

## Tiling Polygons with Parallelograms\*

Sampath Kannan<sup>1</sup> and Danny Soroker<sup>2</sup>

<sup>1</sup> University of Arizona, Tucson, AZ 85721, USA

<sup>2</sup> Shell Development Company, P.O. Box 481,  
Houston, TX 77001-0481, USA

**Abstract.** Under what conditions can a simple polygon be tiled by parallelograms? In this paper we give matching necessary and sufficient conditions on the polygon to be tilable and characterize the set of possible tilings. We also provide an efficient algorithm for constructing a tiling.

### 1. Introduction

Given a polygon,  $P$ , in the Euclidean plane, we are interested in tiling  $P$  by parallelograms, i.e., partitioning it into parts, each of which is a parallelogram. The allowable set of tiles is the entire class of parallelograms. A particular tiling may use several noncongruent parallelograms. The polygons we consider are simple (no self-intersection) and without holes.

The current tiling literature deals mostly with tiling of infinite regions (see, e.g., [10]). Most of the papers on tiling finite regions deal with situations where the decision problem is trivial (a tiling always exists), and the issue is to find a tiling which is optimal in some sense [13] (best running time, least number of tiles, etc.). Examples are: tiling rectilinear polygons with rectangles [12], tiling polygons with trapezoids [1], and triangulating polygons [2], [14].

Thurston [15] uses combinatorial group theory to handle problems of tiling polygonal regions with simple shapes such as dominos and hexominos. Kenyon [11] has independently considered the problem of tiling polygonal regions with

---

\* This work was done while S. Kannan was with the University of California, Berkeley, and was supported by NSF Grant CCR 88-13632. The work of D. Soroker was done while he was with IBM Almaden Research Center.

parallelograms and shown that conditions similar to ours are necessary for a tiling to exist. He has also proved that this problem for polygons with holes is NP-hard.

Our solution is complete and quite simple. We use several tools from combinatorial optimization and computational geometry to obtain necessary and sufficient conditions and derive an efficient tiling algorithm. Our algorithm runs in time  $O(n^2)$  for an  $n$ -vertex polygon. This is optimal for certain classes of polygons (such as convex polygons), for which the number of tiles required is proportional to  $n^2$ .

In what follows, unless otherwise stated, *tiling a polygon* will mean tiling it with parallelograms. In the next section we describe a structure which underlies any tiling, and is central to our intuition about the tiling problem. Then we present a simple necessary and sufficient condition for tilability of convex polygons. In Section 4 we introduce an additional condition which is required for tilability of general polygons. In Section 5 we prove the main theorem, namely, the adequacy of these two conditions. In Section 6 we describe our tiling algorithm and give a finer characterization of the set of possible tilings.

## 2. The Underlying Structure of a Tiling

In this section we describe a basic structural property of parallelogram tilings. For tilings  $\mathcal{T}$  and  $\mathcal{R}$  of a polygon,  $P$ , we say that  $\mathcal{R}$  is a *refinement* of  $\mathcal{T}$  if  $\mathcal{R}$  is obtained from  $\mathcal{T}$  by further partitioning some of its tiles. Two tiles are *adjacent* if an edge of one overlaps with an edge of the other. They *share an edge* if their overlapping edges are of equal length. Now, any two adjacent parallelograms can be refined (each partitioned into at most three tiles) so that in the result, the two tiles that are adjacent share an edge (Fig. 1(a)). By iterating this operation (Fig. 1(b)), we obtain a refinement, whose structure is described in Lemma 1 below.

A tiling *partitions the boundary* of the polygon into (edge) segments, where each segment is the intersection of the boundary of the polygon with an edge of a tile. A *path of parallelograms* (Fig. 2) is a sequence  $p_1, p_2, \dots, p_k$  for which  $p_i$  shares an edge with  $p_{i+1}$  for all  $i$ , and all the edges shared along the path are parallel to each other. The *path connects edges*  $e_1$  and  $e_k$ , which are the edges of  $p_1$  and  $p_k$  respectively, that are not shared by other tiles and are parallel to the shared edges of the path.

