7

VISIBILITY GRAPHS

7.1. INTRODUCTION

It is my belief that some of the fundamental unsolved problems involving visibility in computational geometry will not be solved until the combinatorial structure of visibility is more fully understood. Perhaps the purest condensation of this structure is a visibility graph. The nodes of a visibility graph correspond to geometric components, such as vertices or edges, and two nodes are connected by an arc of the graph if the components can "see" one another, perhaps under some restricted form of visibility. The canonical example is the vertex visibility graph of a polygon: its nodes correspond to the vertices of a polygon, and its arcs to lines of visibility between vertices in the interior or along the boundary of the polygon. No characterization of these graphs is available, and in fact almost no general properties are known. ElGindy has pioneered their investigation in his thesis (ElGindy 1985). He obtained a specialized result by restricting the class of graphs to maximal outerplanar graphs. This result is presented in Section 7.2. Although this result is very restricted, it is the most general obtained to date.

Section 7.3 explores a different type of visibility graph: the nodes correspond to edges of an orthogonal polygon, and two nodes are connected by an arc if their edges can see one another along a line orthogonal to the edges. These graphs are especially simple for polygons in "general position": then there are always precisely two components, both of which are trees. A partial characterization of tree pairs is presented in Section 7.3.

Finally, in Section 7.4, visibility graphs for a set of disconnected vertical line segments are studied. Here a node is associated with each segment, and arcs correspond to horizontal visibility. In this case a very pleasing characterization theorem has been obtained by Wismath (1985), and independently by Tamassia and Tollis (1985). We present Wismath's proof.

The investigation of visibility graphs is in its infancy, and new results can be expected not only for the graphs discussed in this chapter, but also for other types of visibility graphs.
7.2. VERTEX VISIBILITY GRAPHS

As just mentioned, two nodes are adjacent in the vertex visibility graph (or just "visibility graph" when no confusion is possible) of a polygon if and only if the line segment determined by the associated vertices is at no point exterior to the polygon. Thus the connections represent lines of sight between vertices. For a polygon of \( n \) vertices, the vertex visibility graph can have as many as \( \binom{n}{2} \) arcs, when the polygon is convex and the graph is the complete graph \( K_n \) (Fig. 7.1a), or as few as \( 2n - 3 \) arcs, when the polygon has \( n - 3 \) reflex vertices (Fig. 7.1b). Although visibility graphs seem "abundant," not every graph of \( e \) edges with \( 2n - 3 \leq e \leq n(n - 1)/2 \) is a visibility graph. Consider the \( n = 5 \) node graph \( G \) shown in Fig. 7.2a with \( e = 8 > 2n - 3 \) edges. We present now an ad hoc argument showing that it is not the visibility graph of any polygon.

The vertex \( v_1 \) is adjacent to all other vertices. This implies that the polygon is a fan: a star polygon with \( v_1 \) in the kernel. It is clear that the arcs corresponding to the boundary edges of the polygon form a Hamiltonian cycle of the graph, since the definition of visibility permits the endpoints of each polygon boundary edge to see one another. There are four distinct Hamiltonian cycles in \( G \), as illustrated in Fig. 7.2b; but due to the symmetry of \( G \), all four lead to the same structure of internal visibility lines. So we can restrict our attention to Fig. 7.2a. Because \( v_2 \) can see \( v_5 \), \( v_1 \) cannot be reflex. Thus \((v_1, v_2, v_5)\) forms an ear of the polygon. Because \( v_3 \) and \( v_4 \) can see \( v_1 \), \( v_2 \) and \( v_5 \) must be convex, as illustrated in Fig. 7.2c. Vertex \( v_4 \) must be reflex to block \( v_3 \)'s line of sight to \( v_5 \). But this implies that \( v_2 \) can see \( v_4 \), an arc not in \( G \). This establishes that \( G \) is not a visibility graph of a polygon.

One of the few general properties of visibility graphs that can be stated is the obvious one used in the preceding paragraph: a visibility graph must contain at least one Hamiltonian cycle, corresponding to the boundary of the polygon. Unfortunately, the problem of deciding whether an arbitrary graph contains a Hamiltonian cycle is NP-complete (Garey and Johnson).

Fig. 7.1. Visibility graphs can have as many as \( n(n-1)/2 \) edges (a) or as few as \( 2n - 3 \) (b).

---

1. How abundant is not known. It is not even known if the number of visibility graphs is \( \Omega(n^2) \).
1979). It is an interesting open question to see if the problem of finding a Hamiltonian cycle in a visibility graph remains intractable.

Given this uncertainty surrounding the complexity of finding a Hamiltonian cycle, ElGindy and Avis posed the problem of determining whether a given graph can be embedded in the plane as a visibility graph with a given Hamiltonian cycle forming the boundary of the polygon. Equivalently, we can assume the vertices of the given graph are labeled 1, ..., n such that the path (1, ..., n) forms a Hamiltonian cycle. We call the problem of determining (if possible) an embedding for such a labeled graph the visibility graph reconstruction problem. Even in this (apparently) easier form, the problem remains unsolved. However, ElGindy obtained an interesting result by further specialization.

### 7.2.1. Maximal Outerplanar Graphs

Maximal outerplanar graphs are an important subclass of planar graphs with many applications. A graph is outerplanar if it can be embedded in the
plane so that all of its nodes lie on the exterior face. In our setting this exterior face is the boundary of the polygon. A maximal outerplanar graph (or mop) is an outerplanar graph such that the addition of a single arc results in a graph that is not outerplanar. ElGindy showed that every mop is a visibility graph, and provided an algorithm for constructing a representative embedding (ElGindy 1985).

We will only consider mops of at least three nodes. The arcs on the exterior face will be called exterior arcs; all others are interior arcs. We will first state a few properties of mops.

**LEMMA 7.1.** A graph is a mop if it is a triangulation graph of a polygon.

*Proof.* Let $G$ be a triangulation graph of a polygon. $G$ is clearly outerplanar. No internal diagonals may be added without crossing other diagonals, and any external diagonal necessarily hides a vertex from the exterior face. Thus $G$ is maximal, and a mop. This establishes the lemma in one direction.

Let $G$ be a mop. We establish that:

1. There is exactly one vertex adjacent to both endpoints of each exterior edge.
2. There are exactly two vertices adjacent to both endpoints of each interior edge.
3. $G$ has no cutpoints.

Together these three conditions imply that $G$ is a triangulation graph of a polygon.

Let $e$ be an exterior edge of $G$, and suppose in contradiction to (1), vertices $x$ and $y$ are both adjacent to both endpoints of $e$. In an embedding of $G$, $x$ and $y$ must lie in the same half-plane determined by $e$, otherwise $e$ would not be exterior. But then either $G$ is non-planar (Fig. 7.3a), or one of $x$ or $y$ is interior (Fig. 7.3b) so $G$ is not outerplanar. In either case, we reach a contradiction, establishing (1).

Let $e$ be an interior edge of $G$. By the same argument as above, there can be at most one vertex in each of the two half-planes determined by $e$.

*Fig. 7.3.* The endpoints of an exterior edge $e$ can both be adjacent to only one vertex.
adjacent to both endpoints, and there must be one in each since \( e \) is interior. This establishes (2).

Let \( x \) be a cutpoint\(^2\) of \( G \), and let \( G_1, \ldots, G_k, k \geq 2, \) be the components of \( G - x \). Then if \( G \) is embedded with \( G_i \) angularly adjacent to \( G_{i+1} \) about \( x \), an arc may be added between \( G_i \) and \( G_{i+1} \) for \( i = 1, \ldots, k - 1 \) without making \( x \) internal; see Fig. 7.4. Thus \( G \) is not maximally outerplanar, establishing (3) by contradiction.

Finally, (1) and (2) show that \( G \) is composed entirely of triangles, and (3) shows that the boundary is a polygon. \( \square \)

Now that the class of mops has been revealed to be the familiar class of polygon triangulation graphs, the next property we need is obvious.

**Lemma 7.2.** A mop \( G \) has a unique Hamiltonian cycle.

**Proof.** The exterior edges of \( G \) correspond to the boundary of the polygon, and clearly form a Hamiltonian cycle. Since each interior edge cuts the polygon in two pieces, the inclusion of an interior edge in a Hamiltonian cycle would force the path into one piece or the other without possibility of return. Thus the exterior edge Hamiltonian cycle is unique. \( \square \)

It is easy to find the unique Hamiltonian cycle in linear time from any standard representation of \( G \) (Beyer et al. 1979). Thus for mops the usually difficult problem of identifying a Hamiltonian cycle becomes easy.

We may now state ElGindy's result.

**Theorem 7.1 [ElGindy 1985].** Every mop \( G \) is a vertex visibility graph of a monotone polygon.

Following ElGindy, we will establish this theorem by presenting an embedding algorithm and proving its correctness. First we will present a small example to illustrate the main ideas.

