DISCRETE APPLIED MATHEMATICS

ELSEVIER Discrete Applied Mathematics 54 (1994) 169–213

Obstruction set isolation for the gate matrix layout problem

Nancy G. Kinnersley[a,*,1], Michael A. Langston[b,2]

[a]*Department of Computer Science, University of Kansas, Lawrence, KS 66045-2192, USA*
[b]*Department of Computer Science, University of Tennessee, Knoxville, TN 37996-1301, USA*

Received 9 January 1991; revised 4 December 1992

Abstract

Gate matrix layout is a well-known $\mathcal{NP}$-complete problem that arises at the heart of a number of VLSI layout styles. Despite its apparent general intractability, it has recently been shown that it can be solved in $O(n^2)$ time whenever the number of tracks is fixed. Curiously, the proof of this is nonconstructive, based on finite but unknown obstruction sets. What then are such sets, and what is their underlying structure? The main result we report in this paper is a proof that the obstruction set for three tracks contains exactly 110 elements. We also describe a number of methods for obstruction identification that extend to any number of tracks.

Key words: Circuit layout; Finite-basis characterizations; Polynomial-time complexity

1. Introduction

Traditionally, decision problems[3] have been classified as either "easy" or "hard", dependent on whether low-degree polynomial-time decision algorithms exist to solve them. Until recently, one could expect any proof of easiness to be *constructive*. That is, the proof itself should provide "positive evidence" in the form of the promised polynomial-time decision algorithm. Surprising advances, however, dramatically alter this appealing picture. See, for example, [6–8] for applications of tools from [14–17] that *nonconstructively* establish the existence of low-degree polynomial-time decision algorithms for a number of challenging combinatorial problems.

*Corresponding author.

[1]This author's research has been supported in part by the University of Kansas under general research allocation 3800-20-0038 and by the National Science Foundation under grant CCR-9008725.
[2]This author's research has been supported in part by the National Science Foundation under grant MIP-8919312 and by the Office of Naval Research under contract N00014-90-J-1855.

[3]With its roots in set theory, computational complexity poses questions in terms of *decision* problems, rather than more natural *search* or *optimization* problems. Fortunately, the process of *self-reduction* often suffices to transform decision algorithms into search or optimization algorithms [3, 9, 10].

SSDI 0166-218X(93)E0099-K

In general, problems amenable to this approach are modeled as graphs. The algorithm can decide whether a given encoding of a problem is a "yes" instance or a "no" instance by determining if it contains an element of a finite basis of forbidden graphs (the obstruction set). Strikingly, the underlying theory does not tell how to identify all members of such a set, the cardinality of the set, or even the order of the largest member of the set. The only fact we are given is that the set is finite.

Perhaps the best-known example of an algorithm based on such "negative evidence" is the celebrated finite-basis characterization of planar graphs [13]: a graph is planar if and only if it contains no member of a two-element obstruction set in the topological order. The main result we present in this paper is a similar finite-basis characterization for the three-track gate matrix layout problem: a graph represents a circuit with a three-track layout if and only if it contains no member of a 110-element obstruction set in the minor order.

Interestingly, it has recently been recognized [10] that gate matrix layout with parameter k is identical to the path-width problem with parameter $k-1$. (That is, a graph G represents a circuit with a k-track layout if and only if G has a path decomposition [14] of width at most $k-1$.) Because the work we report here was originally derived in terms of circuit layout, and because gate matrix layout has received considerable attention in the literature, we shall neither state nor prove our results in terms of path-width. Instead, we only note that it is fortuitous that our efforts contribute to the understanding of this important width metric.

Our proofs are of two general types. Some describe characteristics of obstructions, and thereby help to delimit the search space. Others show how a number of obstructions can be constructively obtained. Since these techniques alone are sufficient to bound but insufficient to isolate all obstructions, many obstructions were identified with the aid of exhaustive case-checking. To assist in this heroic undertaking, massive computational power[4] was used to verify that each obstruction represents a circuit that has no three-track layout, and to check that each proper minor of each obstruction represents a circuit that does have a three-track layout.

In the next two sections, we discuss relevant background information. In Section 4, we present the notation and terminology used throughout the remainder of this paper. In Sections 5 and 6, we prove several general results and constructions that hold for any number of tracks. In Section 7, we determine some specific properties required of three-track layouts and isolate all nonouterplanar obstructions. In Section 8, we enumerate the entire three-track obstruction set and prove that this set is complete. In the final two sections, we summarize our work and pose a few related open problems.

[4]We employed the dynamic programming formulation as given in [4] and as streamlined in [12]. The algorithm's consumption of both time and space was enormous; its use was generally restricted to instances of moderate size (graphs with no more that about twenty edges) on an IBM 3090-300E.

