Representations of Graphs on a Cylinder

Roberto Tamassia and Ioannis G. Tollis

Technical Report No. CS-89-10

February 1989

REPRESENTATIONS OF GRAPHS ON A CYLINDER

Roberto Tamassia*

Department of Computer Science
Box 1910
Brown University
Providence, RI 02912

Ioannis G. Tollis**

Department of Computer Science
P.O. Box 830688, MP 3.1
The University of Texas at Dallas
Richardson, TX 75083-0688

Abstract

We present a complete characterization of the class of graphs that admit a *cylindric visibility representation*, where vertices are represented by intervals parallel to the axis of the cylinder and the edges correspond to visible intervals. Moreover, we give linear time algorithms for testing the existence of and constructing such a representation. Important applications of cylindric visibility representations can be found in the layout of regular VLSI circuits, such as linear systolic arrays and bit-slice architectures. Also, we present alternative ''dual'' characterizations of the graphs that admit visibility representations in the plane and in the cylinder. It is interesting to observe that neither of these two classes is contained in the other, although they have a nonempty intersection.

Key Words: visibility graph, visibility representation, design and analysis of algorithms, computational geometry, cylinder, planar graph, centipede graph.

AMS (MOS) Subject Classifications: 68R10, 68U05, 05C10, 05C75.

(January 1989)

---

* Research partially supported by Cadre Technologies, Inc..

** Research partially supported by the Texas Advanced Research Program under grant no. 3972.

1. INTRODUCTION

The concept of *visibility* plays a fundamental role in a variety of geometric applications [10]. Of particular interest are the problems dealing with visibility graphs between parallel *intervals*, which have various applications in VLSI layout [4,7,8,13,17], motion planning [5,11], and graph drawing [1,12,14,15]. The combinatorial properties of these graphs have also been extensively investigated [2,14,16,18]. Visibility graphs also arise in the well-known hidden-surface elimination problem for two- and three-dimensional figures.

Let I be a set of parallel intervals in the plane, where an interval is a segment that might or might not contain one or both of its endpoints. Two intervals are said to be *visible* if they can be joined by a line orthogonal to them which does not intersect any other interval of I. The *visibility graph* of I is the graph whose vertices are the intervals of I, and whose edges connect pairs of visible intervals. Conversely, a *visibility representation* for a graph G is a set of intervals whose visibility graph is isomorphic to G. It has been shown that G admits a visibility representation in the plane if and only if there exists a planar embedding for G such that all the cutpoints appear on the boundary of the same face [14,18]. Furthermore, such a representation can be constructed in linear time [14].

In this paper we consider visibility on a cylindric surface, where vertices are associated with intervals parallel to the axis of the cylinder. Namely, we present a characterization of the class of graphs that admit a visibility representation in the cylinder, and give linear time algorithms for testing the existence of and constructing such a representation. Important applications of *cylindric visibility representations* can be found in the layout of regular VLSI circuits, such as linear systolic arrays and bit-slice architectures [9]. Also, we present alternative "dual" characterizations of the graphs that admit visibility representations in the plane and in the cylinder. It is interesting to observe that neither of these two classes is contained in the other, although they have a nonempty intersection.

Let $G=(V,E)$ be a connected graph. A *cutpoint* of G is a vertex whose removal disconnects G. If G has at least one cutpoint, it is said to be *1-connected*, otherwise it is said to be *2-connected*. A *block* of G is a maximal 2-connected subgraph of G. The *block-cutpoint* tree T of G is the graph whose vertices are the blocks and the cutpoints of G, and whose edges connect every block B to the cutpoints contained in B. T can be constructed in $O(|E|)$ time by using depth-first search [3]. A *centipede* graph is a tree whose non-leaf vertices form a chain. Figure 1.1 shows a graph whose block-cutpoint tree, given in Figure 1.2, is a centipede graph. Blocks and cutpoints are represented by large and small circles, respectively.

We show that a graph admits a cylindric visibility representation if and only if it is planar and its block-cutpoint tree is a centipede graph. We present linear time algorithms for testing the existence of and constructing a cylindric visibility representation for a given graph. Also, we characterize visibility representations in the plane and in the cylinder by means of the block-cutpoint trees of the dual graphs. Namely, we show that a planar graph G admits (i) a planar visibility representation if and only if it admits a dual graph G^* whose block-cutpoint tree is a star; and (ii) a cylindric visibility representation if and only if it admits a dual graph G^* whose block-cutpoint tree is a chain.

Notice that there exist planar graphs whose block-cutpoint tree is a centipede graph and such that in no planar embedding all the cutpoints appear on the boundary of the same face. Indeed, this is the case for the graph of Figure 1. Conversely, the existence of a visibility representation in the plane does not impose any restriction on the structure of the block-cutpoint tree. For example, Figure 2.1 shows a graph that admits a visibility representation in the plane, but whose block-cutpoint tree (Figure 2.2) is not a centipede graph. Therefore, visibility in the plane does not imply visibility in the cylinder, and vice versa.

In the next section we introduce the concept of *cylindric orientation* and show its relation with cylindric visibility representations. In Section 3 we present a linear time algorithm that constructs a cylindric visibility representation for any 2-connected planar graph. The characterization of cylindric visibility representations is given in Section 4. Finally, the dual characterizations are presented in Section 5.

2. CYLINDRIC ORIENTATIONS

An (*infinite*) *cylinder* C is the locus of points at the same distance from a straight line, called the *axis* of the cylinder. A cylinder is also the union of an infinite family of circles with center on the axis and drawn on a plane orthogonal to the axis. Alternatively, C is the union of an infinite family of lines parallel to the axis, and at the same distance from it. The circles and lines of C naturally define a coordinate system, since every point of C is the unique intersection of a line and a circle. Every point of C will be denoted by a pair (x,θ), where x is measured on the axis and θ is the angle with respect to some reference line and some "clockwise" orientation, with $0 \leq \theta < 2\pi$.

We will consider *cylindric embeddings* of graphs, where vertices are mapped into points of the cylinder, and edges are mapped into nonintersecting Jordan curves on the cylinder. Clearly, a graph admits a cylindric embedding if and only if it is planar. Unlike planar embeddings, cylindric embeddings can have two unbounded faces, which are referred to as the *leftmost face* and *rightmost face* of the embedding.