Consider the mop \( G \) of seven nodes shown in Fig. 7.5a. We first embed the triangle \((1, 2, 6)\) with \((1, 2)\) horizontal. We then embed 7 horizontally between 1 and 6 and above the \((2, 6)\) line, and embed 4 between 6 and 2

---

\(^2\) A cutpoint \( x \) of a graph \( G \) is a node whose removal (deletion of \( x \) and all incident edges) disconnects the graph. The remainder of \( G \) is denoted by \( G - x \).
and above the (1, 6) line; see Fig. 7.5b. Finally, 5 is embedded between 6 and 4 and above the (2, 4) line, and 3 is embedded between 4 and 2 and above the (6, 4) line. The result is a polygon monotone with respect to the horizontal with $G$ as its visibility graph. The algorithm can be phrased as a recursive procedure for embedding a triangle. Its three inputs are $B$, a bottom bounding line segment, and $l$ and $r$, the left and right embedded vertices forming the base of the triangle. The procedure marks each vertex it embeds; initially all vertices are unmarked.

**procedure** triangle($B$, $l$, $r$)

$m$ ← a point $ε$ above the midpoint of $B$.

if there exists an unmarked vertex $v$ adjacent to both $l$ and $r$ then

Erase ($l$, $r$) [unless ($l$, $r$) = (1, 2)].

Embed $v$ at $m$ and mark $v$.

Draw ($l$, $v$) and ($v$, $r$).

Update adjacency lists.

$B$ ← ($v$, $r$) extended between $l$ and $v$.

triangle($B$, $l$, $v$).

$B$ ← ($l$, $v$) extended between $v$ and $r$.

triangle($B$, $v$, $r$).

The procedure is called initially with $l = 1$, $r = 2$, and $B = (1, 2)$. We now argue for its correctness.
7.2. VERTEX VISIBILITY GRAPHS

Consider a particular invocation of triangle($B, l, r$). Assume as an induction hypothesis that the algorithm has performed correctly so far. Then $(l, r)$ represents an exterior edge of the polygon embedded so far, and so both are adjacent to exactly one vertex $x$. $B$ is determined by $xr$. Let $a$ and $b$ be the intersection points of $(m, l)$ and $(m, r)$ with the polygon boundary when extended, as illustrated in Fig. 7.6. Then the trapezoid $(a, b, r, l)$ must be empty of embedded vertices: $x$ is outside since $m$ is above $B$, and any other vertex inside would see both $l$ and $r$, contradicting the uniqueness of $x$. Thus $m$ is only visible to $l$ and $r$, correctly embedding the visibility edges of the given graph.

Although we have not detailed the data structure manipulations in the algorithm, it is not difficult to implement the algorithm to achieve $O(n)$ time complexity.

7.2.2. Convex Fans

ElGindy also studied a special class of polygons he called “convex fans.” A fan is a star polygon whose kernel includes a vertex of the polygon. A convex fan is a fan whose kernel includes a convex vertex. An example is shown in Fig. 7.7. The problem of reconstructing a representative polygon that achieves a given vertex visibility graph of a convex fan seems far easier than the general problem. ElGindy conjectured a characterization (ElGindy 1985), but reconstruction remains elusive. In fact, it is not even clear how to reconstruct an orthogonal convex fan from its visibility graph, despite the highly constrained staircase structure of such polygons (see Fig. 7.8). At this writing I only can see how to reconstruct orthogonal convex fans of uniform step height, a very special case.

Finally, it should be mentioned that Ghosh has recently established necessary conditions for a graph to be a vertex visibility graph, and conjectures that his conditions are also sufficient (Ghosh 1986). One of his conditions is that every “ordered” cycle (one whose labels are in sorted order) in the graph of at least four nodes, must have a chord. This explains why the graph of Fig. 7.2a is not a visibility graph: the 4-cycle $(2, 3, 4, 5)$ has no chord. A proof of his conjecture (his other conditions are not easy to state succinctly) seems difficult.
Considerable simplification of the visibility graph problem results by restricting the polygon and the visibility relation to the orthogonal world. Define an orthogonal edge visibility graph (or just a visibility graph) $G$ for an orthogonal polygon $P$ as follows. $G$ contains a node for each edge of $P$, and two nodes associated with horizontal [vertical] edges $e_i$ and $e_j$ are

---

3. The research reported in this section is the result of collaboration with Heather Booth.
connected by an undirected arc in $G$ iff they can see one another along a vertical [horizontal] line—that is, iff there exists a vertical [horizontal] line segment interior to $P$ with endpoints on $e_i$ and $e_j$ and which does not otherwise intersect the boundary of $P$. We will restrict our attention to polygons in general position: those such that no two vertices can be connected by an internal horizontal or vertical line segment that does not intersect $P$'s boundary. Throughout this section we will use “polygon” to mean “orthogonal polygon in general position.”

We will show in the next subsection that the visibility graph of an orthogonal polygon consists of two disjoint trees. Together these trees always have exactly $n - 2$ edges for a polygon of $n$ vertices; this is in marked contrast with the wide variability possible with vertex visibility graphs. We will say that a tree is realizable if there is a polygon that has the tree as one of the two components of its visibility graph. A labeling of a tree of $n$ nodes is a bijection between the nodes and the set of integers $\{0, 1, \ldots, n-1\}$. A labeling of a tree is realizable if there is a polygon that realizes the tree, and such that the polygon edges may be numbered $0, 1, \ldots, n-1$ in a counterclockwise traversal of the boundary to agree with the labeling. Finally, we say that two trees can mesh if they are jointly realizable by the same polygon.

In this section we characterize which single trees are realizable and which of their labelings are realizable. Meshable labelings of tree pairs are also characterized, and two algorithms for constructing a realization of two labeled trees are provided. Finally, we provide a partial characterization of when two (unlabeled) trees can mesh. Extending this to a complete characterization is the major open question raised by this investigation.

The next subsection establishes the basic properties of visibility trees, and presents an algorithm for constructing a realization. Section 7.3.2 proves the characterization theorems for labeled trees. Section 7.3.3 studies unlabeled trees, and concludes with a characterization of “universal” trees.

### 7.3.1. Realization of Visibility Trees

Without the general position assumption, the visibility graph could have many disconnected components, as shown in Fig. 7.9. For polygons in general position, however, the visibility graph has just two components, the horizontal and vertical trees:

**Lemma 7.3.** The orthogonal edge visibility graph $G$ of a polygon $P$ of $n$ edges in general position consists of two disconnected trees, $T_H$ and $T_V$, each of $n/2$ nodes.

**Proof.** By our definition of visibility, horizontal edges cannot see vertical edges. Thus there must be at least two components. We now show that the horizontal edges form a single tree, which we call the vertical tree $T_V$ for the polygon; note that “vertical” here refers to the direction of visibility, not to the orientation of the edges.
We first show that the nodes corresponding to any two horizontal edges $e_i$ and $e_j$ of $P$ are connected by a path in $G$. Let $x_i$ and $x_j$ be points in the interior of $P$ vertically visible to $e_i$ and $e_j$, respectively. Because the interior of $P$ is connected, there is a simple path $\pi$ within $P$ connecting $x_i$ to $x_j$. Let $a_x$ and $b_x$ be the horizontal edges above and below a point $x$ of $\pi$. There is an arc between the nodes in $G$ corresponding to $a_x$ and $b_x$. Now imagine $x$ moving from $x_i$ to $x_j$ along $\pi$. Because of the general position assumption, at most one of $a_x$ or $b_x$ changes at any point $x$. Thus for any transition point $x$, there are arcs in $G$ between the nodes corresponding to the edges just before $x$ to those corresponding to the edges at $x$. Thus the collection of nodes associated with $a_x$ and $b_x$ for all $x$ on $\pi$ forms a connected graph. See Fig. 7.10. Thus the vertical visibility component of $G$ is connected.

It only remains to show that this component of $G$ has no cycles. But it is clear that if it did, then $P$ would have a hole, contradicting the assumption that $P$ is a polygon without holes. \(\square\)

We now make a brief exploration of realizability, a concept which will be studied further in the following subsections.

**Lemma 7.4.** Every tree is realizable.

**Proof.** Given a tree $T$, choose any node as root, and assign it level 0. Assign to each other node $a$ a level equal to the number of arcs in $T$ from the root to $a$. First construct a "staircase" polygon as shown in Fig. 7.11a that realizes $T$ as its vertical visibility tree to level 1. Then add staircases to
the steps to include level 2 nodes (Fig. 7.11b). This process can clearly be extended indefinitely, at each stage capturing the nodes at one level lower (Fig. 7.11c). The resulting polygon realizes $T$ as its vertical visibility tree.

This simple result naturally leads to the next question:

**Lemma 7.5.** Not every pair of trees are jointly realizable.

**Proof.** The tree shown in Fig. 7.12 cannot mesh with itself, providing the smallest example of a pair of trees that cannot be simultaneously realized by one polygon. We defer a proof of this to Lemma 7.14 in Section 7.3.3.

Before investigating the properties of visibility trees further, we describe algorithms to compute the visibility trees for a given polygon, and to construct a realization for a labeled tree pair.