2. The minor order

A graph H is less than or equal to a graph G in the *minor* order, written $H \leqslant_m G$, if and only if a graph isomorphic to H can be obtained from G by a series of these two operations: taking a subgraph and contracting an arbitrary edge. A family F of graphs is said to be *closed* under the minor order if the facts that G is in F and that $H \leqslant_m G$ together imply that H must be in F. The *obstruction set* for a family F of graphs is the set of graphs in the complement of F that are minimal in the minor order. Therefore, if F is closed under the minor order, it has the following characterization: G is in F if and only if $H \not\leqslant_m G$ for every H in the obstruction set for F.

Theorem 2.1 [17]. *Any set of finite graphs contains only a finite number of minor-minimal elements.*

Theorem 2.2 [16]. *For every fixed graph H, the problem that takes as input a graph G and determines whether $H \leqslant_m G$ is solvable in polynomial time.*

Theorems 2.1 and 2.2 guarantee only the existence of a polynomial-time decision algorithm for any minor-closed family of graphs. In particular, *no* proof of Theorem 2.1 can be entirely constructive [10].

Letting n denote the number of vertices in G, the time bound for algorithms ensured by these theorems is $O(n^3)$. If F excludes a planar graph, then the bound reduces to $O(n^2)$. In general, these algorithms possess enormous constants of proportionality [15], although new techniques greatly mitigate them [18], and methods specific to layout problems such as the one we address here lower them even more [10].

3. The gate matrix layout problem

Gate matrix layout is a combinatorial problem that arises in several VLSI layout styles, including gate matrix, PLAs under multiple folding, Weinberger arrays and others. It was originally posed in terms of operations on Boolean matrices. Formally, we are given an $n \times m$ Boolean matrix M and an integer k, and are asked whether we can permute the columns of M so that, if in each row we change to $*$ every 0 lying between the row's leftmost and rightmost 1, then no column contains more than k 1s and $*$s. Such a $*$ is termed a *fill-in*. We refer the interested reader to [4] for sample instances, figures and additional background on this challenging problem.

Although the general problem is $\mathcal{NP}$-complete, it has been shown that, for any fixed value of k, an arbitrary instance can be mapped to an equivalent instance with only two 1s per column, then modeled as a graph on n vertices such that the family of "yes" instances is closed under the minor order and excludes a planar graph.

Theorem 3.1 [6]. *For any fixed k, gate matrix layout can be decided in $O(n^2)$ time.*

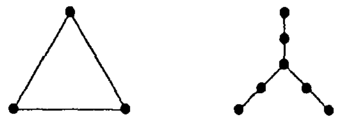


Fig. 1. Obstruction set for 2-GML.

Thanks to this mapping defined on arbitrary Boolean matrices, it suffices to restrict our attention to connected, simple graphs.

In the sequel, we shall use the term k-GML to denote the k-track variant of gate matrix layout. Thus, an obstruction for k-GML is a graph that represents a "no" instance for parameter k (it has no k-track layout) and that is minimal for parameter k (each of its proper minors does have a k-track layout). For 1-GML, it is trivial to see that the obstruction set contains only K_2. For 2-GML, the only obstructions are K_3 and $S(K_{1,3})$[5] [2] (see Fig. 1). (The connected graphs that are "yes" instances for 2-GML are known as *caterpillars*.)

4. Definitions and notation

Let G denote a graph, with vertex set V and edge set E, and let M denote an incidence matrix for G, augmented as necessary with fill-ins. For convenience, we assume a labeling for V and some appropriate bijection between these labels and the rows of M. Thus we shall, for example, refer merely to "row u" rather than to the more precise but cumbersome "row that corresponds to vertex u".

We term the matrix M a *permutation* for G, since the ordering of its columns determines an ordering for E. The *cost of a column* is the total number of 1s and fill-ins it contains. The *cost of a permutation* is the maximum cost of any of its columns. The *cost of a graph* is the minimum cost of any of its permutations. These costs represent the number of tracks required in a layout of the associated circuit.

A vertex of degree one is a *pendant vertex*. A (simple) *path* is a sequence of distinct vertices $v_1, v_2, \ldots, v_h$ such that edge $v_iv_{i+1} \in E$ for $1 \leqslant i < h$. Vertices that form such a sequence are *consecutive*. A *pendant path* is a path in which v_1 has degree three or more, v_h has degree one, and each v_i, $1 < i < h$, has degree two. The *length* of such a path is $h - 1$, the number of edges it contains.

A planar graph along with a planar embedding is called a *plane* graph. Similarly, an outerplanar graph [11] along with an outerplanar embedding is called an *outerplane* graph. The regions of the plane bounded by the embedding are called *faces*. (The unbounded region is known as the "exterior" face. Unless otherwise noted, a face is understood to mean an interior face.) Two faces in a plane graph are *edge adjacent* if

[5] $S(K_{1,3})$ is the graph obtained by subdividing each edge of $K_{1,3}$.

their intersection contains one or more edges. Two faces are *vertex adjacent* if their intersection contains one or more vertices but no edges.