Let Γ be a cylindric embedding with distinct leftmost and rightmost faces. A cycle γ of Γ is said to *wrap around* cylinder C if it intersects any Jordan curve on the surface of C with endpoints in the leftmost and rightmost faces of Γ, respectively. In other words, the removal of γ disconnects C into two unbounded pieces. A *cylindric orientation* Γ' of Γ is an orientation of the edges of Γ such that:

(1) Γ' has no sources (vertices without incoming edges) and no sinks (vertices without outgoing edges);

(2) every directed cycle wraps around C in the clockwise direction.

The following lemma gives two important properties of cylindric orientations. It can be proved using arguments similar to the ones in the proof of Lemmas 1 and 2 of [14], which give analogous properties for planar st-graphs.

Lemma 1 Let Γ' be a cylindric orientation. We have:

(1) for every vertex v of Γ', the incoming (outgoing) directed edges appear consecutively around v, see Figure 3.1;

(2) the boundary of each internal face of Γ' consists of two directed paths with common origin and destination, see Figure 3.2. □

In the following, the concepts of *left* and *right* refer to the orientation of the edges in a cylindric orientation. For example, the face to the left of a directed edge $[u,v]$, denoted $LEFT(u,v)$, is the face containing $[u,v]$ which appears on the left side when traversing $[u,v]$

from vertex u to vertex v. Face $RIGHT(u,v)$ is symmetrically defined. By Lemma 1, for each vertex v, there are two distinct faces that separate the incoming edges from the outgoing edges. These faces are denoted by $LEFT(v)$ and $RIGHT(v)$, where $LEFT(v)$ is the face to the left of the leftmost incoming and outgoing edges, and $RIGHT(v)$ is the face to the right of the rightmost incoming and outgoing edges, see Figure 3.1.

An *interval* of cylinder C is a topologically connected subset of a line of C. An *arc* of C is a topologically connected subset of a circle of C. Let I be a set of disjoint intervals of C. Two intervals i_1 and i_2 of I are said to be *visible* if there is an arc a of C with endpoints on i_1 and i_2 that does not intersect any other interval of I. Arc a is said to be a *visibility-arc* between i_1 and i_2, and is directed according to the clockwise orientation. In other words, let θ_1 and θ_2 be the angles of i_1 and i_2, respectively, with $\theta_1 < \theta_2$; if a does not intersect the reference line then we direct a from i_1 to i_2, otherwise we direct a from i_2 to i_1.

A *cylindric visibility representation* K for a graph G is a mapping of vertices of G into disjoint intervals of C, called *vertex-intervals*, such that there is an edge (u,v) if and only if the intervals associated with u and v are *visible*. Each edge of G is mapped into either one or two visibility-arcs. In the latter case, the union of the two visibility-arcs is a circle of C. For simplicity, we use the same name for the vertices of the graph and their corresponding vertex-intervals. Figure 4.2 shows a cylindric visibility representation for the graph of Figure 4.1. Note that the top and bottom heavy lines represent the same line of the cylinder, say the reference line.

From K we can construct a cylindric orientation Γ by shrinking every vertex-interval into a point, and accordingly deforming the visibility-arcs, as shown in Figure 4.3. Note that the undirected graph obtained from Γ by ignoring the edge directions and the double edges is isomorphic to G. The above construction shows that any graph that admits a cylindric visibility representation must be planar.

Given a planar embedding Π of a 2-connected planar graph G, and two distinct faces f_1 and f_2 of Π, we can construct a cylindric orientation Γ of G with the same topology as Π, leftmost face f_1, and rightmost face f_2.

Algorithm *CYL-ORIENT*

Input: A 2-connected n-vertex planar graph $G = (V, E)$, with $n \geq 3$. A planar embedding Π for G, and two distinct faces f_1 and f_2 of Π.

Output: A cylindric orientation Γ of G with the same topology as Π, leftmost face f_1, and rightmost face f_2.

(1) Embed Π on the surface of a sphere.

(2) Pierce two holes in the sphere inside faces f_1 and f_2, and deform the pierced sphere into a cylinder. This gives a cylindrical embedding Π'.

(3) Orient the edges on the boundary of face f_1 in the clockwise direction. Mark face f_1 as "oriented".

(4) Let f be an unmarked face that is adjacent to a marked face. Since G is 2-connected, the edges on the boundary of f can be partitioned into two simple chains, γ_1 and γ_2, where γ_1 contains the oriented edges, and γ_2 contains the unoriented edges. Let u and v be the common endpoints of these two chains such that γ_1 is directed from u to v. We orient all the edges of γ_2 in the direction from u to v. This step is repeated until all faces are marked. { Note that this process is essentially a visit of the dual graph. } □

Theorem 1 Algorithm *CYL-ORIENT* constructs a cylindric orientation Γ for an n-vertex ($n \geq 3$) 2-connected planar graph G in $O(n)$ time.

Proof: The algorithm maintains the invariant that after each iteration of Step 4 the oriented portion of the graph is a cylindric orientation. If the orientation of γ_2 creates a cycle in the oriented portion that does not wrap around C in the clockwise direction, such cycle must be the union of γ_2 and of a directed chain γ_3 from v to u. This implies that the union of γ_1 and γ_3 is a cycle that does not wrap around C, which violates the invariant. The correctness of the algorithm easily follows by induction. Regarding the time complexity, we observe that $O(1)$ time is spent at each edge. □

Note that the cylindric orientation Γ constructed by the above algorithm does not have double edges. The cylindric orientation for a 2-connected graph with two vertices is a cycle consisting of two symmetrically directed edges, and can be trivially constructed.

3. CONSTRUCTION OF CYLINDRIC VISIBILITY REPRESENTATIONS

Given an acyclic digraph $D = (V, E)$, a *topological ordering* τ on D maps each vertex v into a nonnegative integer $\tau(v)$ such that $\tau(u) < \tau(v)$ for every directed edge $[u, v] \in E$. A topological ordering can be computed in $O(|V| + |E|)$ time by means of the following recursive formula:

(1) $\tau(s) = 0$, for every source-vertex $s \in V$

(2) $\tau(v) = 1 + \max_{[u,v] \in E} \tau(u)$.

A *face* of a cylindric visibility representation is a maximal topologically connected region of the cylinder delimited by the vertex-intervals and the visibility-arcs. The faces of a cylindric visibility representation are in one-to-one correspondence with the faces of the associated cylindric orientation. In the rest of this section we restrict our attention to 2-connected graphs, and show how to construct a cylindric visibility representation in linear time.

Algorithm *VISIB-2C*

Input: A 2-connected n-vertex planar graph $G = (V, E)$. A planar embedding Π for G, and two distinct faces f_1 and f_2.