Computation of, say, the horizontal visibility tree can be accomplished easily in $O(n \log n)$ time with a plane sweep. The vertical edges of the polygon are sorted by their upper end point. A sweep line $H$ is then passed over the polygon from top to bottom, stopping at each vertex. A data structure $S$ holds all the vertical edges pierced by $H$, organized into a dictionary. Vertical edges pierced by $H$ alternately bound the interior of $P$ and the exterior of $P$. When two vertical edges become newly adjacent in $S$, then if they bound the interior of $P$, an arc is connected between their corresponding nodes. Insertion and deletion of vertical edges into $S$ can be performed in $O(\log n)$ time with appropriate implementation. The result is that the entire visibility tree can be constructed with a single pass in $O(n \log n)$.

However, the Tarjan–Van Wyk triangulation algorithm constructs a trapezoidization in $O(n \log \log n)$ time (Section 1.3.2), and this trapezoidization contains all the information necessary to construct a visibility tree. This yields the following lemma.

**Fig. 7.12.** The smallest tree that cannot mesh with itself, $S_5(3)$. 
**Lemma 7.6.** The visibility trees of a given polygon can be constructed in $O(n \log \log n)$ time.

Algorithms for the other direction, constructing a polygon that realizes two given labeled trees, are more interesting. Here we will briefly sketch one algorithm, and later (Theorem 7.2) provide another.

It will be convenient to introduce here a concept equivalent to labeling but which dispenses with the integer labels. Define a *circle embedding* of a tree (or just an *embedding*) as a layout of the tree within a circle in the plane such that each arc is mapped to a chord of the circle, and all nodes of the tree are mapped to lie on the circle. The circle corresponds to the boundary of the polygon "inflated" to a circle, and an embedding is topologically equivalent to a layout of the visibility trees within the polygon itself. Figure 7.13 shows an example. Any embedding of a tree of $n$ nodes corresponds to $n$ counterclockwise labelings of the tree, corresponding to the $n$ choices for the location of the 0 label. A labeling of a tree maps to a unique embedding, and often labels will be drawn around the circle. Embeddings ignore the irrelevant distinction between labelings that result from the circular shifts around the circle.

Now consider all the arcs incident to some particular node, say node 0 in Fig. 7.13. The portion of the polygon that realizes the subtree from 0 to its immediate descendants is a "histogram" (Section 2.3) or "Manhattan skyline" (Wood and Yap 1985) polygon, consisting of bottom edge 0 and top edges 12, 14, 18, and 24 in that order counterclockwise. In fact, this polygon is the orthogonal *edge visibility* polygon (Avis and Toussaint 1981b) for edge 0, enclosing all the points visible to edge 0 by an internal vertical line segment. We would now like to identify which vertical edges of the

![Fig. 7.13. A circle embedding of a visibility graph. The solid lines represent vertical visibility arcs, and the dashed horizontal arcs.](image-url)
Let \( e \) be a bottom edge whose node has degree greater than one, and let \( a \) and \( b \) be two edges visible from \( e \), with the edges occurring in counterclockwise order \( e, a, b \), and with no edges between \( a \) and \( b \) also visible to \( e \). Then the vertical edge of \( e \)'s edge visibility polygon between \( a \) and \( b \) is either determined by \( a + 1 \) (mod \( n \)) or \( b - 1 \) (mod \( n \)), as illustrated in Figs. 7.14a and 7.14b, respectively. Which case obtains can be decided by checking whether node \( a + 1 \) (mod \( n \)) is connected by an arc to any node in the range from \( b \) counterclockwise to \( e \). Thus in Fig. 7.13, let \( e = 0 \), \( a = 14 \), and \( b = 18 \). Since \( a + 1 = 15 \) connects to 19, \( a + 1 \) is the determining vertical edge; for \( a = 18 \) and \( b = 24 \), \( b - 1 = 23 \) is the determining edge. Continuing in this manner, we can conclude from edge 0's edge visibility polygon that \( 25 < 23 < 15 < 13 < 1 \), where \( i < j \) means that \( x \)-coordinate of edge \( i \) is less than that of edge \( j \).

Combining this information for every horizontal edge's edge visibility polygon, we can construct a partial order for the vertical edges. Similarly, a partial order for the horizontal edges can be constructed by examining the edge visibility polygons for the vertical edges. The results for the example of Fig. 7.13 are:

**Vertical edges:**
\[
21 < 19 < 23 < 15 < 13 < 1 < 11 < 9 < 3 < 5 < 7
\]
\[
25 < \ldots < 17
\]
\[
0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 9 \ 10
\]

**Horizontal edges:**
\[
0 < 24 < 22 < 2 < 10 < 20 < 14 < 16 < 18
\]
\[
6 < 4 < \ldots < 8 < 12
\]
\[
0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8
\]

Below each partial order are listed integers that will be used as the \( x \)-coordinate for the odd vertical edges, and the \( y \)-coordinate for the even
horizontal edges. The construction of a polygon that realizes these partial orders is now straightforward: the vertex between an adjacent horizontal and vertical edge is given an $x$-coordinate by the vertical edge and a $y$-coordinate by the horizontal edge, using the indices listed below the partial orders shown above. Thus the vertex between edges 0 and 1 lies at $(5,0)$, between 1 and 2 at $(5,3)$, and so on. The polygon generated from this process is illustrated in Fig. 7.15, and indeed it realizes the visibility graph in Fig. 7.13. This polygon may self-overlap (the one in Fig. 7.15 does at $(4,7)$), but these overlaps may be removed easily by adjusting edge lengths.

**Lemma 7.7.** A joint realization of two given labeled trees can be constructed in $O(n)$ time.

**Proof Sketch.** Each edge visibility polygon produces a chain of orderings. Each vertical edge may appear in at most four visibility polygons of horizontal edges, and similarly for each horizontal edge. Thus the total number of elements of these chains is no more than $4n$. It is easy to merge these chains in linear time (a claim we will not support here), resulting in $O(n)$ time to compute the partial orders. The remainder of the construction also takes just linear time. Details will not be presented. 

### 7.3.2. Realization of Labeled Trees

In this section we characterize when one tree or a pair of labeled trees are realizable. Note that Lemmas 7.4 and 7.5 address the equivalent questions for unlabeled trees. The answers we provide in this section are more complicated.

We first study embeddings of one tree. Call two nodes of an embedded tree 2-adjacent if they are adjacent on the circle if the other tree is ignored; note that 2-adjacent nodes receive labels $i$ and $i + 2(\text{mod } n)$ in a labeling. Define the *distance* between two 2-adjacent nodes as the number of arcs in the path in the tree between them. Let $d_i$ be the distance between node $i$
and its counterclockwise neighbor. Our first characterization theorem is that an embedding of a tree is realizable iff (1) the chords are non-crossing, and (2) for all $i$, $d_i \leq 3$. We now present a series of lemmas leading to the proof of this theorem.

**Lemma 7.8.** $\sum d_i = 2(n - 1)$, for any non-crossing embedding of a tree of $n$ nodes, where the sum is taken over all nodes $i$ of the tree.

*Proof.* Figure 7.16 shows that summing up the distances between all 2-adjacent nodes is equivalent to traversing the tree twice. Since a tree of $n$ nodes has $n - 1$ edges, the sum of the distances is $2(n - 1)$. □

**Lemma 7.9.** Let $n_k$ be the number of 2-adjacent node pairs to one side of a chord of a non-crossing embedding, that are separated by distance $k$ in the tree (the endpoints of the chord are included). Then $\sum_{k=1}^{\infty} (2 - k)n_k = 1$.

*Proof.* Remove the portion of the tree to the other side of the distinguished chord, and let the resulting tree have $n$ nodes. Then $\sum n_k + 1 = n$, since this sum counts the total number of nodes. Also we have $\sum k n_k + 1 = \sum d_i$, since both sides of the equation represent the total sum of distances. Applying Lemma 7.8 yields $\sum d_i = 2(n - 1)$. Substituting these relationships into the claimed equation proves the lemma:

$$\sum (2 - k)n_k = 2 \sum n_k - \sum kn_k$$

$$= 2(n - 1) - [2(n - 1) - 1]$$

$$= 1 \quad \square$$

Lemmas 7.8 and 7.9 both hold for arbitrary circular embeddings. We now focus on realizable embeddings.

**Lemma 7.10.** In a realizable embedding, the distance $d_i$ between each pair of 2-adjacent nodes satisfies $d_i \leq 3$. Moreover, the two angles in a realizing polygon between two 2-adjacent nodes are determined by $d_i$ as...
Fig. 7.17. Four possible angle sequences at two vertices: cc, cr, rc, and rr.

follows, where c and r mean convex and reflex angles, respectively:

\[ d_i = 1: \quad \text{cc} \]
\[ d_i = 2: \quad \text{rc or cr} \]
\[ d_i = 3: \quad \text{rr} \]

**Proof.** Let an arbitrary horizontal edge of a polygon, which we take to be a bottom edge without loss of generality, be labeled 0, and label the remaining edges with increasing index counterclockwise. The right endpoint of 0 is either a convex or a reflex vertex. If this endpoint is convex, then distinguish two further cases, depending on whether the upper endpoint of 1 is convex or reflex. The former case is illustrated in Fig. 7.17a, and justifies the claimed correspondence between \( d_i = 1 \) and cc. In the latter case (Fig. 7.17b), there must be an edge \( a \) above 1, leading to \( d_i = 2 \) and angles cr.