**Lemma 1** (Structure Lemma). *Any tilable polygon,  $P$ , admits a tiling,  $\mathcal{R}$ , which has the following properties:*

1. *Every two adjacent tiles in  $\mathcal{R}$  share an edge.*
2.  *$\mathcal{R}$  partitions the boundary of  $P$  into segments such that each segment is connected to another segment by a path of parallelograms in  $\mathcal{R}$ .*

*Proof.* By the discussion above, any tiling of  $P$  can be refined into a tiling,  $\mathcal{R}$ , with these two properties. Thus  $\mathcal{R}$  is also a tiling of  $P$ .  $\square$

A *matching* (on a polygon,  $P$ ) is a partition of the boundary of  $P$  into segments together with a pairing of these segments such that each segment is paired with

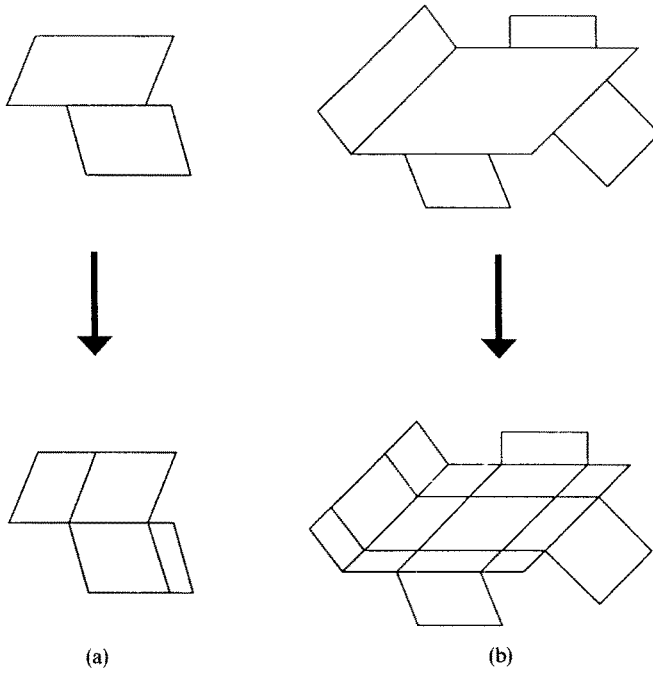


Fig. 1. (a) Refining adjacent tiles. (b) Refining a whole tiling.

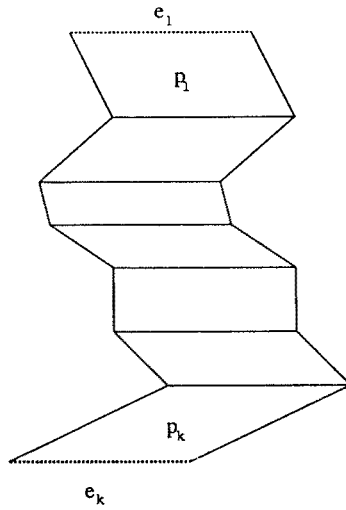


Fig. 2. A path of parallelograms.

an antiparallel segment of equal length. Note that the tiling of Lemma 1 naturally induces a matching, where two segments are matched (paired) if they are connected by a path of parallelograms. In this paper we concentrate on such tilings.

### 3. The Balance Condition and Convex Polygons

In this section we present the *balance condition*, and show that it gives an exact characterization of tilable convex polygons. We start with several definitions.

It is convenient to view  $P$ 's boundary as oriented clockwise. With respect to this orientation we speak of  $tail(e)$  and  $head(e)$  of an edge  $e$  of the polygon. The edges of the polygon can be partitioned into equivalence classes where two edges are in the same class if they are parallel. We call the equivalence classes *slope classes*. Two edges in the same slope class are *opposite* (or *antiparallel*) if their orientations are  $180^\circ$  apart.

**Balance Condition.** In every slope class,  $S$ , the vector sum of all the (oriented) edges in  $S$  is zero.

**Lemma 2.** *A polygon,  $P$ , is tilable only if it meets the balance condition.*

*Proof.* If two boundary segments are matched, they must be opposite and of equal length. Thus this lemma follows from Lemma 1.  $\square$

For a convex polygon, each slope class contains at most two edges. Therefore, in order to meet the balance condition, each class must contain exactly two equal-length edges. This turns out to be sufficient for tilability.