Given a permutation for a graph, the *span for a vertex* is the collection of columns that contain either a 1 or a fill-in in its row. If the graph is plane, then the *span for a face* is the collection of columns that lie between the leftmost and rightmost columns that represent edges of the face, inclusive.

Finally, we assume the reader is familiar with standard graph operations, in particular subtraction ($\backslash$), union ($\cup$) and intersection ($\cap$) [1].

5. Obstruction characterization tools

In this section and the next, we shall derive[6] a number of results that help to characterize or construct obstructions. These results hold for arbitrary k.

Lemma 5.1. *No obstruction for k-GML contains two or more pendant paths of length one incident on a common vertex.*

Proof. Assume otherwise, and let G denote an obstruction for k-GML with pendant vertices v and w, both adjacent to vertex u. Let $G' = G\backslash\{w\}$. Since G is minimal for parameter k, G' possesses a permutation M' with cost at most k. (Recall that permutations are augmented only as necessary with fill-ins, and so M' has no fill-ins whatsoever in row v.) Consider the matrix M obtained from M' by adding row w and placing column uw adjacent to column uv. The cost of column uw is identical to that of column uv, because still no fill-ins are needed in row v. Moreover, the costs of all other columns remain unchanged, because no fill-ins are required anywhere in row w. Therefore, M is a permutation for G with cost at most k, contradicting our assumption that G has no k-track layout. □

Lemma 5.2. *No obstruction for k-GML contains a pendant path of length greater than two.*

Proof. Assume otherwise, and let G denote an obstruction for k-GML with pendant path $x, \ldots, u, v, w$ of length three or more. Let $G' = G\backslash\{w\}$. Because G is minimal and because $G' <_m G$, there is an optimal permutation M' for G' with cost at most k. Since u has degree two, we may assume by symmetry that column uv contains the rightmost 1 in row u. Consider the matrix M obtained from M' by adding row w and inserting column vw to the immediate right of column uv. Since column vw does not need

[6] As a matter of style, we shall omit proofs when they follow immediately from previous results, and shall merely point the reader to an earlier proof when analogous arguments suffice. We realize that the responsibility for deciding when to suppress details in this presentation is ours alone, and remark that full proofs, even of the corollaries, can be found in [12].

a fill-in in row u, its cost is the same as that of column uv. The costs of all other columns remain unchanged, because no fill-ins are required in row v or in row w. Therefore, M is a permutation for G with cost at most k, contradicting the assumption that G has no k-track layout. $\square$

Thus, a pendant vertex is an endpoint of either a pendant path of length one (which we shall henceforth call a *pendant edge*) or a pendant path of length two (which we shall without ambiguity henceforth term simply a *pendant path*, omitting reference to its length).

Lemma 5.3. *If a graph has a pendant path, then there is an optimal permutation for that graph in which the edges of the path are represented by adjacent columns.*

Proof. Let G denote a graph with pendant path u, v, w, and let M denote an optimal permutation for G. Suppose that columns uv and vw are not adjacent, and that column uv is to the left of column vw. The rightmost 1 in row u must be either (1) in column uv, (2) in a column between columns uv and vw, or (3) in a column to the right of column vw.

Suppose (1) holds. We construct a new matrix M' from M by moving column vw to the left until it is to the immediate right of column uv. Since column vw does not require a fill-in in row u, its cost is no more than that of column uv. Moreover, no column now requires a fill-in in row v, and the cost of M' is no more than that of M.

Suppose (2) holds. For the sake of discussion, assume that the rightmost 1 in in row u is in column ux. We construct matrix M' from M by first moving column uv to the right until it is to the immediate right of column ux. Since column ux had a fill-in in row v, the cost of column uv is no more than the original cost of column ux. To complete the construction of M', move column vw to the left until it is to the immediate right of column uv. Since column vw does not require a fill-in in row u, its cost is no more than that of column uv. Therefore, the cost of M' is no greater than that of M.