If the right endpoint of 0 is reflex, we again have two cases depending on whether the lower endpoint of 1 is convex or reflex. In the former case (Fig. 7.17c), there is again an edge \( a \) above 1, and \( d_i = 2 \) with angles rc. In the latter case (Fig. 7.17d), there must be an edge \( a \) above 1 and an edge \( b \) below, leading to \( d_i = 3 \) with angles rr.

As there are no further possibilities, the lemma is established. □

Let \( C \) and \( R \) be the number of convex and reflex angles in a polygon. Then it follows from Lemma 2.12 that \( C - R = 4 \). We can derive this relationship from Lemmas 7.9 and 7.10 as follows. By Lemma 7.10, \( d_i \leq 3 \), so the equation in Lemma 7.9 becomes \( n_1 - n_3 = 1 \). The correspondence between values of \( d_i \) and the included convex and reflex angles in Lemma 7.10 shows that the excess of \( C \) over \( R \) is determined by \( 2n_1 \) and the excess of \( R \) over \( C \) by \( 2n_3 \). Choosing any arc of an embedded tree and applying
Lemma 7.9 to each side shows that \( C \) exceeds \( R \) by \( 2(n_1 - n_3) = 2 \) on each side. Therefore, \( C - R = 4 \) for the entire polygon.

This argument shows that if an embedding satisfies \( d_i \leq 3 \), then a polygon with the appropriate number of convex and reflex angles exists, since it is known that all angle sequences that satisfy \( C - R = 4 \) are achievable (Culberson and Rawlins 1985). It is only a short step further to show that a polygon exists that realizes the embedding:

**THEOREM 7.2** [Booth and O'Rourke 1985]. An embedding of a tree is realizable iff

1. No two chords cross: the embedding is planar; and
2. For all nodes \( i, d_i \leq 3 \).

**Proof.** Assume that polygon \( P \) realizes an embedding of \( T \) as its vertical visibility tree. Lemma 7.10 establishes that (2) holds for the embedding. Suppose that (1) does not hold. Let arc \( ab \) be crossed by \( cd \). Since \( ab \) is an arc, the corresponding horizontal edges \( a \) and \( b \) of \( P \) may be connected with a vertical line segment \( L \). Since the edge \( c \) is distinct from \( a \) and \( b \), it must be strictly to the right or left of \( L \). If the arcs cross, then edge \( d \) must be on the other side of \( L \). But then edges \( c \) and \( d \) cannot be connected by a vertical line, contradicting the inclusion of \( cd \) as an arc of the visibility tree. This establishes the easy direction of the theorem.

Assume that an embedding of \( T \) satisfies (1) and (2). We seek to construct a polygon \( P \) that realizes the embedding. The construction starts with a rectangle, and successively refines it, encompassing more and more of the given embedding. Choose any arc of the embedding as a starting point. The vertical visibility tree consisting of this single arc is realized by a rectangle.

For the general refinement step, assume that we have constructed a polygon \( P' \) that realizes an interior subtree \( T' \) of the embedding. An interior subtree \( T' \) is one such that all arcs not in \( T' \) are towards the outer boundary of the circle, in the sense that each arc of \( T - T' \) has only arcs of \( T - T' \) to one side, or equivalently, no arc of \( T - T' \) has arcs of \( T' \) to both sides. For example, the edges \((0, 12), (0, 14), \) and \((2, 12)\) in Fig. 7.13 form an interior subtree of the vertical visibility tree (solid lines). In the same figure, the edges \((0, 12) \) and \((0, 18)\) do not form an interior subtree, because the edge \((0, 14)\) has edges of the subtree to both sides. Choose an arbitrary boundary arc \( ab \) of \( T' \)—that is, one that has only arcs of \( T - T' \) to one side \( S \), or equivalently, no arcs of \( T' \) to one side. We now specify two nodes \( c \) and \( d \) within this sector \( S \). All the nodes in \( S \) are connected by a path in \( T \) to either \( a \) or \( b \) without using arc \( ab \). Because no two arcs cross (1), it must be the case that there are two 2-adjacent nodes \( c \) and \( d \) such that \( c \) is connected to \( a \) and \( d \) is connected to \( b \). See Fig. 7.18. (It may be that \( c = a \) and/or \( d = b \).) By (2), the distance \( \delta_{cd} \) between \( c \) and \( d \) is no more than 3. The refinement step has three cases depending on the value of \( \delta_{cd} \).

**Case** \( \delta_{cd} = 1 \). Here we must have \( c = a \) and \( d = b \), and no refinement is needed.
Case $\delta_{cd} = 2$. Either $c = a$ or $d = b$ but not both; assume that $a = c$ without loss of generality. Then replace polygon $P'$ with $P$ as shown in Fig. 7.19a by replacing the indicated vertical edge by a step. (This introduces $cr$ angles; $rc$ angles could be introduced instead. Either leads to a realization. This incidentally shows that the realization need not be unique.) This new polygon realizes $T' \cup bd$, which is an internal subtree.

Case $\delta_{cd} = 3$. Both $c$ and $d$ are distinct from $a$ and $b$. Replace polygon $P'$ with $P$ as shown in Fig. 7.19b by introducing a tab into the indicated vertical edge. $P$ now realizes $T' \cup ac \cup bd$, which is an internal subtree.

This completes the description of the refinement step. Starting from an arbitrary arc and a rectangle, repeating this refinement step until all arcs have been included results in a polygon that realizes the original embedding. □
Note that conditions (1) and (2) of the theorem can both be phrased as constraints on labelings. For example, (2) can be phrased as: the distance in the tree between the nodes labeled $i$ and $i + 2(\text{mod } n)$ (where $n$ is the total number of vertices in the polygon) is no more than 3.

We now turn to the problem of characterizing when a pair of labeled or embedded trees are jointly realizable. We start with some observations on insufficient characterizations. If both the horizontal and vertical trees in an embedding individually satisfy the conditions of Theorem 7.2, they are not necessarily jointly realized by the same polygon. It may be that the two unlabeled trees cannot mesh (Lemma 7.5), so there is no jointly realizable embedding. Or it may be that the particular embeddings do not consistently mesh the $c/r$ patterns implied by Lemma 7.10. More surprisingly, it is possible for the embeddings to individually satisfy the conditions of Theorem 7.2, and to consistently mesh their implied $c/r$ patterns, and still not be jointly realizable. Figure 7.20a shows an example. The next lemma shows why this embedding is not realizable. Define two nodes as 1-adjacent if their labels differ by 1 (mod $n$); thus 1-adjacent nodes correspond to consecutive edges of a polygon, and lie in different trees.

**Lemma 7.11.** In a realizable embedding of two trees, all the regions on

![Fig. 7.20. Unrealizable embeddings: in (a), the 12-13 region is bound by 6 chords; in (b) 13 violates the projection constraint.](image)
the outer boundary between two 1-adjacent nodes of the circle are bound by either two or four chords. Moreover, if such a region is bound by two chords, then the included angle in a realization is convex; if by four chords, the angle is reflex.

Proof. If an angle of a polygon is convex, then the visibility arcs to the two 1-adjacent edges cross, and the closest such arcs to the corner bound the corner by two chords; see Fig. 7.21a. If an angle is reflex, then the corner is bound by four alternating horizontal and vertical arcs, as shown in Fig. 7.21b. □

The region between nodes 12 and 13 of Fig. 7.20a is bound by six chords, and so violates Lemma 7.11. However, it is possible for an embedding to satisfy the conditions of Lemma 7.11 and still not be realizable. Figure 7.20b shows an example. The next lemma explains why.

**Lemma 7.12.** In a realizable embedding of two trees, each consecutive triple of nodes $a, x, b$ must satisfy the following projection constraints (where $\delta_{ab}$ is the distance between $a$ and $b$ in their tree):

$\delta_{ab} = 1$. [no constraint]

$\delta_{ab} = 2$. Let $acb$ be the length 2 path in the tree from $a$ to $b$. Then every arc incident to $x$ must cross either $ac$, or $bc$, but not both. See Fig. 7.22a.

$\delta_{ab} = 3$. Let $acdb$ be the length 3 path in the tree from $a$ to $b$. Then every arc incident to $x$ must cross $cd$. See Fig. 7.22b.

Fig. 7.22. The two projection constraints.
7.3. EDGE VISIBILITY TREES IN ORTHOGONAL POLYGONS

Fig. 7.23. Embeddings illustrating the two constraints of Fig. 7.22.

Proof. By Lemma 7.10, \( \delta = 2 \) corresponds to \( cr \) and \( rc \) angles, and \( \delta = 3 \) to \( rr \) angles. Figure 7.23a shows that \( x \) can only project across \( bc \) when the included angles are reflex and convex; if the step is reversed, then \( x \) can only project across \( ac \). Figure 7.23b shows that \( x \) can only project across the middle of the three link path between \( a \) and \( b \).