Output: A cylindric visibility representation K for G with leftmost face f_1 and rightmost face f_2.

(1) Construct a cylindric orientation Γ of G with leftmost face f_1 and rightmost face f_2, using algorithm *CYL−ORIENT*.

{ The digraph Γ intuitively represents a "circular order" of the vertex-intervals in the θ-direction }

(2) Construct the dual digraph Γ^* of Γ, where dual edges are oriented "from left to right". Namely, the dual of $[u, v]$ is the directed edge $[LEFT(u, v), RIGHT(u, v)]$. Γ^* is acyclic an has exactly one source (f_1) and exactly one sink (f_2).

{ A directed edge $[f,g]$ in Γ^* implies that face f will be to the left of face g in the cylindric visibility representation }

(3) Compute a topological ordering α on the digraph obtained from Γ by removing the edges that intersect a path of Γ^* from f_1 to f_2.

{ The ordering α will be used for determining the θ-coordinates of the vertex-intervals and the visibility-arcs }

(4) Compute a topological ordering β on Γ^*.

{ The ordering β will be used for determining the x-coordinates of the vertex-intervals and the visibility-arcs }

(5) let $\theta_0 = \frac{2\pi}{n}$;

for each $v \in V$ **do**

draw a vertex-interval from $(\beta(LEFT(v)), \alpha(v)\theta_0)$ to $(\beta(RIGHT(v)), \alpha(v)\theta_0)$, which includes the left endpoint but not the right one;

endfor;

(6) **for each** $[u,v] \in \Gamma$ **do**

let $x = \frac{1}{2}\left[\beta(LEFT(u,v)) + \beta(RIGHT(u,v))\right]$;

draw a visibility-arc directed from $(x, \alpha(u)\theta_0)$ to $(x, \alpha(v)\theta_0)$;

endfor;

An example of the construction performed by algorithm *VISIB-2C* is shown in Figure 5: Part 1 shows a planar embedding of a 2-connected graph G. Part 2 shows the cylindric orientation Γ computed in Step 1, in heavy lines, and its dual Γ^* computed in Step 2; primal and dual vertices are labeled with the value of the corresponding topological ordering, computed in Step 3 or 4. Finally, part 3 shows the cylindric visibility representation K computed in Steps 5 and 6.

Now, we discuss the complexity of the algorithm. Since G is a planar graph, both G and its dual G^* have $O(n)$ vertices and edges. From the results of Section 2, Steps 1 and 2 take $O(n)$ time. The computation of α and β in Steps 3 and 4 can be performed in $O(n)$ time. Finally, Steps 5 and 6 take $O(n)$ time. Hence, we have the following theorem:

Theorem 2 Algorithm *VISIB-2C* constructs a cylindric visibility representation K for an n-vertex 2-connected planar graph G in $O(n)$ time. □

4. CYLINDRIC VISIBILITY REPRESENTATIONS AND CENTIPEDE GRAPHS

As discussed in the introduction, not every 1-connected planar graph admits a cylindric visibility representation. Here, we provide a necessary and sufficient condition that characterizes the class of graphs that admit such a representation. Before we prove the main theorem of this section, we need some preliminary results:

Lemma 2 Let K be a cylindric visibility representation for G, and γ be a circle of the cylinder that intersects at least three vertex-intervals of K. Then there is a cycle in G that consists of exactly the vertices associated with the vertex-intervals intersected by γ.

Proof: Any two consecutive vertex-intervals intersected by γ are visible, and thus the corresponding vertices are adjacent. □

We define a *section* of a cylinder C as the portion of C generated by the rotation of an interval i of C around the axis.

Lemma 3 Let K be a cylindric visibility representation for G, and c be a cutpoint of G. Then the vertex-interval of any other cutpoint c' of G is not completely contained in the section generated by the vertex-interval of c.

Proof: Assume that there is a cutpoint c', distinct from c, that is completely contained in the section generated by c. There must exist a block B that contains c' but not c. Let v be a vertex of B adjacent to c'. By Lemma 2, there is a cycle of G associated with a circle of the cylinder intersecting c, c', and v. Since every cycle must contain vertices of the same block, we obtain a contradiction. □

The following theorem characterizes the class of graphs that admit a cylindric visibility representation.

Theorem 3 A planar graph G admits a cylindric visibility representation if and only if its block-cutpoint tree T is a centipede graph.

Proof: (Necessity) We observe that T is a centipede graph if and only if every cutpoint of G is not contained in more than two distinct nonleaf blocks. Now, assume for a contradiction that such a cutpoint c indeed exists, and let x_L and x_R be the x-coordinates of its left and right endpoints, respectively. Let B_1, B_2, and B_3 be three distinct nonleaf blocks containing c, and for $i = 1,2,3$, consider a cutpoint c_i in B_i distinct from c. Such cutpoints exist since B_1, B_2, and B_3 are not leaves of T, see Figure 6.1. From Lemma 3, at least two of these cutpoints, say c_1 and c_2, are both on the same side of the section generated by c, i.e., either their left endpoints are on the left of x_L or their right endpoints are on the right of x_R. Without loss of generality, assume the first case. There must be vertices v_1 in B_1 and v_2 in B_2 whose vertex-intervals intersect the circle of the cylinder at abscissa x_L, see Figure 6.2. By Lemma 2, there is a cycle of G associated with this circle. Clearly, such cycle contains vertices from both B_1 and B_2, which is a contradiction to the fact that B_1 and B_2 are distinct blocks.

(Sufficiency) We show how to construct a cylindric visibility representation for the graph G from the cylindric visibility representations of its blocks. Denote the nonleaf blocks of G as $B_1, \cdots, B_k$, and the cutpoints of G as $c_0, c_1, \cdots, c_k$, where block B_i contains the cutpoints c_{i-1} and c_i, for $i = 1, \cdots, k$. In the construction of a cylindric visibility representation K_B for a block B of G, we have two cases:

Case 1: B is a leaf of T.

In this case B contains exactly one cutpoint c of G. We construct K_B such that both the leftmost and rightmost faces of K_B contain the cutpoint c. This can be done by selecting faces f_1 and f_2 in algorithm *VISIB-2C* as two faces containing c. Note that all the vertex-intervals of K_B are contained in the section of c.

Case 2: $B = B_i$, for some i in $\{1, \cdots, k\}$.

In this case, we construct K_B such that the leftmost face contains cutpoint c_{i-1} and the rightmost face contains cutpoint c_i.