**Theorem 1.** *A convex polygon,  $P$ , is tilable if and only if it respects the balance condition.*

*Proof.* By Lemma 2, the condition is necessary. To prove sufficiency, we construct a tiling of  $P$  inductively in the number of slope classes of  $P$ . As noted, each slope class contains two equal-length edges, so if  $P$  has two slope classes, it is a parallelogram. Figure 3 illustrates the induction step below. Let  $\{A, A'\}$  be some slope class of  $P$ . Let  $C$  be the part of  $P$ 's boundary from  $head(A')$  to  $tail(A)$  and let  $C'$  be the piecewise-linear curve parallel to  $C$  from  $tail(A')$  to  $head(A)$ . The region bounded by  $C$  and  $C'$  can clearly be tiled with a path of parallelograms. The remaining part of  $P$  is a convex polygon satisfying the balance condition with one fewer slope class (the pair  $\{A, A'\}$  is gone), and is, therefore, tilable by the inductive hypothesis.  $\square$

We note that this tiling minimizes the number of tiles;  $\binom{n/2}{2}$  for an  $n$ -sided convex polygon. We also point out that a convex polygon is tilable if and only if it is a zonotope. The reader is referred to [6] for a discussion of zonotopes.

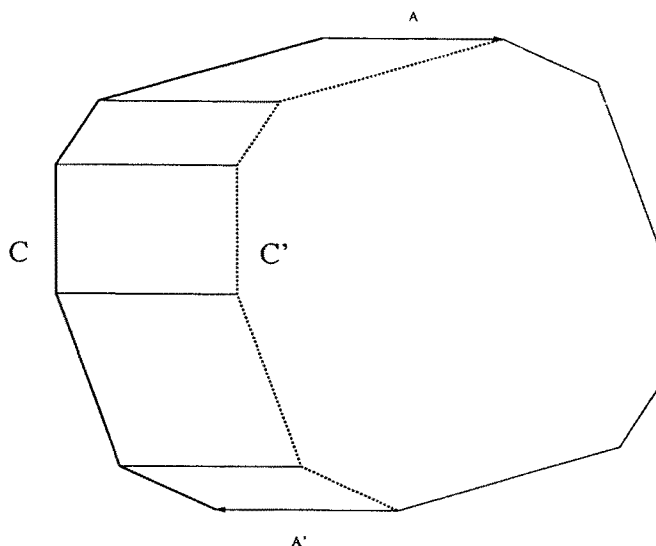


Fig. 3. Tiling a convex polygon.

#### 4. The Convex Crossing Condition

In the following sections, until explicitly stated, we consider only polygons that admit a tiling for which the corresponding boundary segments are whole edges. Furthermore, we deal only with such tilings.

The balance condition is a necessary condition for tiling, which is also sufficient for convex polygons. However, it is not sufficient for general polygons. A simple example is shown in Fig. 4. This polygon has a “waist,” i.e., a place where it is too narrow to permit matching of the horizontal pair of edges. However, the absence of a waist is not sufficient for tilability. An example is shown in Fig. 5 [4].

Two pairs of edges  $\{A, A'\}$  and  $\{B, B'\}$  cross if we encounter exactly one of  $B$  and  $B'$  in the clockwise path from  $A$  to  $A'$ .

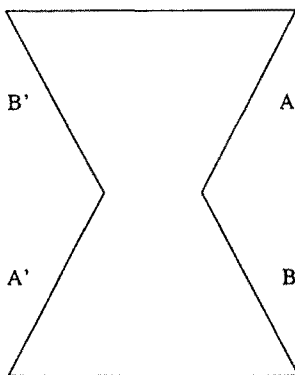


Fig. 4. Untilable balanced hourglass.

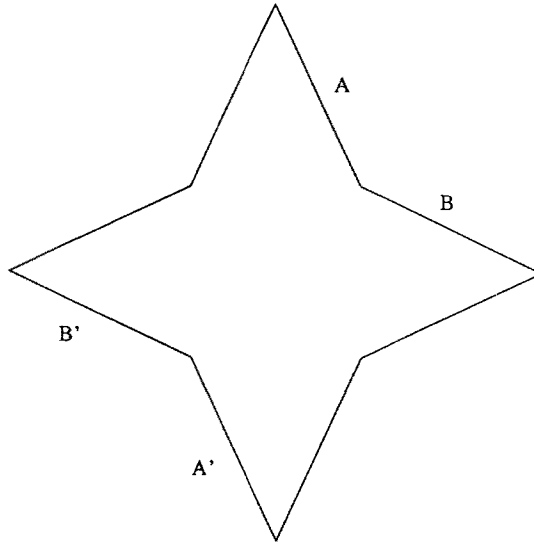


Fig. 5. Untilable waistless polygon.