Node 13 in Fig. 7.20b projects across both \((0, 4)\) and \((4, 8)\), violating the distance 3 case of the lemma.

Finally we show that these projection constraints are sufficient to characterize realizable embeddings.

**Theorem 7.3.** [Booth and O'Rourke 1985]. An embedding of two trees \( T_V \) and \( T_H \) is realizable iff

1. Each tree embedding satisfies the conditions of Theorem 7.2 (which characterized embeddings of single trees), and
2. Every node of both trees satisfies the projection constraints of Lemma 7.12.

Proof. Assume there is a polygon that jointly realizes the embeddings. Then it individually realizes each embedding, so (1) follows from Theorem 7.2. Lemma 7.12 shows that (2) is satisfied. This completes the proof in the easy direction.

Assume that an embedding is given that satisfies conditions (1) and (2) of the theorem. We seek to construct a polygon \( P \) that realizes \( T_V \) and \( T_H \) as its vertical and horizontal visibility trees. The construction is by a refinement procedure very similar to that used in the proof of Theorem 7.2, although here we must refine both trees simultaneously. The refinement will be phrased in terms of \( T_V \), with the refinement of \( T_H \) induced by that of \( T_V \). Choose any arc of the embedding of \( T_V \) as the starting point. The induced subtree of \( T_H \) is also a single arc, and these two subtrees are realized by a rectangle.

For the general refinement step, we need to define refinement induction. Let \( T'_V \) be an interior subtree of \( T_V \), as defined in the proof of Theorem 7.2. The induced subtree \( T'_H \) is constructed by identifying ("lumping" together) all nodes of \( T_H \) that lie inside a sector bound by one boundary arc of \( T'_V \). Thus when \( T'_V \) is a single arc, all nodes on either side of
the arc are lumped into one, and $T'_H$ is also just a single arc; this is the starting configuration. At any stage of the construction, each boundary arc of $T'_V$ encompasses just a single node of $T'_H$.

We now specify the actions taken during a refinement step. An arbitrary boundary arc $ab$ of $T'_V$ is chosen for refinement. The nodes $c$ and $d$ of $T_V$ are specified as in Theorem 7.2, and again the action taken depends on the distance $\delta_{cd}$ between $c$ and $d$.

**Case $\delta_{cd} = 1$.** No refinement needed.

**Case $\delta_{cd} = 2$.** As in Theorem 7.2, assume $a = c$; let $x$ be the node of $T_H$ between $a$ and $d$, and let $y$ represent all the nodes of $T_H$ between $d$ and $b$. By (2) of the theorem statement, $x$ must project across either $ab$ or $bd$. In the former case, modify $P'$ as shown in Fig. 7.24a, adjusting the relative lengths of the vertical edges $x$ and $y$ so that they are visible to the appropriate edges to the left of $ab$. In the latter case, modify as shown in Fig. 7.24b. Here the length of $x$ is arbitrary, as it only sees $y$. In either case, the new polygon $P$ jointly realizes $T'_V \cup bd$, which is an internal subtree, and the subtree of $T_H$ induced by $T'_V \cup bd$.

![Fig. 7.24. Refinement when $\delta_{cd} = 2$ (a and b) and $\delta_{cd} = 3$ (c).](image-url)
Case $\delta_{cd} = 3$. Let $y$ be the node of $T_H$ between $c$ and $d$, and let $x$ and $z$ represent the nodes of $T_H$ bound by the arcs $ac$ and $db$ respectively. Replace polygon $P'$ with $P$ as illustrated in Fig. 7.24c, adjusting the relative lengths of $x$, $y$, and $z$ so that they are opposite the appropriate edges on the opposite side of $ab$. $P$ jointly realizes $T'_V \cup ac \cup bd$ and the induced subgraph of $T_H$.

This completes the description of the refinement step. Applying this repeatedly constructs a polygon that jointly realizes the given embeddings of $T_V$ and $T_H$. □

The technique used in the proof provides an alternative to the algorithm sketched in Section 7.3.1 for constructing a realization of two labeled trees. The procedure can be implemented straightforwardly in linear time, as long as the children of each node are sorted by label. But this can be accomplished in linear time with a radix sort. Thus the alternative algorithm also runs in $O(n)$ time. Details would take us too far afield and will not be presented.

### 7.3.3. Universal Trees

Although Theorem 7.3 completely answers the question of when two labeled trees can mesh, it does not help much in deciding whether two unlabeled trees can mesh. It only says that two trees can mesh if there exists a labeling that satisfies the conditions of the theorem. As there are an exponential number of possible tree labelings, it would be useful to characterize realizability directly in terms of structural features of the unlabeled trees. I have been unable to find such a characterization. I can, however, give a partial characterization, as follows. Define a universal (unlabeled) tree as one that can mesh with any other tree of the same number of nodes. Recall that Lemma 7.5 claimed that not all trees are universal. In this section we prove that a tree is universal iff it is a caterpillar, a tree that does not contain the graph shown in Fig. 7.12 as a subtree. The goal of this section is to prove this theorem.

One of the keys to the theorem is the study of trees similar to that shown in Fig. 7.12, which we call 2-stars. A 2-star $S_2(k)$ is a tree of $k$ length 2 paths joined at one node, called the root of the 2-star. Thus $S_2(k)$ has $2k + 1$ nodes. The tree in Fig. 7.12 is $S_2(3)$.

**Lemma 7.13.** Any tree $T$ that meshes with $S_2(k)$ must have an embedding in which at most one pair of 2-adjacent nodes $a$ and $b$ are separated by a distance of 3. Moreover, if there is such a pair, then a jointly realizable embedding must locate the root of $S_2(k)$ between $a$ and $b$.

**Proof.** The constraints of Theorem 7.2 imply that there are only two essentially different realizable embeddings of $S_2(k)$: one in which all the level 2 nodes (where the root is level 0) are immediately counterclockwise of their level 1 parents (Fig. 7.25a), and one in which a portion of the level
2 nodes are clockwise and the remaining counterclockwise of their level 1 parents (Fig. 7.25b). Any other arrangement leads to nodes being separated by a distance greater than 3, violating (2) of the theorem. Applying Lemma 7.10, we obtain the following sequence of convex and reflex angles for the two embeddings, where standard regular expression notation is used:

(a) \( cc (cc \, rr)^{k-1} \, cc \, (cr + rc) \)
(b) \( (cr + rc)(rr \, cc)^i (cr + rc)(cc \, rr)^j \, cc \, (cr + rc), \) where \( i + j = k - 2. \)

Both sequences start at the root and proceed counterclockwise. For example, in Fig. 7.25b, \( i = 1, j = 2, \) and \( k = 5. \)

Now any embedding of \( T \) that is jointly realizable with \( S_2(k) \) must consistently mesh its angle sequence implied by Lemma 7.10 with either (a) or (b). It is clear that any sequence that meshes with (a) contains no instance of \( rr, \) and that a sequence that meshes with (b) can contain at most one instance of \( rr, \) obtained by choosing the \( rc \) alternative counterclockwise of the root in (b), and choosing the \( cr \) alternative at the end of (b), just clockwise of the root. In this case, the root of \( S_2(k) \) lies between two reflex vertices. •

**Lemma 7.14.** \( S_2(k) \) cannot mesh with any tree \( T \) that contains \( S_2(3) \) as a subtree.

**Proof.** Figure 7.26 shows three distinct ways that the \( S_2(3) \) subtree of \( T \) may be embedded. A fourth embedding may be obtained by replacing the right half of Fig. 7.26c (right of \( rc \)) with the right half of Fig. 7.26a, but we
7.3. EDGE VISIBILITY TREES IN ORTHOGONAL POLYGONS

will see that both cases are handled identically. All other possible embeddings are reflections of these. (Note that since in general $S_2(3)$ will be a proper subtree of $T$, nodes $d$ and $f$ in Fig. 7.26c may satisfy $\delta_{df} \leq 3$ due to intervening arcs.) By Lemmas 7.9 and 7.10, we know that $n_1 - n_3 = 1$, where $n_k$ are the number of pairs of nodes to one side of any arc that are separated by distance $k$. Recall that by Lemma 7.10, $n_1$ counts the number of included $cc$ angles, and $n_3$ the number of $rr$ angles.

In Fig. 7.26a, there must be two instances of $rr$ angles to the left of arc $ra$, since $ab$, $ef$, and $cd$ each contribute at least 1 to $n_1$, which forces $n_3 \geq 2$. Lemma 7.13 now shows that $S_2(k)$ cannot mesh with this embedding, since at most one pair of nodes can be separated by distance 3.

In Fig. 7.26c, there must be at least two instances of $rr$ angles to the left of arc $re$, since $cd$, $ef$, and $re$ each contribute at least one to $n_1$. This eliminates this and the fourth embedding not drawn as embeddings that can mesh with $S_2(k)$.

