from  $A_l$  to  $B_1$  and a concave geodesic  $\pi_2$  from  $B_m$  to  $A_l$ . If this is the case, then  $\pi\pi_1\pi_2$  is a pseudo-triangle according to Lemma 2. In order to compute a minimum decomposition containing  $\pi\pi_1\pi_2$ , we need the values  $w(\pi_1)$  and  $w(\pi_2)$ , which can be computed by walking along  $\pi_1$  and  $\pi_2$  the first time we need these values. In total, we walk along each geodesic a constant number of times. By Proposition 1, each walk takes  $O(n)$  time. According to Remark 1, the total number of concave geodesics is  $O(n^2)$ . This implies that we can determine all values of type  $pw_{ij}$  in  $O(n^3)$  time.

Thus the number of polygons in a minimum PT-convex decomposition of a simple polygon  $P$  with  $n$  vertices can be computed in  $O(n^3)$  time. The algorithm Keil and Snoeyink [KS02] requires  $O(n^3)$  space, while the data structure  $DS$  as well as the lists of type  $L_{ij}$  use  $O(n^2)$  space.

Finally we show how to adjust the above dynamic program so that it also yields a minimum pseudo-convex decomposition. Whenever in that program we compute a value of type  $pw_{ij}$  or  $cw_{ij}$ , we do some additional bookkeeping. From this extra information we can then backtrack and compute the decomposition that corresponds to the result of the dynamic program. This is a usual trick in dynamic programming. In the backtracking phase, we initially set  $i = 1$  and  $j = n$ , and then repeatedly check whether  $w_{ij} = pw_{ij}$  or  $w_{ij} = cw_{ij}$ . The following two paragraphs describe what we do in the first and in the second case, respectively.

Recall that when computing  $pw_{ij}$ , we determine the pseudo-triangle  $T$  that (a) lies in  $P_{ij}$ , (b) is adjacent to  $d_{ij}$ , and (c) yields the minimum decomposition of  $P_{ij}$  among all pseudo-triangles fulfilling (a) and (b). Now the above-mentioned extra bookkeeping consists of storing the indices of the three convex corners of  $T$ . When backtracking we draw the edges of  $T$  (with the help of  $DS$ ) and further decompose the subpolygons that constitute  $P_{ij} \setminus T$ .

When computing  $cw_{ij}$ , we store a copy of the list of representatives of  $P_{ij}$ . Note that this list has been built completely by the time that  $cw_{ij}$  is computed. Take any representative  $(s, t)$  and draw the corresponding diagonals  $d_{is}$  and  $d_{tj}$ . We consider two cases. If  $s = t$ , we simply have to find minimum pseudo-convex decompositions of  $P_{is}$  and of  $P_{sj}$ , see Figure 6a. Otherwise, if  $s \neq t$ , we decompose  $P_{is}$  into  $w_{is}$  polygons without any restrictions and decompose  $P_{sj}$  into  $cw_{sj}$  polygons including a convex polygon  $A_s \dots A_t A_j$ , see Figure 6b. When decomposing  $P_{sj}$  we must respect the diagonal  $d_{tj}$ . This is where we need the list of *all* representatives: by construction  $P_{sj}$  must have a representative  $(s', t)$ , and we can recursively decompose  $P_{s'j}$ .

Let us analyze time and space consumption of the modified dynamic program. The total time we spend for the backtracking part is the number of diagonals we draw plus the time we spend scanning the representative lists. The diagonals (including the edges of  $P$ ) are the edges of a plane graph with  $n$  vertices, thus their number is linear in  $n$ . Due to Observation 2 and Remark 1 the total number of representatives is  $O(n^3)$ . Storing (and scanning) these dominates the time and the space consumption of the modified dynamic program.  $\square$

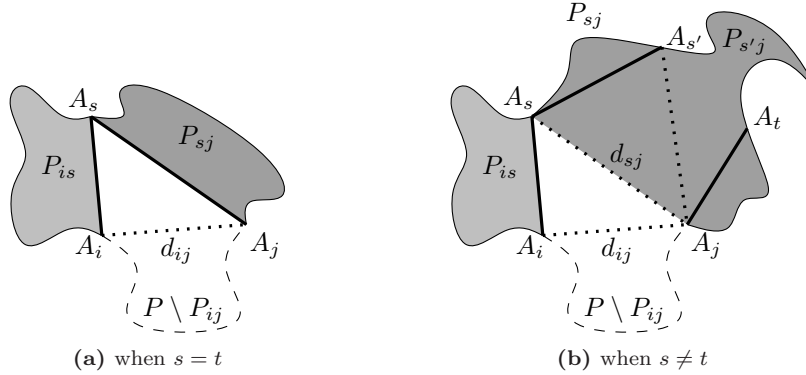


Fig. 6: Constructing a minimum decomposition of  $P_{ij}$ .

## 6 Open Problems

We have given an efficient and relatively simple dynamic program for computing minimum PT-convex decompositions of simple polygons. Can the running time of  $O(n^3)$  be improved, e.g., by making it not only depend on the number  $n$  of vertices, but also on, say, the number  $r$  of reflex vertices?

Can minimum PT-convex decompositions for *point sets* be computed efficiently? Is there a way to get at least a constant-factor approximation for that problem by decomposing the convex hull of the point set into simple polygons and then using our algorithm to further decompose the simple polygons? Note that grouping the subpolygons of a minimum decompositions arbitrarily into simple polygons and decomposing these simple polygons one after the other with our algorithm will yield a minimum decomposition, possibly different from the original. This shows that there are always decompositions of the point set into simple polygons that yield minimum PT-convex decompositions. The problem is just to guess a “right” decomposition. Note that Gudmundsson and Levcopoulos [GL07] use a similar strategy in their 15-approximation of the minimum-*weight* pseudo-triangulation.