**Lemma 3.** *If two pairs cross, then in any tiling in which they are, respectively, matched, the connecting path of one pair intersects the connecting path of the other.*

*Proof.* Since any connecting path is a continuous region in the plane. □

Assume, without loss of generality, that the sequence of edges encountered clockwise is  $A, B, A', B'$ . Attach  $head(A)$  to  $tail(B)$  and consider the angle subtended to the right of  $A$  (i.e., the angle on the inside of  $P$ ). If this angle is no more than  $180^\circ$ , then we say that the pairs  $\{A, A'\}$  and  $\{B, B'\}$  have a *convex crossing*.

**Convex crossing condition** (for a matching). Every two matched pairs that cross have a convex crossing.

A polygon satisfies the *convex crossing condition* if it admits some matching that does.

**Lemma 4.** *A polygon,  $P$ , is tilable only if it satisfies the convex crossing condition.*

*Proof.* By Lemma 3, if two matched pairs cross, then their connecting paths intersect. This intersection is well-defined (i.e., is a parallelogram) exactly when the two pairs have a convex crossing. □

Now note that the polygons in Figs. 4 and 5 both have pairs  $\{A, A'\}$  and  $\{B, B'\}$  that violate the convex crossing condition.

## 5. The Main Theorem

**Theorem 2.** *A polygon,  $P$ , is tilable if and only if it satisfies the balance and convex crossing conditions.*

*Proof.* Both conditions have been proven necessary. As stated earlier, we assume that for the matching which has the convex crossing property, the corresponding boundary segments are the edges of  $P$ . Following the proof we show how to relax this assumption.

By *collapsing a pair* of the polygon, we mean removing a matched pair of edges from the polygon and connecting the two remaining boundary segments. This can be visualized as continuously shrinking the two edges until they are both of length zero. Both the balance and the convex crossing conditions are invariant under this operation. We call a polygon for which no two edges cross *simple*.

**Note.** Under this definition, in a simple polygon a vertex or edge may overlap with another vertex or edge.

*Adding back a pair* is the opposite of collapsing the pair. We call a pair of edges  $\{A, A'\}$  *minimal* if the clockwise boundary path from  $A$  to  $A'$  contains no pairs.

The outline of the proof is as follows. We create a sequence of polygons:

$$P_2 \subset P_3 \subset \cdots \subset P_{n/2} = P,$$

where, for  $2 \leq i < n/2$ ,  $P_i$  has  $i$  pairs, and is obtained from  $P_{i+1}$  by collapsing a minimal pair.

Now the proof proceeds by induction, by showing that:

**Claim 1.**  $P_2$  is simple and tilable.

**Claim 2.** If  $P_i$  is simple, then it is tilable.

**Claim 3.** If  $P_i$  is not simple, then neither is  $P_{i+1}$ .

Thus, since the original polygon ( $P_{n/2}$ ) is simple, it follows that it is tilable.

We now proceed to prove the above claims.

*Proof of Claim 1.* Recall that  $P_2$  respects the balance condition. If the sequence of edges is  $A, B, A', B'$ , then it is a parallelogram, and thus is simple and tilable. If the sequence of edges is  $A, A', B, B'$ , then it is two connected line segments, and thus is simple and trivially tilable (since it has no area).  $\square$

*Proof of Claim 2.* Let  $\{A_i, A'_i\}$  be the pair added back to  $P_{i-1}$  to obtain  $P_i$ . Without loss of generality, assume the pair is horizontal with  $A_i$  oriented right and  $A'_i$  oriented left. Since this pair is, by assumption, minimal in  $P_i$ , the boundary from  $A_i$  to  $A'_i$  (clockwise) contains no pairs. We call this the *right boundary*. It

follows from the convex crossing condition that all the edges in the right boundary must be oriented downward (not necessarily vertically). A consequence of this is that  $P_i$  is the union of  $P_{i-1}$  and a path of parallelograms connecting  $A_i$  and  $A'_i$  (along their right boundary). If  $P_i$  is simple, then this union is a disjoint union, and the tilability of  $P_i$  is a consequence of the tilability of  $P_{i-1}$ .  $\square$

*Proof of Claim 3.* First we claim that if  $P_i$  is not simple, then there is a point on the plane,  $x$ , such that the boundary of the polygon wraps around it more than once in the clockwise direction. This follows from the above observation that  $P_i$  is the union of two simple polygons ( $P_{i-1}$  and a path of parallelograms).