Figure 7.26b represents a potential mesh, since there can be just one instance of $rr$ angles, to the left of arc $rc$ and $ra$. In fact, it is clear that this one instance must lie between $d$ and $e$. By Lemma 7.13, if $S_2(k)$ is to mesh with this embedding, the root must be fall between these two reflex vertices, and so lies between $d$ and $e$. In Lemma 7.12 we proved that a node between two 2-adjacent nodes separated by distance 3 must project through the middle link. A similar statement may be proved for any node between two perhaps non-2-adjacent nodes separated by distance 3; the proof is similar to that of Lemma 7.12 and will not be detailed. In this particular situation, it says that the root of $S_2(k)$ cannot project across $re$ or $cd$, but may either stay within $ercd$ or project across $rc$. But this means that it is impossible for $S_2(k)$ both to reach nodes between $e$ and $f$, and between $f$ and $r$. This establishes that $S_2(k)$ cannot mesh with $T$. □

This lemma finally provides the proof for Lemma 7.5. The lemma establishes that a tree that contains $S_2(3)$ as a subtree is not universal, because there is at least one tree with which it cannot mesh. Trees that do not contain $S_2(3)$ are known as caterpillars, and have the general appearance shown in Fig. 7.27a. They may be characterized as trees with the property that removal of all leaf nodes (and their incident arcs) results in a simple path.

**Theorem 7.4** [O'Rourke 1985]. A tree is universal iff it is a caterpillar.

**Proof.** Lemma 7.14 establishes that every universal tree is a caterpillar. We now show that every caterpillar can mesh with every other tree with the same number of nodes, and so is universal. The proof is constructive: given an arbitrary caterpillar and any other tree with the same number of nodes, we construct a polygon that jointly realizes them. The proof is somewhat complicated, so we will present an overview first.

Define an *hourglass* polygon that realizes a vertical visibility tree $T$ as follows. Let $e$ be a distinguished edge of $T$, and assign a level to all nodes of
Arrange the nodes of a caterpillar $C$ along two vertical lines, as shown in Fig. 7.27b. Let $l$ and $r$ with $l \geq r$ be the number of nodes on the left and right lines, respectively. Clearly each $C$ has a unique $(l, r)$ pair. We claim that if $T$ contains an edge $e$ such that the remaining nodes of $T$ can be embedded to either side of $e$, with $l - 1$ nodes to the left and $r - 1$ to the right, then $C$ and $T$ can be jointly realized by an hourglass polygon. To see this, first construct an hourglass polygon realizing the described embedding of $T$. Next, adjust the lengths of the vertical edges to achieve $C$ as the horizontal tree, as illustrated in Fig. 7.28. Clearly this can always be done if the hourglass polygon has $l$ vertical edges on the left and $r$ on the right, which it must by our assumptions. When an hourglass polygon realizes $C$ and $T$ in this way, we say the two trees can balance.

![Fig. 7.27. A caterpillar (a) and its arrangement in two vertical columns (b).](image)

$T$ as their minimum distance from an endpoint of $e$. The polygon is defined by a series of refinements, as was done in Lemma 7.4. At the level 0 step, the polygon is a rectangle, and realizes just $e$. Next inward and outward staircases replace the vertical sides of the rectangle to include all level 1 nodes of $T$, as shown in Fig. 7.28, giving the polygon its characteristic shape. Staircases are added to each step to include level 2 nodes (just as in Fig. 7.11), and so on. Hourglass polygons have the property that they easily realize certain caterpillars.

![Fig. 7.28. An hourglass polygon realizes a caterpillar as its horizontal tree; here the vertical tree is refined only to level 1.](image)
We can now sketch the proof. Given C and T, if they can balance, then the balancing hourglass polygon establishes the theorem. If they cannot balance, then we will show that we can arrange to balance a subtree C' of C and a subtree T' of T with an hourglass polygon, and gather the remaining nodes C - C' and T - T' in an isolated region. The process is repeated, with the "leftover" diminishing at each step. The final result is a series of hourglass polygons, horizontally displaced and connected top to bottom, which together jointly realizes C and T. We now proceed with the details.

Let the given C have the (l, r) pair \( l_1 \) and \( r_1 \), with \( l_1 + r_1 = n \) and \( l_1 \geq r_1 \), and let \( l_1 - r_1 = \delta \geq 0 \). Choose an arbitrary node \( b_1 \) of T as the "base." Choose an arc \( e_1 \) of T incident to \( b_1 \) such that the remaining nodes of T may be arranged on either side of \( e_1 \), with \( L_1 \) nodes to the left and \( R_1 \) to the right (so \( L_1 + R_1 + 2 = n \)), such that the quantity

\[
\beta_1 = |(L_1 - R_1) - \delta_1|
\]

is minimal among all arcs \( e_1 \) incident to \( b_1 \) and all arrangements of nodes. That is, \( L_1 \) and \( R_1 \) are as close to \( l_1 \) and \( r_1 \) as is possible for the choice of \( b_1 \); \( \beta_1 \) is a measure of "imbalance" between C and T. Consider now two cases: \( \beta_1 = 0 \) and \( \beta_1 > 0 \).

**Case \( \beta_1 = 0 \).** Then

\[
L_1 - R_1 = \delta_1 = l_1 - r_1.
\]

Substituting \( R_1 = n - 2 - L_1 \), and \( r_1 = n - l_1 \), yields

\[
L_1 = l_1 - 1
\]

\[
R_1 = r_1 - 1
\]

There are precisely the conditions for exact balance, so the hourglass polygon jointly realizes C and T, establishing the theorem.

**Case \( \beta_1 > 0 \).** Let \( t_1 \) be the top node at the other end of \( e_1 \). Call whichever side (right or left) has an excess (or equal amount) of T nodes over C nodes the high side. We first prove that \( t_1 \) must have descendants to the high side. Suppose that left is the high side, and that \( t_1 \) has no descendants to that side. Then \( b_1 \) must have descendants to the left, otherwise \( L_1 = 0 \), and since \( L_1 \geq l_1 \), \( l_1 = 0 \), which is impossible. Then by choosing any edge \( e'_1 \) incident to \( b_1 \) and contributing to \( L_1 \) as a new partition edge as illustrated in Fig. 7.29, \( L_1 \) can be decremented by 1 and \( R_1 \) can be incremented by 1. This decreases \( L_1 - R_1 \) by 2. Since \( \beta_1 = |(L_1 - R_1) - \delta_1| \) is minimal, and since

![Fig. 7.29. Shifting one node from \( L_1 \) to \( R_1 \) when \( t_1 \) has no left descendants.](image-url)
$$(L_1 - R_1) > \delta_1 > 0$$ in the case under consideration (left side high and $$l_1 > r_1$$), it can only be the case that $$\beta_1 = 1$$ and the change by 2 changes the signed term from +1 to −1. But we now show that $$\beta_1$$ can never equal 1. Suppose

$$(L_1 - R_1) - \delta_1 = 1.$$ 

Substituting $$\delta_1 = l_1 - r_1$$, $$R_1 = n - 2 - L_1$$, and $$r_1 = n - l_1$$ yields

$$L_1 - l_1 = -\frac{1}{2},$$

which is impossible as the difference must be an integer. The argument is similar if the right is the high side. Therefore, $$t_1$$ has descendants to the high side.

Let $$\Delta_1$$ be the minimal number of nodes in a subtree descendant of $$t_1$$ to the high side. We prove that $$\Delta_1 \geq \beta_1$$. For suppose $$\Delta_1 < \beta_1$$. Assume first that left is the high side, so that $$L_1 \geq l_1$$ and $$R_1 < r_1$$. Then flipping $$\Delta_1$$ nodes from $$L_1$$ to $$R_1$$ changes them to

$$L'_1 = L_1 - \Delta_1$$

$$R'_1 = R_1 + \Delta_1.$$ 

These imply a $$\beta'_1$$ of

$$\beta'_1 = |(L'_1 - R'_1) - \delta_1|.$$ 

Since $$(L_1 - R_1) > \delta_1$$ in this case, we have $$\beta_1 = (L_1 - R_1) - \delta_1$$. Substituting for $$L'_1$$, $$R'_1$$, and $$\delta_1$$ yields

$$\beta'_1 = -2\Delta_1 + \beta_1,$$

which is less than $$\beta_1$$ under the assumption $$\Delta_1 < \beta_1$$, contradicting the assumed minimality of $$\beta_1$$. If on the other hand right is the high side, then flipping $$\Delta_1$$ from right to left, and noting that now $$-\beta_1 = (L_1 - R_1) - \delta_1$$, leads to

$$\beta'_1 = 2\Delta_1 - \beta_1,$$

again a contradiction for the same reason.

This establishes that $$\Delta_1 \geq \beta_1$$.

Now the plan is to remove $$\Delta_1$$ nodes from the high side, balance the remainder with an hourglass polygon, and start another hourglass polygon off to the side for the $$\Delta_1$$ nodes. If $$\Delta_1$$ nodes are removed from the high side, then calling the new quantities to either side of $$e_1 L_1$$ and $$R_1$$, it must be that both $$L'_1 < l_1$$ and $$R'_1 < r_1$$, since $$\Delta_1$$ is more than enough to reduce the excess to a deficit on the high side. (Note that it may be that one of $$L'_1 = 0$$ or $$R'_1 = 0$$ holds.) Call $$T$$ minus these $$\Delta_1$$ nodes $$T_1$$.