At this point the cylindric visibility representations of all the blocks have been constructed. For each block B, slice K_B at abscissas $\beta(f_1)$ and $\beta(f_2)$. The sections so obtained are then glued together along a common axis, in such a way that, for each $i = 1, \cdots, k-1$, all the sections corresponding to leaf blocks connected to c_i are placed between the sections of blocks B_i and B_{i+1}. The leaf blocks connected to c_0 and c_k are placed before block B_1 and after block B_k, respectively. To complete the construction, the sections must be rotated so that the vertex-intervals of different sections corresponding to the same cutpoint become aligned. □

The construction described in the proof of Theorem 3 is illustrated in Figure 7. Part 1 shows the block-cutpoint tree of a graph, where blocks are denoted by upper-case letters, and cutpoints by lower-case letters. Part 2 shows the arrangement of the sections corresponding to the blocks along the cylinder. Note that leaf blocks A, B, D, E, G, H, and I are contained in the sections of their respective cutpoints. Also, nonleaf blocks C and F have one cutpoint at the left and the other cutpoint at the right, which link them to the rest of the representation.

The time complexity of the construction described in the proof of Theorem 3 is analyzed as follows. Let n be the number of vertices of G. The blocks and cutpoints of G can be computed in time $O(n)$ using depth-first search. By Theorem 2, the cylindric visibility representation of each block B is constructed time $O(m_B)$, where m_B is the number of edges in B. This sums up to $O(n)$ because each edge belongs to a single block. Finally, combining the sections of the blocks into a unique cylindric visibility representation is easily done in $O(n)$ time.

To test whether a planar graph G admits a cylindric visibility representation, we construct its block-cutpoint tree T, and determine whether it is a centipede graph. This can be done by removing all the leaves from T, and verifying that the resulting graph is a simple chain. The above computation takes $O(n)$ time. Hence, we conclude:

Theorem 4 Given a graph G with n vertices, there are $O(n)$ time algorithms for testing the existence of and constructing a cylindric visibility representation for G. □

5. A DUAL CHARACTERIZATION OF VISIBILITY REPRESENTATIONS

In this section we provide ''dual'' characterizations of the classes of graphs that admit planar visibility representations and cylindric visibility representations.

Theorem 5 A planar graph G admits a planar visibility representation if and only if it admits a dual graph G^* whose block-cutpoint tree is a star.

Proof: G is 2-connected if and only if all of its duals are [6, p. 124, ex. 11.4]. Hence, the theorem is trivially true for 2-connected graphs. Now, suppose that G is 1-connected. We recall that G admits a planar visibility representation if and only if it has a planar embedding such that all the cutpoints of G are on the same face, say the external face [14]. Hence, if G admits a planar visibility representation there exists a planar embedding of G such that all blocks are embedded in the external face. Let G^* be the dual graph associated with this embedding. G^* has exactly one cutpoint, i.e., the external face.

Conversely, suppose that G admits a dual graph G^* whose block-cutpoint tree is a star. The embedding associated with G^* has a face (the center of the star) containing all blocks, and hence all cutpoints of G. This implies that G admits a planar visibility representation. □

Theorem 6 A planar graph G admits a cylindric visibility representation if and only if it admits a dual graph G^* whose block-cutpoint tree is a chain.

Proof: Suppose that G admits a dual graph G^* whose block-cutpoint tree T^* is a chain. Let T be the block-cutpoint tree of G. If T is not a centipede, then there exists a block B of G which contains (at least) three cutpoints, denoted c_1, c_2, and c_3. Let B_1, B_2, and B_3 be blocks of G distinct from B that contain cutpoints c_1, c_2, and c_3, respectively. Since B^* has at most 2 cutpoints, one of them, denoted f, must contain at least two of these blocks, say B_1 and B_2, which are connected to B only through cutpoints c_1 and c_2. Hence in T^* f is adjacent to B^*, B_1^*, and B_2^*, a contradiction. Therefore, T is a centipede and, by Theorem 3, G admits a cylindric visibility representation.

Now, suppose that G admits a cylindric visibility representation, and consider the one constructed in the proof of Theorem 3. By construction, for every block B, the rest of the cylindric visibility representation is contained in the leftmost and rightmost faces of K_B. Shrink every vertex-segment to obtain a cylindric embedding of G. The above property is preserved by this transformation, and hence the block-cutpoint tree of the dual graph G^* is a simple path. □

REFERENCES

[1] G. Di Battista and R. Tamassia, "Algorithms for Plane Representations of Acyclic Digraphs," *Theoretical Computer Science*, vol. 61, pp. 175-198, 1988.

[2] P. Duchet, Y. Hamidoune, M. Las Vergnas, and H. Meyniel, "Representing a Planar Graph by Vertical Lines Joining Different Levels," *Discrete Mathematics*, vol. 46, pp. 319-321, 1983.

[3] S. Even, *Graph Algorithms*, Computer Science Press, Potomac, MD, 1979.

[4] M. Garey, D. Johnson, and H. So, "An Application of Graph Coloring to Printed Circuit Testing," *IEEE Trans. on Circuits and Systems*, vol. CAS-23, no. 10, pp. 591-598, 1976.

[5] L.J. Guibas and F.F. Yao, "On Translating a Set of Rectangles," in *Advances in Computing Research*, vol. 1, F.P. Preparata (Ed.), JAI Press Inc., pp. 61-77, 1983.

[6] F. Harary, *Graph Theory*, Addison-Wesley, Reading, MA, 1969.

[7] M.Y. Hsueh and D.O. Pederson, "Computer-Aided Layout of LSI Circuit Building-Blocks," *Proc. IEEE Int. Symp. on Circuits and Systems*, pp. 474-477, 1979.

[8] E. Lodi and L. Pagli, "A VLSI Algorithm for a Visibility Problem," in *VLSI: Algorithms and Architectures*, P. Bertolazzi and F. Luccio (Eds.), North-Holland, pp. 125-134, 1985.

[9] K. Mehlhorn and W. Rulling, "Compaction on the Torus," *VLSI Algorithms and Architectures* (Proc. AWOC '88, Corfu, Greece, 1988), J.H. Reif (Ed.), *Lecture Notes in Computer Science*, vol. 319, Springer-Verlag, pp. 212-225, 1988.

[10] J. O'Rourke, *Art Gallery Theorems and Algorithms*, Oxford University Press, 1987.

[11] I. Rival and J. Urrutia, "Representing Orders by Translating Convex Figures in the Plane," *Order*, vol. 4, pp. 319-339, 1988.