Suppose (3) holds. We construct matrix M' from M by moving column uv to the right until it is to the immediate left of column vw. Since column vw has a fill-in in row u, the cost of column uv is no more than the cost of column vw. Thus, the cost of M' cannot exceed that of M. $\square$

Lemma 5.4. *No obstruction for k-GML contains more than three pendant paths incident on a common vertex.*

Proof. Assume otherwise, and let G denote an obstruction for k-GML with four or more pendant paths incident on vertex u. Let u, v, w be one such pendant path, and let $G' = G \backslash \{uv, vw\}$. Because G is minimal and because $G' \leqslant_m G$, G' possesses a permutation M' with cost at most k in which (due to Lemma 5.3) each pendant path incident on u is represented by a pair of adjacent columns. Let the second such pair of columns represent pendant path u, x, y. (We choose the second pair of columns since this guarantees that column xy has a fill-in in row u.) We construct matrix M from M' by adding rows v and w, and by placing columns uv and vw to the immediate right of columns ux and xy. Since no fill-ins are required in rows x and y, the costs of columns uv and vw are the same as the costs of columns ux and xy, respectively. Therefore, M is a permutation for G with cost at most k, contradicting our assumption that G has no k-track layout. □

Lemma 5.5. *For $k > 2$, no obstruction for k-GML contains more than two consecutive vertices of degree two.*

Proof. Assume otherwise, and let G denote an obstruction for k-GML, $k > 2$, with consecutive vertices u, v and w, each of degree two. Let G' be the graph obtained from G by contracting the edge uv to u. (Observe that u and w each retain degree two in G': no increase in degree is possible; a decrease would imply either that G is K_3 and hence not an obstruction for $k > 2$, or that G is not connected and hence not an obstruction for any k.) Because G is minimal and because $G' \leqslant_m G$, G' must possess an optimal permutation M' with cost at most k. From the facts that M' has no unnecessary fill-ins in rows u and w and that both u and w have degree two, it follows that either (1) the spans for these two rows overlap only in column uw or (2) the span for one properly contains the span for the other.

Suppose (1) holds. For the sake of discussion, assume the single column of overlap (column uw) contains the rightmost 1 in row u and thus the leftmost 1 in row w. We construct at no extra cost a new matrix M from M', by replacing column uw with columns uv and vw.

Suppose (2) holds. For the sake of discussion, assume the span for u properly contains the span for w, with column uw the rightmost in both spans. Let c denote the column that contains the leftmost 1 in row w. We construct at no extra cost a new matrix M from M', by replacing column uw with column vw, and by inserting column uv to the immediate left of column c.

In either case, M is a permutation for G with cost at most k, contradicting the assumption that G has no k-track layout. □

Lemma 5.6. *Suppose G contains a pair of adjacent vertices, u and v, each of degree two. If G' is obtained from G by contracting the edge uv to u, adding a new vertex w, and adding the edge uw, then the cost of G equals that of G'.*

Proof. Let G, G', u, v and w be as defined in the statement of the lemma. Let t (x) denote the other vertex adjacent to u (v) in G. Let M denote an optimal permutation for G.

We construct a new matrix M' from M by replacing column vx with column ux and changing the label on row v to w. Row w contains a single 1 and requires no fill-ins. Any column that now needs a new fill-in in row u originally had a fill-in in row v. Thus, the cost of M' is no more than that of M. Because M' is a permutation for G', the cost of G' cannot exceed that of G.

Let M' denote an optimal permutation for G'. Note that, in G', vertex u has degree three and is adjacent to vertices t, w, and x. It suffices to consider two cases for M', in that either (1) column uw lies between columns tu and ux or (2) column uw contains the leftmost 1 in row u.

Suppose (1) holds. We construct a new matrix M from M' by replacing column ux with column wx and changing the label on row w to v. Any fill-ins required in row v lie in columns that no longer require fill-ins in row u. Thus, the cost of M is no more than that of M'.

Suppose (2) holds. If column uw has a fill-in in row t (or row x) then, at no extra cost, we move column tu (column ux) to the immediate left of column uw. M can now be constructed as in (1). If column uw has 0s in both row t and row x, then we may assume that columns tu and ux contain the leftmost 1s in rows t and x, respectively. (To see this, note that if another column c holds the leftmost 1 in row t (row x), then c has a fill-in in row u and column tu (column ux) can be placed to the left of c with no increase in cost). If column ux is to the right of column tu then, at no extra cost, we move column uw to the immediate left of column ux. Otherwise, at no extra cost, we move column uw to the immediate left of column tu. M can now be constructed as in (1).

Because M is a permutation for G, the cost of G cannot exceed that of G'. □

Corollary 5.7. *If G and G' denote graphs as defined in Lemma 5.6, then G is an obstruction for k-GML if and only if G' is.*

Corollary 5.8. *No obstruction for k-GML contains two adjacent vertices of degree three each adjacent to a pendant vertex as well.*

Corollary 5.9. *No obstruction for k-GML contains a vertex of degree three adjacent to both a pendant vertex and a vertex of degree two.*

Lemma 5.10. *Let G denote an arbitrary graph with cost k, and let v denote any vertex of G. G possesses an optimal permutation in which every column with cost k has a nonzero entry in row v if and only if $G\backslash\{v\}$ does not contain an obstruction for $(k-1)$-GML.*

Proof. Let G denote a graph with cost k and let v denote any vertex of G. If $G\backslash\{v\}$ contains an obstruction for $(k-1)$-GML, then every optimal permutation for $G\backslash\{v\}$ has cost k. Therefore, every optimal permutation for G contains a column with cost k that has a 0 in row v.