We now specify an appropriate spot to “slide” the $$\Delta_1$$ nodes into a second hourglass. Consider again the layout of $$C$$ between two vertical lines, as illustrated in Fig. 7.30. Define a diagonal of $$C$$ to be either the lowest or the highest arc in the layout, or one whose endpoints are both of degree more than 1. Let a diagonal $$d$$ have $$d_L$$ and $$d_R$$ nodes of $$C$$ below it on the left and
right lines of the layout respectively. Define diagonal $d^*$ to be the lowest diagonal such that either $L'_1 < d^*_L$ and $d^*_R < R'_1$ (Fig. 7.30a), or $L'_1 > d^*_L$ and $d^*_R > R'_1$ (Fig. 7.30b); $d^*$ always exists.

If $d^*_L > L'_1$, then the slide will be to the left; if $d^*_R > R'_1$, then the slide will be to the right. Assume the latter case without loss of generality. Form an hourglass polygon with $L'_1 + 1$ and $R'_1 + 1$ nodes on the left and right that realizes $T$. Let $l_2 = l_1 - L'_1 - 1 > 0$ and $r_2 = r_1 - R'_1 - 1 > 0$. Adjust the lengths of the vertical edges of the hourglass polygon so that (a) it realizes $C$ as far up the layout as possible (call this subtree $C_1$), and (b) the remaining $l_2$ and $r_2$ nodes can be laid out at the top right. This is illustrated in Fig. 7.31. Nodes above $d^*_L$ but below $L'_1 + 1$ must be realized in the hourglass polygon $H_1$; nodes below $d^*_R$ but above $R'_1 + 1$ must be slid to the right.

The hourglass polygon $H_1$ now realizes $T_1$ and $C_1$. It should be clear that the remaining $\Delta_1$ nodes of $T$, and $l_2$ and $r_2$ of $C$, bring us back to the exact same situation as we faced at the start. The base node $b_2$ for the vertical tree is fixed; an edge $e_2$ is chosen to minimize $\beta_2$. If $\beta_2$ is zero, then the polygon can be completed with an hourglass. If $\beta_2 > 0$, then the extra nodes are slid off the top to either the right or left as appropriate, and the process repeated.

Let $\Delta_i$ be the number of nodes slid off from $T$ at the $i$th step of this procedure. Clearly $\Delta_i > \Delta_{i+1}$, since $\Delta_{i+1}$ is chosen from among the $\Delta_i$.

**Fig. 7.31.** A "slide" to the right.
Fig. 7.32. An example of meshing a caterpillar (a) with a tree (b). The first hourglass polygon is shown in (c), and the final polygon realizing both trees in (d). Only the diagonals of the caterpillar are shown in (d).
nodes, and $\Delta_{i+1}$ cannot include the endpoints of $e_i$. Thus the sequence of $\Delta$'s is strictly decreasing. Also we proved that $\Delta_i \geq \beta_i$, so the sequence of $\beta$ imbalances are bounded by a strictly decreasing sequence. Thus we must eventually reach a step $j$, where $\beta_j = 0$, and the polygon can be completed with a balanced hourglass. This completes the construction. □

Figure 7.32 illustrates the result of the construction procedure of the theorem in a particular case. $C$ is shown in Fig. 7.32a; it has $l_1 = 14$ and $r_1 = 10$, so $\delta_1 = l_1 - r_1 = 4$. The tree $T$ to be meshed is shown in Fig. 7.32b, with $e_1$ and $b_1$ distinguished. It is easy to check that the minimal $\beta_1$ is achieved as illustrated, with $L_1 = 12$ and $R_1 = 10$. Thus $\beta_1 = |(L_1 - R_1) - \delta_1| = 2$. Right is the high side, since $R_1 \geq r_1$. There is no choice for $\Delta_1$: we must slide off all of $R_1$. Thus $\Delta_1 = 10$. After removal of $\Delta_1$, we have $L_1' = 12$ and $R_1' = 0$. The diagonal $d^*$ is the lowest edge in Fig. 7.32a whose endpoints both have degree greater than 1. The construction of the first hourglass polygon is shown in Fig. 7.32c; the caterpillar is not drawn in this figure. For the second polygon, $l_2 = l_1 - L_1' - 1 = 1$ and $r_2 = r_1 - R_1' - 1 = 9$ nodes remain. Thus $\delta_2 = 8$. Now it is easy to arrange the nodes in $T$ such that balance is achieved: $L_2 = 0$ and $R_2 = 8$. These imply that $\beta_2 = 0$: exact balance. The final polygon is shown in Fig. 7.32d.

7.3.4. Discussion

The major question left unresolved is: given two unlabeled trees, determine whether or not they can mesh, and if so, construct a realization. If either of the given trees is a caterpillar, Theorem 7.4 provides the answer. But the ratio of the number of caterpillars on $n$ nodes to the number of free trees on $n$ nodes goes to zero as $n$ goes to infinity, so caterpillars are relatively infrequent. If both trees are not caterpillars, we can only answer this question by trying all possible labelings and applying Theorem 7.3. A characterization in terms of the structure of the two trees would be desirable.

7.4. BAR VISIBILITY GRAPHS

Perhaps the most satisfying result obtained to date in the area of visibility graphs is the characterization theorem of “bar visibility graphs” obtained by Wismath (1985), and independently by Tamassia and Tollis (1985). In a bar visibility graph (or just a bar graph), the nodes represent vertical line segments, and two nodes are connected by an arc iff their two vertical bars $A$ and $B$ can see each other horizontally and non-degenerately. More precisely, there must exist a non-zero height rectangle bounded by $A$ and $B$ to the right and left that does not intersect any other bar. This notion of visibility is sometimes called $\epsilon$-visibility (Tamassia and Tollis 1986), since the bars must be able to see one another over a beam of visibility of
thickness $\varepsilon > 0$. Note that this non-degenerate visibility was used in the previous section, but there it was enforced by requiring the polygon to be in general position.

Bar visibility graphs differ from orthogonal polygon visibility trees in two respects: the bars do not have to connect into a polygon, and they have two “sides.” These two changes considerably widen the class of graphs that are bar representable—that is, those for which there is a collection of bars that realize the visibility graph. First it is clear that every bar graph is planar. Embed the nodes of the graph at, say, the lower endpoint of each bar. Since none of the visibility rectangles intersect, the arcs of the graph may be embedded along the length of these rectangles and down the sides of the bars without crossovers. But not every planar graph is bar representable. Figure 7.33a shows a graph $G$, and Fig. 7.33b shows an attempted realization that fails to embed a bar for node 6. After presentation of the characterization theorem, we will see that this is the smallest bar unrepresentable planar graph. We will follow Wismath’s presentation throughout (Wismath 1985).

Our first positive result, halfway to the characterization theorem, is that every biconnected planar graph (defined below) is bar representable. To prove this result we must first introduce “st-numbering.” An st-numbering of a graph $G$ of $n$ nodes is a 1-1 function $\lambda$ that maps each node to a distinct number in $\{1, 2, \ldots, n\}$, with $\lambda(s) = 1$ and $\lambda(t) = n$, where $s$ and $t$ are two distinguished nodes, such that for every node $y$ different from $s$ and $t$, there are nodes $x$ and $z$ adjacent to $y$ such that $\lambda(x) < \lambda(y) < \lambda(z)$. The two distinguished nodes $s$ and $t$ can be thought of as source and termination points of a “PERT” digraph, where each edge is oriented to point from lower to higher labels. We now quote two results on st-numbering as lemmas (Lempel et al. 1967; Even and Tarjan 1970).
7.4. BAR VISIBILITY GRAPHS

LEMMA 7.15 [Lempel, Even, and Cederbaum 1966]. For every edge $st$ of a biconnected graph $G$, there is an $st$-numbering of $G$.

LEMMA 7.16 [Even and Tarjan 1976]. Given an edge $st$ of a biconnected graph, an $st$-numbering can be found in $O(n)$ time.

A biconnected graph is one that contains no cut point, no node whose removal disconnects the graph; in other words, there are at least two disjoint paths between any two distinct nodes. It is easy to construct connected planar graphs which are not biconnected, and which are not $st$-numerable. Therefore, attention will be restricted to biconnected graphs. Wismath's first result is the following.

LEMMA 7.17. Every planar biconnected graph $G$ is bar representable.

Proof. By Lemma 7.15, $G$ is $st$-numerable for any edge $st$. Fix an edge $st$ and an $st$-numberation. The proof is by construction of a layout of bars, based on this $st$-numbering, that realizes $G$. We will illustrate the proof with the graph $G$ shown in Fig. 7.34a with the indicated $st$-numbering.

---

Fig. 7.34. An $st$-numbered graph (a), and embedding into strips (b) of nodes (solid dots) and pseudo-nodes (open circles), and a realization (c).