We call the number of times the polygon wraps around a point  $x$  in the clockwise direction, the *winding number* of  $x$ . If  $x$  has a winding number greater than 1, the *inner loop* (around  $x$ ) is the simple, closed curve that most immediately encircles  $x$  in the clockwise direction and is made up of sections of the boundary of the polygon. The *outer loop* (around  $x$ ) is the simple, closed curve made up of sections of the boundary that encircles  $x$  in the clockwise direction and is not encircled by any other sections of the boundary.

The last part of the theorem states that despite the addition of more pairs there will always be a point with a winding number more than 1. As above, let  $A_{i+1}$ ,  $A'_{i+1}$  be the pair added at stage  $i+1$ , and let  $C_{i+1}$  be the clockwise path along the boundary from  $A_{i+1}$  to  $A'_{i+1}$ . Let  $x$  be the point with the winding number greater than 1 before the addition of  $A_{i+1}$ ,  $A'_{i+1}$ . We distinguish between two cases:

*Case 1.*  $C_{i+1}$  contains some edge of the inner loop around  $x$ . Let  $e$  be such an edge. We choose  $x'$  to be infinitesimally close to  $e$  and to the right of  $e$  (with respect to the direction  $e$  is pointing) in the polygon  $P_i$ . Note that  $P_i$  has a winding number greater than 1 around  $x'$ .

Treating the boundaries of  $P_i$  and  $P_{i+1}$  as curves, we can show that the curve  $P_i$  can be continuously deformed into the curve  $P_{i+1}$  in such a way that, in the process of deformation, no intermediate curve passes through the point  $x'$ . Homotopy theory [9] tells us that this implies that the winding numbers of  $P_i$  and  $P_{i+1}$  around  $x'$  are the same.

*Case 2.*  $C_{i+1}$  does not contain any edge of the inner loop around  $x$ . In this case the inner loop around  $x$  remains intact. Consider the boundary of the polygon,  $P_i$ , with the sections forming the inner loop deleted. This is still a closed curve wrapping around  $x$  at least once in the clockwise direction. We claim that the corresponding sections of the boundary of  $P_{i+1}$  wrap around  $x$  at least once. We distinguish two subcases.

*Case 2a.*  $x$  is not covered by the path of parallelograms of the pair  $\{A_{i+1}, A'_{i+1}\}$ . The outer loop around  $x$  in  $P_i$  can be continuously deformed to the outer loop around  $x$  in  $P_{i+1}$  and hence the outer loop still has a winding number at least 1 around  $x$  and the entire boundary has a winding number at least 2.

*Case 2b.*  $x$  is covered by the path of parallelograms of the pair  $\{A_{i+1}, A'_{i+1}\}$ . By the observation that  $P_{i+1}$  is the union of  $P_i$  and the path of parallelograms, we find



that the portion of the boundary of  $P_{i+1}$ , excluding the inner loop, wraps around  $x$  at least once. Hence the entire boundary of  $P_{i+1}$  wraps around  $x$  more than once.  $\square$

### 5.1. Remarks

First we explain how the proof of the main theorem works without the assumption that the boundary segments corresponding to the tiling are whole edges. This is easy—assume there is a tiling that partitions the boundary into arbitrary segments; simply introduce dummy vertices at the ends of all segments, and carry out the proof on this “new” polygon. Consequently, proof of tilability of this new polygon translates immediately to a proof for the original one.

The second point we should mention is that the proof is constructive. If we are given a matching which satisfies the convex crossing condition, then our proof implies the following simple tiling algorithm: repeatedly find a minimal pair and tile the path connecting it along the part of the boundary that contains no pairs. A minimal pair can be found in  $O(n)$  time, so this tiling procedure runs in time  $O(n^2)$ . We use it as the final step of our algorithm, which is described in the next section.