If $G\backslash\{v\}$ does not contain an obstruction for $(k-1)$-GML, then $G\backslash\{v\}$ possesses an optimal permutation M' with cost at most $k-1$. Consider the matrix M obtained from M' by adding row v and, for each vertex w adjacent to v in G, inserting column vw adjacent to any column with a 1 in row w. In every case, the cost of column vw is at most k. Since v is the only row that may need additional fill-ins, M is an optimal permutation for G of cost k in which every column with cost k has a nonzero entry in row v. □

Corollary 5.11. *Let G denote an obstruction for k-GML and let v denote a vertex of G. G has cost exactly $k + 1$, and possesses an optimal permutation in which every column of cost $k + 1$ has a 1 or a fill-in in row v.*

Lemma 5.12. *Adding an edge to a graph increases its cost by at most one.*

Proof. Straightforward. □

Lemma 5.13. *If G contains K_k as a subgraph, then G has cost at least k and possesses an optimal permutation in which the edges of K_k are represented in adjacent columns.*

Proof. Follows immediately from Lemma 4.1 of [6]. □

Corollary 5.14. *If G contains K_k as a minor, then G has cost at least k.*

Corollary 5.15. *K_k is an obstruction for $(k - 1)$-GML.*

Lemma 5.16. *Let G denote an arbitrary graph and let H_1 and H_2 denote obstructions for k-GML. If $G \cap H_1 = G \cap H_2 = \{v\}$ for some vertex v, then the cost of $G \cup H_1$ equals that of $G \cup H_2$.*

Proof. Let G, H_1, H_2 and v be as defined in the statement of the lemma. Let M denote an optimal permutation for $G \cup H_1$.

Since H_1 is an obstruction for k-GML, some column c of H_1 in M has cost $k + 1$ in the rows of H_1. Either c is contained in the span for v (in which case c contains a 1 or a fill-in in row v), or else the connectedness of H_1 ensures that every column of G lying between c and the span for v has a fill-in in some other row of H_1.

Due to Corollary 5.11, H_2 possesses a cost $k + 1$ permutation M_2 in which every column with cost $k + 1$ has a 1 or a fill-in in row v. We use M_2 to construct a new matrix M' from M as follows. We first eliminate the rows of $H_1 \backslash \{v\}$, then all resultant columns with at most one 1 (one of which is c). We next insert M_2 into the position formerly occupied by c (which requires a new row for each vertex of $H_2 \backslash \{v\}$). No inserted column can require more fill-ins than did c. Due to the way c was chosen and its relation to the span for v, no column originally in M can incur an increase in its number of fill-ins. Thus, the cost of M' is no more than that of M.

Because M' is a permutation for $G \cup H_2$, the cost of $G \cup H_2$ cannot exceed that of $G \cup H_1$. The inequality is established in the reverse direction analogously. □

Corollary 5.17. *If G, H_1 and H_2 denote graphs as defined in Lemma 5.16, and if $G \cup H_1$ is an obstruction for k'-GML but $G \cup H_2$ is not, then any obstruction for k'-GML contained as a minor in $G \cup H_2$ has the form $G \cup H_2'$ for some $H_2' <_m H_2$.*

Lemma 5.18. *Let G be a plane graph with face F. In any permutation for G, every column in the face span for F has a cost of at least two in the collection of rows that*

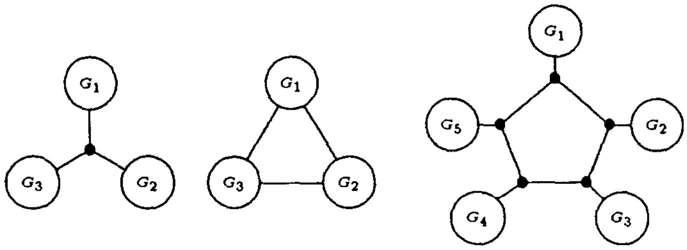


Fig. 2. Constructions used in Lemmas 6.1, 6.2 and 6.4.

correspond to the vertices of F, and every interior column of that span has a total cost of at least three.

Proof. Straightforward. □

6. Obstruction construction tools

The constructions studied in this section are depicted informally in Fig. 2.

Lemma 6.1. *Let G_1, G_2 and G_3 denote disjoint (but not necessarily distinct) obstructions for k-GML, let v_i denote an arbitrary vertex of G_i for $1 \leqslant i \leqslant 3$, and let v denote an isolated vertex not in $G_1 \cup G_2 \cup G_3$. The graph $G = G_1 \cup G_2 \cup G_3 \cup \{v\} \cup \{vv_i$: $1 \leqslant i \leqslant 3\}$ is an obstruction for $(k+1)$-GML.*

Proof. Let G_i, v_i, v and G be as defined in the statement of the lemma. It follows from Lemma 4.3 of [6] that G has cost $k + 2$.