[12] P. Rosenstiehl and R.E. Tarjan, "Rectilinear Planar Layouts of Planar Graphs and Bipolar Orientations," *Discrete & Computational Geometry*, vol. 1, no. 4, pp. 342-351, 1986.

[13] M. Schlag, F. Luccio, P. Maestrini, D.T. Lee, and C.K. Wong, "A Visibility Problem in VLSI Layout Compaction," in *Advances in Computing Research, vol. 2*, F.P. Preparata (Ed.), JAI Press Inc., pp. 259-282, 1985.

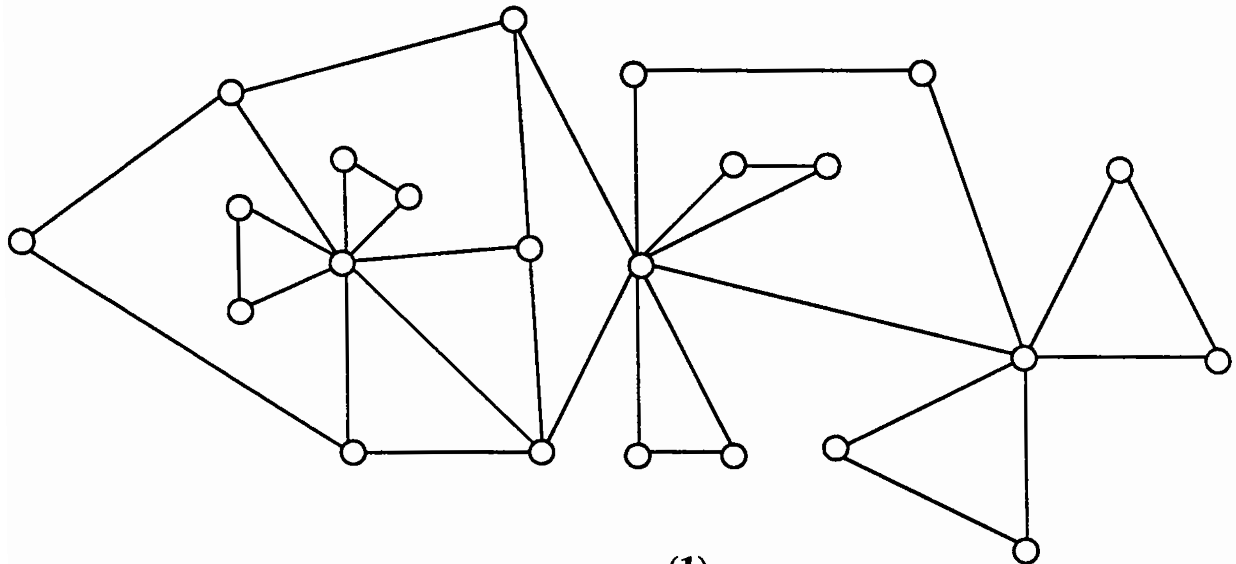
[14] R. Tamassia and I.G. Tollis, "A Unified Approach to Visibility Representations of Planar Graphs," *Discrete & Computational Geometry*, vol. 1, no. 4, pp. 321-341, 1986.

[15] R. Tamassia and I.G. Tollis, "Efficient Embedding of Planar Graphs in Linear Time," *Proc. IEEE Int. Symp. on Circuits and Systems*, Philadelphia, pp. 495-498, 1987.

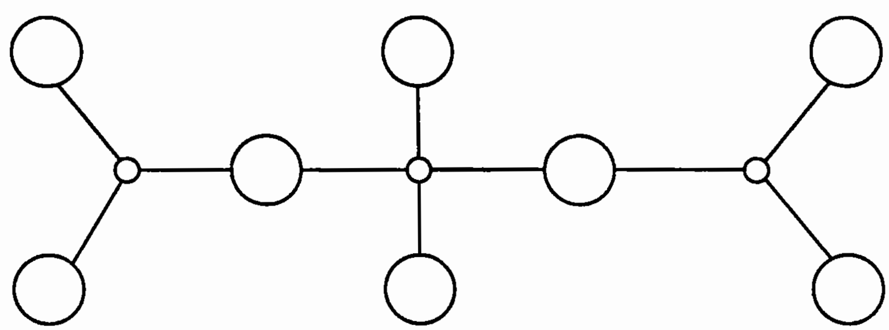
[16] C. Thomassen, "Plane Representations of Graphs," in *Progress in Graph Theory*, J.A. Bondy and U.S.R. Murty (Eds.), Academic Press, pp. 43-69, 1984.

[17] S. Wimer, I. Koren, and I. Cederbaum, "Floorplans, Planar Graphs, and Layouts," *IEEE Trans. on Circuits and Systems*, vol. 35, no. 3, pp. 267-278, 1988.

[18] S. Wismath, "Characterizing Bar Line-of-Sight Graphs," *Proc. ACM Symp. on Computational Geometry*, Baltimore, Maryland, pp. 147-152, 1985.

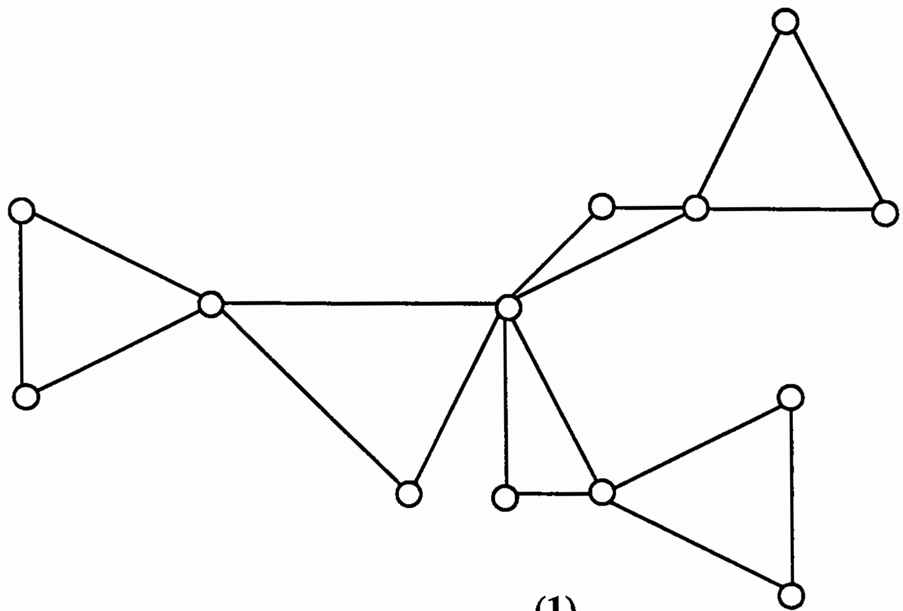


(1)