## 6. The Algorithm

In this section we describe our algorithm that determines if a polygon is tilable by parallelograms, and, if so, produces such a tiling. The algorithm must first construct a feasible matching between the boundary edges of the polygon and then produce a tiling that corresponds to this matching. The algorithm runs in time  $O(n^2)$ , where  $n$  is the number of vertices of the polygon. Note that this algorithm is optimal when the number of tiles required is  $\Theta(n^2)$  (as in the case of convex polygons). For simplifying the discussion here, we assume that the numbers (in the input and throughout the computation) are real, and that arithmetic operations (such as addition, square root) are performed in unit time. The input is a sequence of points (each with  $X$  and  $Y$  coordinates) which are the vertices of the polygon. The output, if the polygon is tilable, is a list of the connecting paths of all matched pairs. Furthermore, when an edge is matched with several others, the lengths of all its pieces are specified.

Before we plunge into the algorithm, we describe an important theorem that states that, up to trivial modifications, the matching of polygon edges is unique. More specifically, we say that a matching is *degenerate* if it contains two pairs from the same slope class that cross. This is because the parallelogram at which they cross is degenerate—a line segment. An example is shown in Fig. 6.

**Lemma 5.** *Any tilable polygon admits a tiling for which the underlying matching is nondegenerate.*

*Proof.* Figure 6 demonstrates a local transformation that removes a degenerate crossing. Note that this transformation may increase the number of tiles. This can be performed independently for each such crossing, and the lemma follows.  $\square$

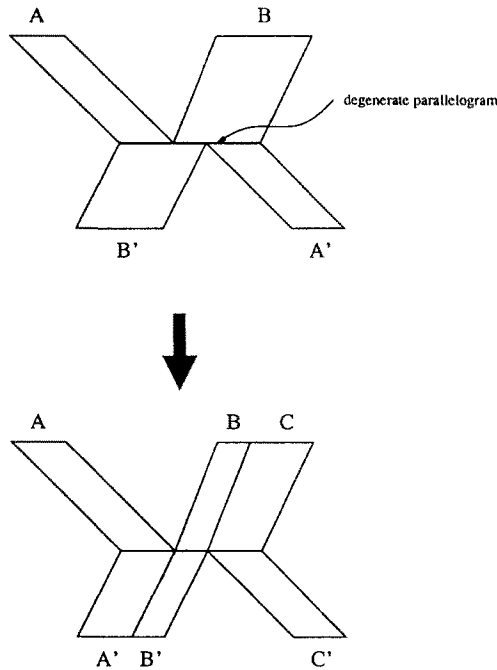


Fig. 6. Transformation for removing a degenerate crossing.

**Theorem 3.** *Any tilable polygon has a unique matching which is nondegenerate.*

The proof is given below. Now we proceed to describe the algorithm. It has the following six main steps:

1. Compute slope classes.
2. Test balance condition.
3. Compute visibility relation (explained below).
4. Compute matching.
5. Test convex crossing condition.
6. Produce output.

A partition into slope classes can be obtained by sorting the edges by slope in  $O(n \log n)$  time. The balance condition can then be tested in  $O(n)$  time by traversing the sorted list once.

We say that two opposite edges of a slope class are *visible* (to each other) if they can be connected by a curve internal to the polygon, which is monotone in the direction perpendicular to the two edges (i.e., starting from one edge, you can continuously “move forward” inside the polygon until you reach the other edge). Note that this condition is equivalent to having the possibility of having a connecting path of parallelograms between them. Thus a feasible matching can match only visible pairs. Since a matching that satisfies the convex crossing condition is a feasible matching, it must match only visible pairs.

The visibility relation is computed for each slope class separately. Once the slope class is fixed, the notion of visibility can be defined between any pair of points of the polygon. We describe the computation for one slope class, which is assumed to be horizontal. (Any slope class can be made horizontal in  $O(n)$  time by applying a suitable linear transformation to the vertices of the polygon.) We thus refer to the two types of edges in the slope class as *floors* and *ceilings*. The idea for computing visibility is the following: the polygon is partitioned into a small number of *visibility regions*, where two points are in the same region if the same set of points is visible to both of them. (This partition turns out to be similar to the “monotone decomposition” that has appeared in the literature [5], [14].) A *visibility digraph*, whose nodes correspond to the visibility regions, is constructed. In this digraph there is a directed path between two nodes if and only if the corresponding regions are mutually visible. It is used later to compute the matching.