We now establish the minimality of G. Due to Corollary 5.11, each G_i, $1 \leqslant i \leqslant 3$ possesses a cost $k + 1$ permutation M_i in which every column with cost $k + 1$ has a 1 or a fill-in in row v_i. Since G is connected, the removal of a vertex necessarily means the removal of an edge and, therefore, we only need consider the effect of removing or contracting a single edge, e. Because of G's symmetry, we may assume that either (1) e is in G_1, or (2) $e = vv_1$.

Suppose (1) holds. Let G_1' and G' denote the minors of G_1 and G, respectively, that are obtained by the removal or contraction of e (for notational simplicity in the case of a contraction, we insist that e be contracted to v_1 if $v_1 \in e$). Because G_1 is minimal for parameter k, G_1' possesses a permutation M_1' with cost at most k. Let M denote the permutation M_2, vv_2, M_1', vv_3, M_3. A column in M_2 or M_3 has cost at most $k + 1$. Columns vv_2 and vv_3 each have cost two. Any column in M_1' incurs one additional fill-in (in row v), bringing its cost to at most $k + 1$. Thus, M has cost $k + 1$. We form M' from M at no extra cost by placing vv_1 adjacent to an arbitrary column in M with a 1 in row v_1.

Suppose (2) holds. If $G' = G\backslash\{e\}$, let M' denote the permutation $M_2, vv_2, vv_3, M_3, M_1$. Since v now has degree two, no fill-ins are required in its row. Because of the way M_2 and M_3 were chosen, any column that requires a new fill-in in row v_2 or row v_3 has cost at most $k+1$, and so M' has cost $k+1$. If G' is obtained from G by contracting e to v_1, let M' denote the permutation $M_2, v_1v_2, M_1, v_1v_3, M_3$. Only the v_i rows may require additional fill-ins. Again, because of the way each M_i was chosen, any new fill-in must lie in a column that has cost at most $k+1$, and so M' has cost $k+1$.

In any case, M' is a permutation for G', and thus the cost of G' is strictly less than that of G. □

Lemma 6.2. *Let G_1, G_2, and G_3 denote disjoint (but not necessarily distinct) graphs of cost k, and let u_i and v_i denote arbitrary (but not necessarily distinct) vertices of G_i for $1 \leqslant i \leqslant 3$. The graph $G = G_1 \cup G_2 \cup G_3 \cup \{v_1v_2, u_1v_3, u_2u_3\}$ has cost at least $k+1$.*

Proof. Let G_i, u_i, v_i and G be as defined in the statement of the lemma. Let M denote an optimal permutation for G and, in M, let c_i denote a column of G_i with cost at least k in the rows of G_i. Without loss of generality, assume c_1 lies to the left of c_2 which lies to the left of c_3. *If* u_1v_3 lies to the left of c_2, then c_2 has a fill-in in some row of G_3. Otherwise, it has a fill-in in some row of G_1. Thus the cost of G is at least $k+1$. □

The graph G just defined may not, however, have cost exactly $k+1$, even if $u_i = v_i$, $1 \leqslant i \leqslant 3$. An example is illustrated in Fig. 3.

Corollary 6.3. *Let G_1, G_2, and G_3 denote obstructions for $(k-1)$-GML, and let v_i denote an arbitrary vertex of G_i for $1 \leqslant i \leqslant 3$. The graph $G = G_1 \cup G_2 \cup G_3 \cup \{v_1v_2, v_1v_3, v_2v_3\}$ has cost exactly $k+1$.*

The graph G just defined may not, however, be an obstruction for k-GML. For example, obstruction 12.3.1 listed in the appendix is properly contained in the graph constructed by setting $G_1 = G_2 = K_3$ and setting $G_3 = S(K_{1,3})$, with v_3 a vertex of degree one.

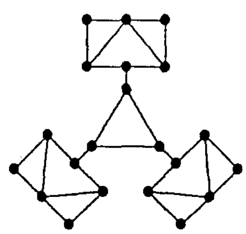


Fig. 3. A graph of cost 5 built from three graphs of cost 3.

Lemma 6.4. *Let G_1, G_2, G_3, G_4 and G_5 denote disjoint (but not necessarily distinct) obstructions for k-*GML*, and let u_i denote an arbitrary vertex of G_i for $1 \leqslant i \leqslant 5$. Let C_5 denote a cycle graph of order five, with vertex set $\{v_i\colon 1 \leqslant i \leqslant 5\}$, disjoint from $G_1 \cup G_2 \cup G_3 \cup G_4 \cup G_5$. The graph $G = G_1 \cup G_2 \cup G_3 \cup G_4 \cup G_5 \cup \{u_iv_i\colon 1 \leqslant i \leqslant 5\}$ is an obstruction for $(k+2)$-*GML.