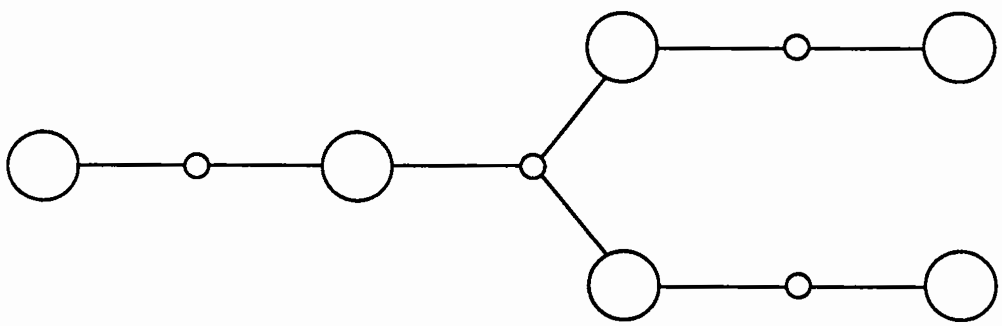


(2)

Figure 1 **(1)** A 1-connected graph and **(2)** its block-cutpoint tree.



(1)



(2)

Figure 2 (**1**) A graph that admits a visibility representation in the plane; (**2**) the block-cutpoint tree of the graph in part (1).

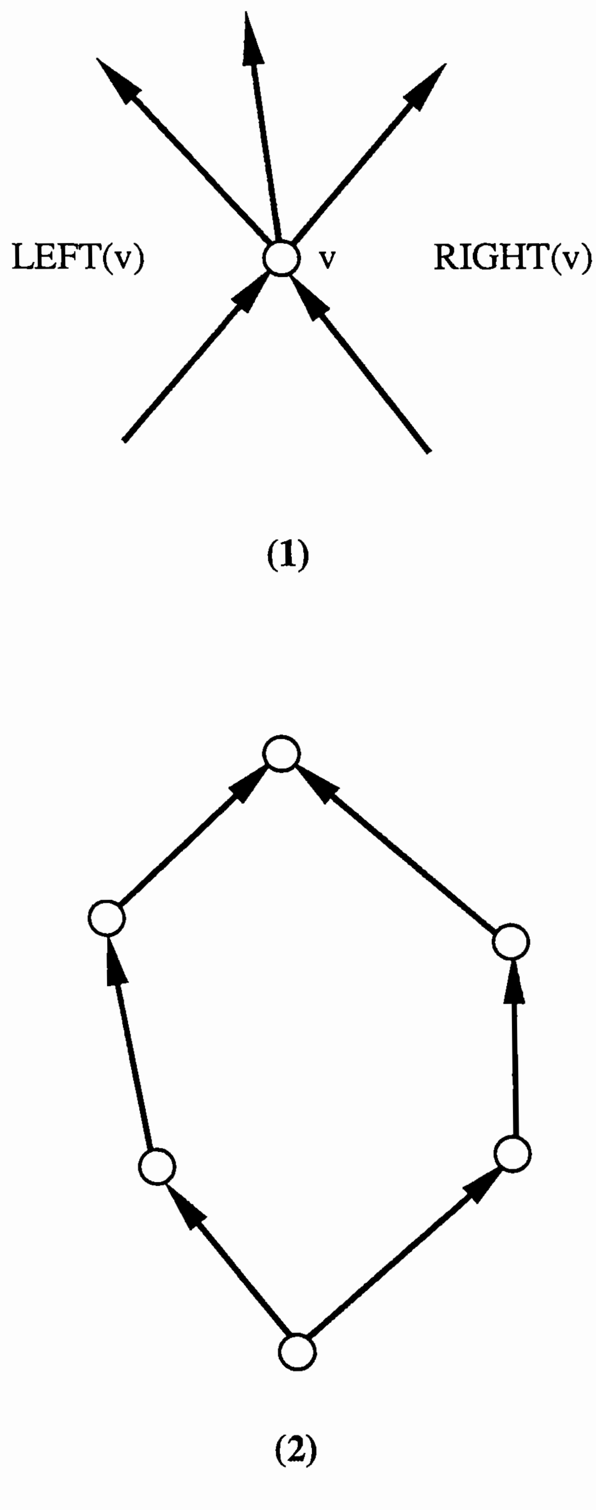


Figure 3 **(1)** Incoming and outgoing directed edge around a vertex; **(2)** directed paths forming the boundary of a face.