The visibility regions are determined by “dents” in the  $Y$  direction. This is best explained by a picture, and is shown in Fig. 7(a). A dent is either a vertex which is a local minimum or maximum for the  $Y$ -coordinate or a floor or ceiling. Each dent breaks the polygon into two or three regions, and at least one of these regions is incident upon another dent. Thus the total number of visibility regions is at most  $2d + 1$ , where  $d$  is the number of dents. The number can be smaller when several dents have the same  $Y$  value.

The visibility digraph (Fig. 7(b)) is defined as follows: it has a node corresponding to each visibility region, and there is an arc from node  $v$  to node  $w$  if region  $v$  touches region  $w$  and is above it. Thus directed paths represent visibility in precisely the manner we want.

The visibility digraph can be constructed in  $O(n)$  time per slope class given a triangulation of  $P$  with a few additional linear-time operations [2], [3], [8], [14]. Details are omitted here.

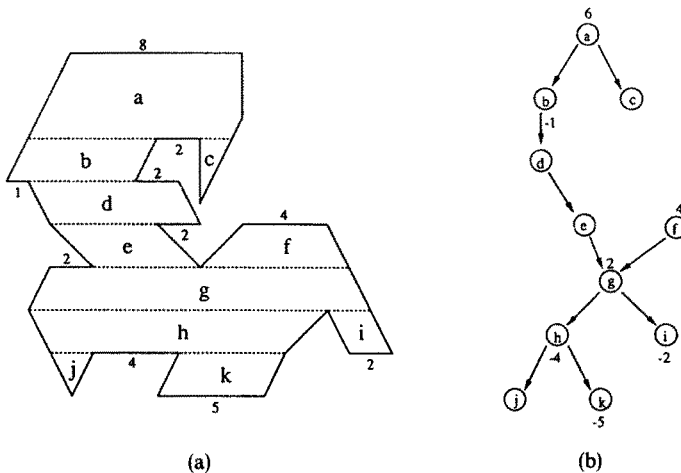


Fig. 7. (a) Polygon with visibility regions and lengths indicated. (b) Visibility digraph with supplies (positive) and demands (negative).

The matching is computed by performing a *supply–demand* [7] computation on the visibility digraph with the following parameters. All arcs have infinite capacity. The supply/demand of a node is the algebraic sum of the lengths of ceilings and floors contained in the region it corresponds to, as shown in Fig. 7. In the tiling the supply/demand of a node corresponds to the total width of connecting paths that must leave/enter the corresponding visibility region. Note that, in general, an edge may be matched with several others. The following observations are crucial for obtaining an efficient algorithm and for proving Theorem 3.

**Lemma 6.** *The visibility digraph has  $O(n)$  nodes and is an oriented tree (i.e., its underlying undirected graph is a tree).*

*Proof.* As noted above, the number of nodes is at most  $2d + 1$ , where  $d$  is the number of vertices at local minimum or maximum. The second fact follows from the assumptions that the polygon has no holes and is connected.  $\square$

**Lemma 7.** *A supply–demand problem on an oriented tree has at most one solution.*

*Proof.* Each arc constitutes a cut. The flow crossing a cut is determined by the supplies and demands on both sides. Therefore, the flow on each arc is uniquely determined.  $\square$

*Proof of Theorem 3.* The lemmas above show that if the visibility digraph admits a flow pattern that matches ceilings to floors, then it is unique. A flow pattern corresponds to a unique nondegenerate matching. The theorem follows.  $\square$

The supply–demand problem we defined can be solved in linear time by repeatedly pruning a leaf from the tree. When a leaf is pruned, the value of its neighbor is updated by adding the leaf’s net supply (or subtracting its net demand), and the flow on the arc connecting to the leaf is equal to its net supply (demand). If a supply leaf has an incoming arc or a demand leaf has an outgoing arc, then there is no solution (the polygon is untilable). After computing the flows, a simple computation (splitting flows at each node and tracing paths) gives the unique, nondegenerate matching and the lengths of segments into which the edges are split. The resulting flow paths and matching for Fig. 7 are shown in Fig. 8. This step takes time  $O(nk)$ , where  $k$  is the number of ceilings. Thus it takes  $O(n^2)$  for the whole polygon. We note here that even though an edge can be matched with several others, there are less than  $n$  pairs. This follows from the fact that the pairing relation (when described as a graph) contains no cycles.