Finally we ask whether PT-convex decompositions with other optimality criteria can be computed efficiently, e.g., minimum total edge weight. It seems that the method of representatives [KS02] will fail here since it is tailored towards minimizing the *number* of subpolygons.

One of the anonymous referees of this paper suggested to allow Steiner points to get even smaller PT-convex decompositions. This is an interesting variant of the problem. Recall that the algorithm of Chazelle and Dobkin [CD85] that computes minimum convex decompositions with Steiner points is even faster than the fastest known algorithm for the corresponding problem without Steiner points [KS02].

## Acknowledgments

We thank Bettina Speckmann for pointing us to reference [GL07] and the anonymous referees for their very helpful and precise comments. We also thank Leonidas Palios for continuous support during his editorship of this article.

## References

- [AHK<sup>+</sup>06] Oswin Aichholzer, Clemens Huemer, Sarah Kappes, Bettina Speckmann, and Csaba D. Tóth. Decompositions, partitions, and coverings with convex polygons and pseudo-triangles. In Rastislav Kráľovič and Pawel Urzyczyn, editors, *Proc. 31st Intern. Sym.*

*pos. Math. Foundat. Comput. Sci. (MFCS'06)*, volume 4162 of *Lecture Notes Comput. Sci.*, pages 86–97. Springer-Verlag, 2006. 2

- [ARSS03] Oswin Aichholzer, Günter Rote, Bettina Speckmann, and Ileana Streinu. The zigzag path of a pseudo-triangulation. In Frank Dehne, Jörg-Rüdiger Sack, and Michiel Smid, editors, *Proc. 8th Workshop on Algorithms and Data Structures (WADS'03)*, volume 2748 of *Lecture Notes Comput. Sci.*, pages 377–388. Springer-Verlag, 2003. 2
- [Ber05] Sergey Bereg. Enumerating pseudo-triangulations in the plane. *Comput. Geom. Theory Appl.*, 30(3):207–222, 2005. 2
- [CD85] Bernard Chazelle and David P. Dobkin. Optimal convex decompositions. In G. T. Toussaint, editor, *Computational Geometry*, pages 63–133. North Holland, Amsterdam, 1985. 2, 10
- [CEG<sup>+</sup>94] Bernard Chazelle, Herbert Edelsbrunner, Michelangelo Grigni, Leonidas J. Guibas, John Hershberger, Micha Sharir, and Jack Snoeyink. Ray shooting in polygons using geodesic triangulations. *Algorithmica*, 12(1):54–68, 1994. 1
- [FMR01] Thomas Fevens, Henk Meijer, and David Rappaport. Minimum convex partition of a constrained point set. *Discrete Applied Mathematics*, 109(1–2):95–107, 2001. 1
- [GL07] Joachim Gudmundsson and Christos Levcopoulos. Minimum weight pseudo-triangulations. *Comput. Geom. Theory Appl.*, 38(3):139–153, 2007. 2, 8, 10
- [GN06] Jesús García-López and Carlos M. Nicolás. Planar point sets with large minimum convex partitions. In *Proc. 22nd European Workshop Comput. Geom. (EWCG'06)*, pages 51–54, Delphi, 2006. 2
- [GT97] Michael T. Goodrich and Roberto Tamassia. Dynamic ray shooting and shortest paths in planar subdivisions via balanced geodesic triangulations. *J. Algorithms*, 23(1):51–73, 1997. 1
- [Kei85] J. Mark Keil. Decomposing a polygon into simpler components. *SIAM J. Comput.*, 14:799–817, 1985. 2, 3
- [KL98] Drago Krznaric and Christos Levcopoulos. Quasi-greedy triangulations approximating the minimum weight triangulation. *J. Algorithms*, 27(2):303–338, 1998. 1
- [KS02] J. Mark Keil and Jack Snoeyink. On the time bound for convex decomposition of simple polygons. *Internat. J. Comput. Geom. Appl.*, 12(3):181–192, 2002. 2, 3, 4, 5, 6, 7, 9, 10
- [KS06] Christian Knauer and Andreas Spillner. Approximation algorithms for the minimum convex partition problem. In *Proc. 10th Scandinavian Workshop Algor. Theo. (SWAT'06)*, volume 4059 of *Lecture Notes Comput. Sci.*, pages 232–241. Springer-Verlag, 2006. 2
- [KSS02] David G. Kirkpatrick, Jack Snoeyink, and Bettina Speckmann. Kinetic collision detection for simple polygons. *Internat. J. Comput. Geom. Appl.*, 12(1-2):3–27, 2002. 1
- [MR06] Wolfgang Mulzer and Günter Rote. Minimum weight triangulation is NP-hard. In *Proc. 22nd Annu. ACM Sympos. Comput. Geom. (SoCG'06)*, pages 1–10, Sedona, Arizona, 2006. 1
- [PV96] Michel Pocchiola and Gert Vegter. Topologically sweeping visibility complexes via pseudo-triangulations. *Discrete Comput. Geom.*, 16:419453, 1996. 1

- [Spi05] Andreas Spillner. Optimal convex partitions of point sets with few inner points. In *Proc. 17th Canadian Conf. Comput. Geom. (CCCG'05)*, pages 34–37, Windsor, 2005. 2
- [ST05] Bettina Speckmann and Csaba D. Tóth. Allocating vertex  $\pi$ -guards in simple polygons via pseudo-triangulations. *Discrete Comput. Geom.*, 33(2):345–364, 2005. 1
- [Str00] Ileana Streinu. A combinatorial approach to planar non-colliding robot arm motion planning. In *Proc. 41st Annu. IEEE Sympos. Foundat. Comput. Sci. (FOCS'00)*, pages 443–453, 2000. 1, 2