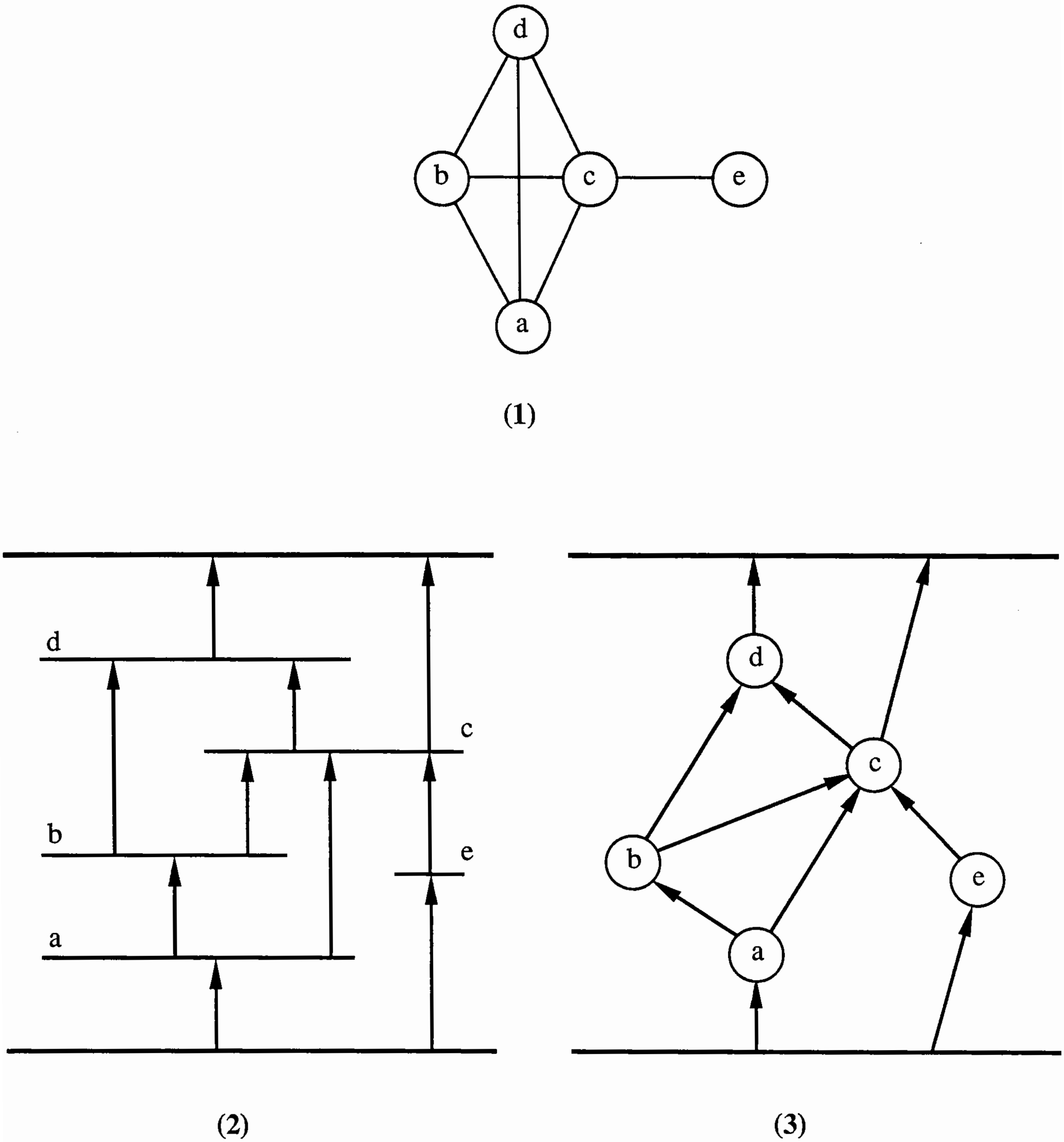


Figure 4 **(1)** A planar graph G; **(2)** a cylindric visibility representation K for G; and **(3)** the cylindric orientation associated with K.

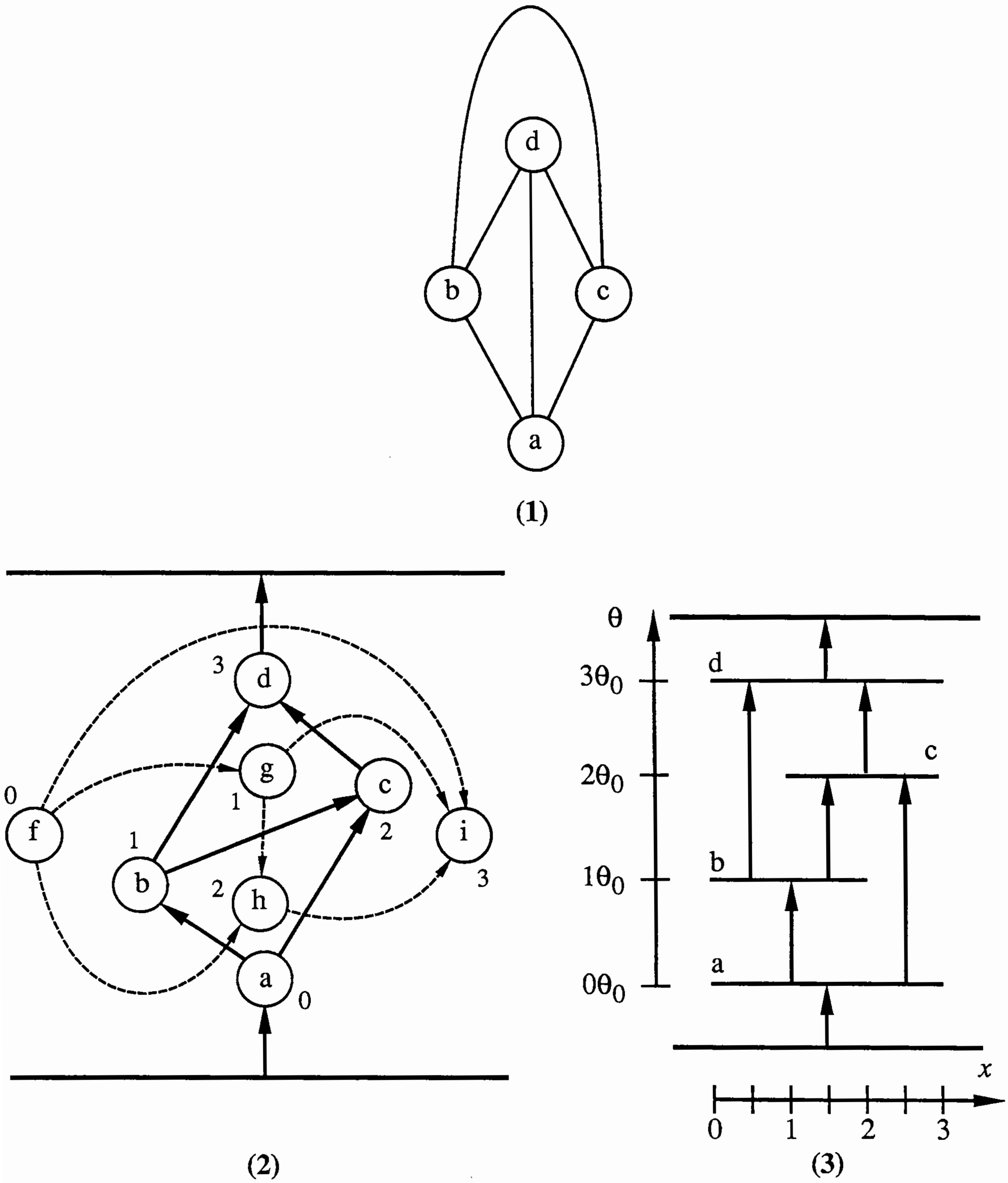


Figure 5 A running example for algorithm *VISIB*$-2C$: **(1)** a planar embedding of a 2-connected graph G; **(2)** the cylindric orientation Γ for G (in heavy lines), computed in Step 1, and its dual Γ^* computed in Step 2; **(3)** the cylindric visibility representation for G computed in Steps 3-6.