4. "Nonseparable" is a synonym for "biconnected," and "articulation point" is a synonym for "cut point."
Start with a planar embedding of $G$ as in Fig. 7.34a, and deform $G$, maintaining planarity, so that the vertices are arranged in vertical strips $S_1, S_2, \ldots$, such that

1. $s \in S_1$.
2. For all $v \neq s$, $v \in S_i$ iff all lower numbered nodes adjacent to $v$ are in strips $S_1, \ldots, S_{i-1}$.

This is shown in Fig. 7.34b (solid nodes). No two vertices in the same strip are adjacent. Thus they can safely be mapped to bars with the same $x$-coordinate, since they cannot see one another. All that remains is adjustment of the lengths and placements of the bars to match the graph edges. Wismath accomplished this with the following clever scheme.

Introduce a "pseudo-node" on each edge that passes completely through a strip; see the open nodes in Fig. 7.34b. Let $v_{ij}$ be the $j$th node (real or pseudo) from the bottom in strip $i$, and let $x_{ij}$, $a_{ij}$, and $b_{ij}$ be the $x$-coordinate, top, and bottom $y$-coordinates, respectively, of the bar associated with node $v_{ij}$, and let $L_{ij} = a_{ij} - b_{ij}$. For $v_{11} = s$, let $x_{11} = 0$, $a_{11} = 1$, $b_{11} = 0$, so $L_{11} = 1$. Define the length of the bar for $v_{ij}$ to be long enough to encompass all its incident "beams" from the left, each of which is a proportional fraction of the lengths of the bars from which they emanate:

$$L_{ij} = \sum_k \frac{L_{i-1,k}}{\deg_f(v_{i-1,k})},$$

where $\deg_f(v)$ is the forward degree of $v$—that is, the number of neighbors to its right. The bars are placed at the same $x$-coordinate within each section, stacked end to end vertically:

- $b_{i1} = b_{11} = 0$,
- $a_{ij} = b_{ij} + L_{ij}$,
- $b_{ij} = a_{i,j-1}$.

Applying this procedure to Fig. 7.34b produces the layout shown in Fig. 7.34c. For example, the length of the only bar for a real node (7) in $S_5$ is $L_{51} = L_{41} + (1/2)L_{31} + L_{32} + (1/3)L_{23}$. Finally, the bars representing the introduced pseudo-nodes are removed. Clearly the resulting layout realizes $G$. □

It is easy to implement the construction in this proof with a linear algorithm using Lemma 7.16.

The class of bar representable graphs is wider than established by this lemma. This class can be described by loosening the definition of $st$-numerability, as follows. Let $\lambda$ be a 1-1 labeling function from the nodes to $\{1, 2, \ldots, n\}$. Define a $\lambda$-$\max$ to be a node that has no higher labeled neighbor, and similarly define a $\lambda$-$\min$. An $st$-numbering has one $\lambda$-$\min$ and $\lambda$-$\max$, at $s$ and $t$, respectively. Define a graph $G$ to be $st^*$-numerable if there is a $\lambda$ function and a planar embedding of $G$ such that all $\lambda$-$\max$ and
all \( \lambda \)-min nodes are on the exterior face, and they are separable into a \( \lambda \)-max group and a \( \lambda \)-min group, separable in the sense that it is possible to introduce two new vertices \( u_- \) and \( u_+ \) in the exterior face such that \( u_- \) connects to each \( \lambda \)-min node and \( u_+ \) to each \( \lambda \)-max node, while preserving planarity. This definition is admittedly ungainly, but it will be rephrased shortly. First we show that it precisely captures the bar representable graphs.

**THEOREM 7.5.** A biconnected graph \( G \) is bar representable iff \( G \) is st*-
numerable.

**Proof.** Suppose \( G \) is st*-
numerable. Then embed \( G \) as guaranteed by the definition of st*-
numerability. If there is more than one \( \lambda \)-max node, add a new vertex \( v_+ \) in the exterior face and connect it to all \( \lambda \)-max nodes. Similarly add \( v_- \) connected to all \( \lambda \)-min nodes if there is more than one. Call the resulting graph \( G' \). Now assign \( \lambda(u_-) = 0 \) and \( \lambda(v_+) = n + 1 \). Then it is clear that each node except \( u_- \) and \( v_+ \) has a smaller and larger labeled neighbor. Thus \( G' \) is st-
numerable. Since \( G \) was assumed to be biconnected, and since \( v_+ \) and \( u_- \) (if present) have degree at least two, \( G' \) is also biconnected. Lemma 7.17 then guarantees that \( G' \) is bar representable. Removing the bars associated with the \( v_- \) and \( v_+ \) nodes results in a realization for \( G \).

Suppose \( G \) is bar representable, realized by a particular layout of bars. Convert the layout into normal form by translating the bars to the vertical lines \( x = 1, 2, \ldots \), and moving each one as far to the left as is possible. Now number the bars 1 to \( n \), from left to right, top to bottom. We claim that this is a st*-
numbering of the associated graph \( G \). Consider any interior node \( v \) of \( G \), one not on the boundary of the exterior face. Then it must have a neighbor to its left and its right. But since the left neighbors have lower and the right neighbors higher labels, \( v \) is neither a \( \lambda \)-min nor a \( \lambda \)-max node. The \( \lambda \)-min and \( \lambda \)-max nodes are separable, since all the \( \lambda \)-max nodes occur for bars at tops of the columns, and \( \lambda \)-min nodes at the bottoms, and these bars are on the exterior face.

We now show that this theorem can be reformulated to replace st*-
numbering with more direct graph-theoretic features.

**THEOREM 7.6** [Wismath, Tomassia and Tollis 1985]. \( G \) is bar representable iff there is a planar embedding of \( G \) with all cutpoints on the exterior face.

**Proof.** Suppose \( G \) is bar representable. Then by Theorem 7.5, it is st*-
numerable, which by definition implies there is a planar embedding realizing the st*-
numeration. Suppose, in contradiction to the theorem, that there is a cutpoint \( x \) not on the exterior face of this embedding. Since \( x \) is a cutpoint, its removal disconnects \( G \) into two or more components. Because \( x \) is not on the exterior face, one of these components, say \( B \), must be interior to a circuit \( C \) containing \( x \); see Fig. 7.35. Now let \( b_+ \) be the node of
$B$ that has the largest label among all the nodes of $B$, and let $b_-$ be the node that achieves the smallest. Then $x$ is the only connection between $B$ and $G - B$, and since $\lambda(x)$ cannot be both larger than $\lambda(b_+)$ and smaller than $\lambda(b_-)$, either $b_+$ is a $\lambda$-max or $b_-$ is a $\lambda$-min. This violates the definition of $st^*$-numbering, establishing the "if" direction of the theorem.

Suppose there is a planar embedding of $G$ with all cutpoints on the exterior face. Then we can construct an $st^*$-numbering by using the $st$-numbering algorithm of Even and Tarjan (Lemma 7.16) to number the "blocks" of $G$ depth-first. A block of a graph is a maximal biconnected component. Treat their algorithm as a procedure $ST(s, t, B, i)$ that $st$-numbers block $B$ starting with $\lambda(s) = i$, where $st$ is an edge of $B$. Each call to $ST$ marks the block that it numbers. Then an algorithm for $st^*$-numbering from cutpoint $s$ with $\lambda(s) = i$ is $ST^*(s, i)$ below.

$$ST^*(s, i)$$

for all unmarked blocks $B_i$ containing a cutpoint $x$ do

Choose unmarked $t$ adjacent to $x$ on the exterior face of $B_i$.

$ST(x, t, B_i, i)$.

$i \leftarrow i + |B_i| - 1$.

for all cutpoints $x_j$ in $B_i$ do

$ST^*(x_j, i)$.

Note that only the first cutpoint passed to $ST^*$ becomes a $\lambda$-min; all the other cutpoints become $\lambda$-maxima. Therefore the $\lambda$-min and $\lambda$-max nodes are trivially separable. They are all on the exterior face because the cutpoints are. Thus a valid $st^*$-numbering is produced. Theorem 7.5 then implies that $G$ is bar representable, concluding the proof of the theorem. □

We obtain the final form of the characterization with a simple observation.

**COROLLARY.** Let $G^+$ be the graph obtained from $G$ by connecting every cut point to a new vertex. Then $G$ is bar representable iff $G^+$ is planar.

Since cutpoints of a graph can be identified in linear time by a depth-first search (Aho et al. 1974), and planarity may be tested in linear time with the Hopcroft–Tarjan algorithm (Hopcroft and Tarjan 1974), this yields a
linear-time algorithm for recognizing bar representability. A simple modification of the layout procedure used in the proof of Theorem 7.5 establishes that a realization can be constructed in linear time.

We may finally see why the graph $G$ in Fig. 7.33a is not bar representable. Add a node 0 connected to the three cutpoints 2, 5, and 7. Then the graph may be redrawn as in Fig. 7.36, revealing that $K_{3,3}$ is a subgraph. By Kuratowski's theorem, then, $G$ is non-planar. It is also easy to see that this is the smallest bar unrepresentable graph.