Proof. Let G_i, u_i, C_5, v_i and G be as defined in the statement of the lemma. Let M denote an optimal permutation for G and, in M, let c_i denote a column of G_i with cost at least $k+1$ in the rows of G_i. If any c_i lies between the leftmost and rightmost columns of C_5, then it incurs at least two additional fill-ins (in rows of C_5). Otherwise, without loss of generality, assume c_1 lies to the left of c_2 which lies to the left of c_3 which lies to the left of the leftmost column of C_5. In this event, c_3 incurs two additional fill-ins (one in a row of $G_1 \cup \{v_1\}$, and one in a row of $G_2 \cup \{v_2\}$). Thus the cost of G is at least $k+3$.

Letting M_i denote a cost $k+1$ permutation for G_i in which every column with cost $k+1$ has a 1 or a fill-in in row u_i, we observe that G has cost exactly $k+3$ as evidenced by the permutation $M_1, u_1v_1, M_2, u_2v_2, v_1v_2, v_2v_3, u_3v_3, M_3, v_3v_4, v_4v_5, v_1v_5, u_4v_4, M_4, u_5v_5, M_5$.

We now establish the minimality of G. As in the proof of Lemma 6.1, we need only consider the effect of removing or contracting a single edge, e, and may assume that either (1) e is in G_1, (2) $e = u_1v_1$, or (3) $e = v_1v_2$.

Suppose (1) holds. Let G_1' and G' denote the minors of G_1 and G, respectively, that are obtained by the removal or contraction of e (if a contraction, e is contracted to u_1 if $u_1 \in e$). Because G_1 is minimal for parameter k, G_1' possesses a permutation M_1' with cost at most k. Let M_1'' denote the matrix formed at no extra cost from M_1' by adding row v_1 and placing column u_1v_1 adjacent to an arbitrary column with a 1 in row u_1. Let M' denote the permutation $M_2, u_2v_2, M_3, u_3v_3, v_2v_3, v_1v_2, M_1'', v_3v_4, v_1v_5, v_4v_5, u_4v_4, M_4, u_5v_5, M_5$, which has cost at most $k+2$.

Suppose (2) holds. If $G' = G\backslash\{e\}$, let M' denote the permutation $M_2, u_2v_2, M_3, u_3v_3, v_2v_3, v_1v_2, v_3v_4, v_4v_5, v_1v_5, u_4v_4, M_4, u_5v_5, M_5, M_1$, which has cost at most $k+2$. If G' is obtained from G by contracting e to u_1, let M' denote the permutation $M_2, u_2v_2, M_3, u_3v_3, v_2v_3, v_2u_1, M_1, u_1v_5, v_4v_5, v_3v_4, u_4v_4, M_4, u_5v_5, M_5$, which has cost at most $k+2$.

Suppose (3) holds. If $G' = G\backslash\{e\}$, let M' denote the permutation $M_2, u_2v_2, v_2v_3, u_3v_3, M_3, v_3v_4, u_4v_4, M_4, v_4v_5, u_5v_5, M_5, v_5v_1, u_1v_1, M_1$, which has cost at most $k+2$. If G' is obtained from G by contracting e to v_1, let M' denote the permutation $M_1, u_1v_1, M_2, u_2v_1, M_3, u_3v_3, v_1v_3, v_3v_4, v_1v_5, v_4v_5, u_4v_4, M_4, u_5v_5, M_5$, which has cost at most $k+2$.

In any case, M' is a permutation for G', and thus the cost of G' is strictly less than that of G. □

7. Special tools for three-track obstructions

Unlike the work of the last two sections, the results we now derive hold only for $k = 3$.

7.1. General properties of the three-track obstructions

Lemma 7.1. *No obstruction for* 3-GML *contains a vertex of degree four or more adjacent to a pendant vertex.*

Proof. Assume otherwise, and let G denote an obstruction for 3-GML with vertex v adjacent to vertices w, x, y and pendant vertex z. Let $G' = G\backslash\{z\}$, and let M' denote a cost-three permutation for G'. Without loss of generality, assume that column vw lies to the left of both vx and vy, and that column vw has a 0 in row x. Let c denote the column that contains the leftmost 1 in row x. We construct matrix M from M' by adding row z and placing column vz to the immediate left of c. M is a permutation for G with cost at most three, contradicting our assumption that G has no three-track layout. □

This result (aided by the corollaries to Lemma 5.6) is easily extended.

Corollary 7.2. *No obstruction for* 3-GML *contains two adjacent vertices each adjacent to a pendant vertex.*

Given a permutation for a plane graph, the *overlap* of two or more face spans is the collection of columns common to all spans.