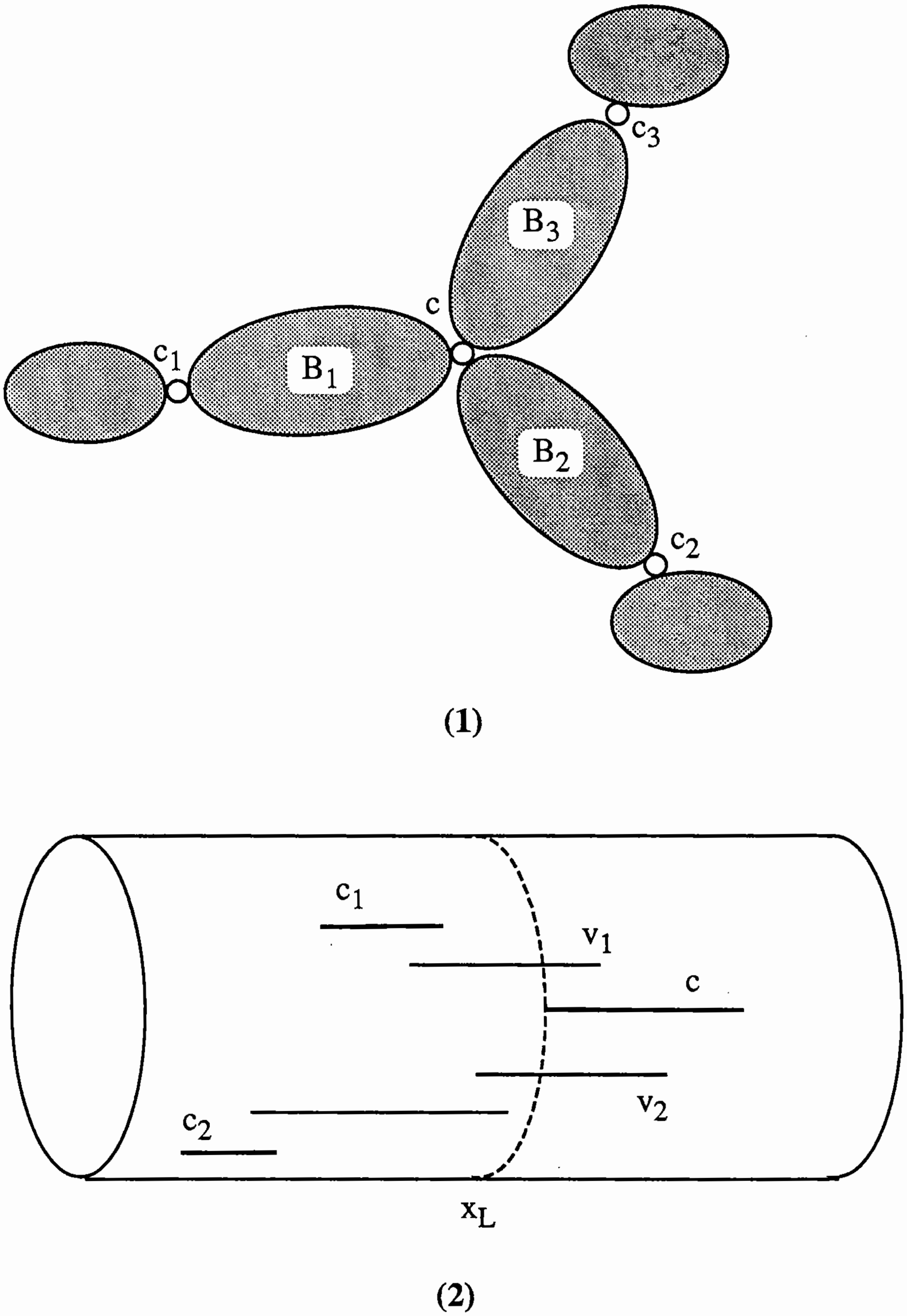
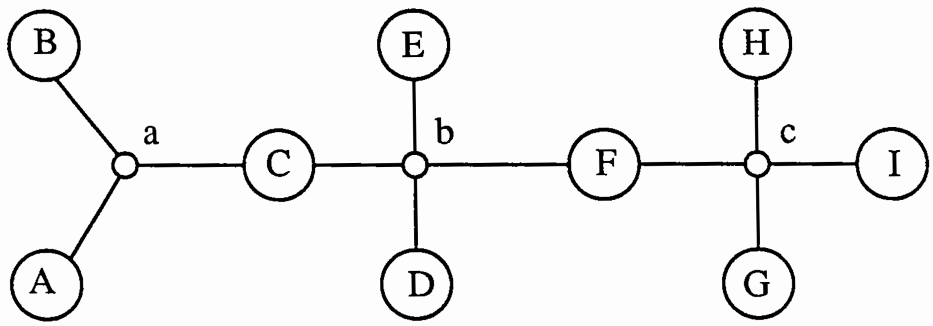
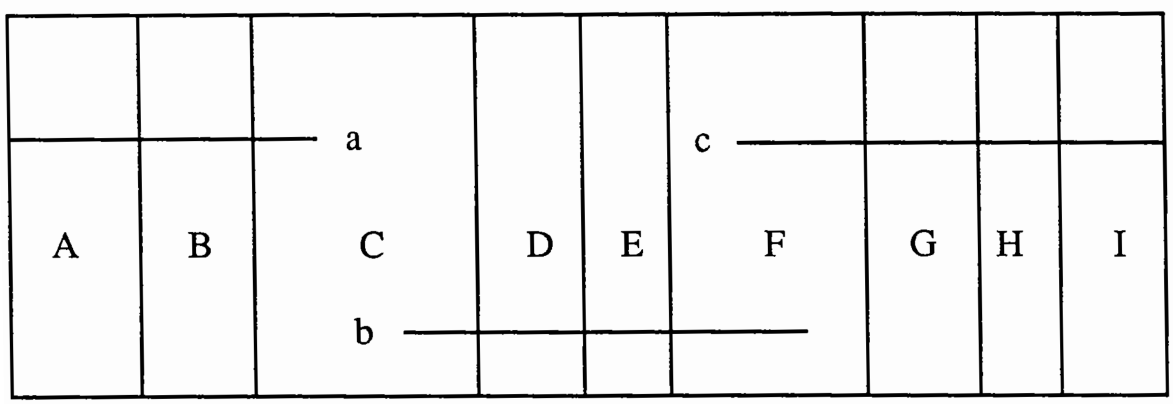


Figure 6 Example for the proof of Theorem 3 (necessity).



(1)



(2)

Figure 7 Construction in the proof of Theorem 3 (sufficiency).