After the matching is determined, the convex crossing condition is checked. This can be done in  $O(n^2)$  time by traversing the sequence of edges while referring to the list of matches (each edge points to its mates). Finally the output can be produced by the construction in the main proof, as explained in the previous section.

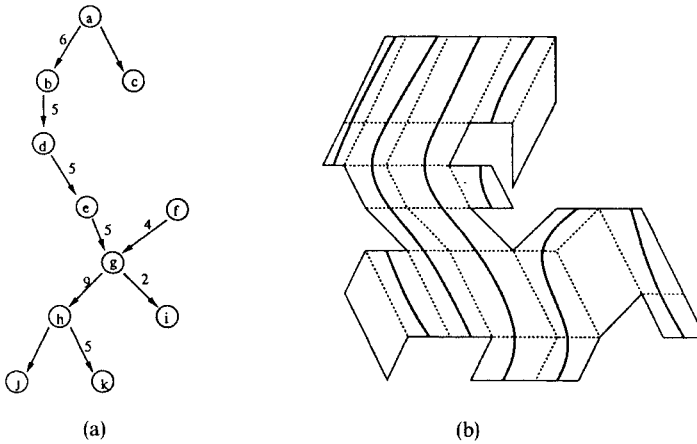


Fig. 8. (a) The folows on the visibility digraph of Fig. 7. (b) A tiling of the polygon of Fig. 7 with the matching of horizontal segments indicated.

**Acknowledgments**

We wish to thank Tsahi Birk, Richard Cleve, Steve Fortune, Branko Grünbaum, and Chris Van Wyk for helpful discussions and suggestions. We would also like to acknowledge the referees' valuable comments.

**References**

1. T. Asano and T. Asano, Minimum Partition of Polygonal Regions into Trapezoids, *IEEE Symp. on Foundations of Computer Science*, 1983, pp. 233–241.
2. B. Chazelle, Triangulating a Simple Polygon in Linear Time, *IEEE Symp. on Foundations of Computer Science*, 1990, pp. 220–230.
3. B. Chazelle and J. Incerpi, Triangulation and Shape-Complexity, *ACM Trans. Graphics*, vol. 3, no. 2, 1984, pp. 135–152.
4. R. Cleve, Private communication.
5. D. P. Dobkin, D. L. Souvaine, and C. J. Van Wyk, Decomposition and Intersection of Simple Splines, *Algorithmica*, vol. 3, 1988, pp. 473–485.
6. H. Edelsbrunner, *Algorithms in Combinatorial Geometry*, Springer-Verlag, Heidelberg, 1987.
7. L. R. Ford, Jr., and D. R. Fulkerson, *Flows in Networks*, Princeton University Press, Princeton, NJ, 1962.
8. A. Fournier and D. Y. Montuno, Triangulating Simple Polygons and Equivalent Problems, *ACM Trans. Graphics*, vol. 3, no. 2, 1984, pp. 153–174.
9. M. J. Greenberg and J. R. Harper, *Algebraic Topology; A First Course*, Addison-Wesley, Reading, MA, 1981.
10. B. Grünbaum and G. C. Shephard, *Tiling and Patterns*, Freeman, New York, 1986.
11. R. Kenyon, Tiling a Polygon with Parallelograms, Manuscript.
12. T. Ohtsuki, Minimum Partition of Rectilinear Regions, *IEEE Internat. Symp. on Circuits and Systems*, 1982, pp. 1210–1213.

13. J. O'Rourke, *Art Gallery Theorems and Algorithms*, Oxford University Press, Oxford, 1987.
14. R. E. Tarjan and C. J. Van Wyk, An  $O(n \log \log n)$ -Time Algorithm for Triangulating a Simple Polygon, *SIAM J. Comput.*, vol. 17, no. 1, 1988, pp. 143–178.
15. W. P. Thurston, Conway's Tiling Groups, *Amer. Math. Monthly*, vol. 97, no. 8, 1990, pp. 757–773.

*Received February 20, 1990, and in revised form February 4, 1991.*