Lemma 7.3. *If a plane graph of cost three contains two faces whose intersection is exactly one vertex, then it possesses an optimal permutation in which the overlap of the spans for these faces is empty.*

Proof. Let G denote a plane graph of cost three with faces F_1 and F_2 such that $F_1 \cap F_2 = v$. Let M denote a cost-three permutation for G, and suppose the overlap of the face spans for F_1 and F_2 is nonempty. Because these faces are not edge adjacent, their overlap contains at least two columns, each with cost two in the rows of F_1, and each with cost two in the rows of F_2 (Lemma 5.18). Since M has cost three, and since v is the only vertex on both F_1 and F_2, each column of the overlap represents an edge of either F_1 or F_2 that is incident on v.

Without loss of generality, assume the leftmost column of the overlap is vw of F_2, with a fill-in in row u of F_1. Since the cost of M is three, the column to the immediate right of vw must be uv. If vx of F_1 is to the right of uv, then vw requires a fill-in in row x as well, contradicting the fact that M has cost three. Therefore, the overlap contains only vw and uv, and interchanging the two columns yields a cost-three permutation for G with the desired property. □

Lemma 7.4. *If a plane graph of cost three contains two faces whose intersection is exactly one edge, then it possesses an optimal permutation in which the overlap of the spans for these faces is exactly one column.*

Proof. Let G denote a plane graph of cost three with faces F_1 and F_2 such that $F_1 \cap F_2 = uv$. Given a cost-three permutation for G, suppose the overlap of the face spans for F_1 and F_2 contains two or more columns (it cannot be empty because it must contain uv). Moreover, suppose the overlap contains no pendant edges incident on u or v (any such edge can be removed initially, then reinserted after our forthcoming permutation modification at no extra cost).

Without loss of generality, assume that the rightmost column of F_1 lies to the right of both uv and the leftmost column of F_2. It is straightforward to verify that the overlap contains at most three columns, that uv and the leftmost column of F_2 are the same, and that the column to the immediate right of uv must have the form uw (or vw) for some $w \in F_1$. Thus column uv must have a fill-in in row w, uw (or vw) must have a fill-in in row v (or u), and so uv and uw (or vw) can be interchanged at no extra cost, an action which eliminates a column from the overlap. At most one more application of this interchange reduces the overlap to uv alone. □

7.2. Three-track obstructions that are not outerplanar

Since K_4, an obstruction for 3-GML, is a minor of both K_5 and $K_{3,3}$, all obstructions for 3-GML are planar. We now establish that K_4 and the four graphs illustrated in Fig. 4 are the only obstructions for 3-GML that are not outerplanar.

Lemma 7.5. *The four graphs depicted in Fig.* 4 *are the only obstructions for* 3-GML *with the property that, for any planar embedding, there exists an edge not adjacent to the exterior face.*

Proof. Computation suffices to check that these four graphs are indeed obstructions for 3-GML; clearly, each has the property stated in the lemma. Thus we need only to establish that these are the only obstructions for 3-GML that possess this property.

Let $G = \langle V, E \rangle$ denote an arbitrary plane obstruction for 3-GML with the desired property, and assume without loss of generality that its embedding maximizes the number of edges on or adjacent to the exterior face. Let V_f denote the set of vertices on this exterior face, and let V_n denote $V \setminus V_f$. Let G' denote the subgraph of G induced by V_n. Thus G' contains at least one edge, uv.

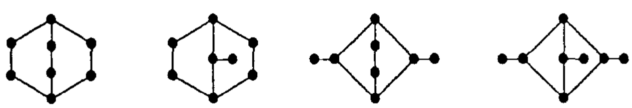


Fig. 4. Four related obstructions.

Let S denote the set of (simple) paths in G with an initial vertex in $\{u, v\}$, internal vertices in V_n, and a terminal vertex in V_f. If three or more distinct terminals are contained in the elements of S, then $G >_m K_4$, contradicting the presumed minimality of G. If every element of S contains the same terminal, then the connected component of G' containing uv can be moved to the exterior face, contradicting the presumed maximality of (the number of edges on or adjacent to) that face. Thus the elements of S contain exactly two different terminals, which we denote by w and x.

It now follows that G contains three vertex-disjoint paths from w to x. Moreover, the maximality of the exterior face dictates that each path either has length at least three, or contains an internal vertex adjacent to a distinct, additional vertex not on any of the three paths. Therefore $G \geqslant_m H$ for some H depicted in Fig. 4. $\square$

Lemma 7.6. *No obstruction for* 3-GML *contains a vertex of degree two adjacent to vertices of degree three or more unless those vertices are also adjacent.*

Proof. Assume otherwise, and let G denote a plane obstruction for 3-GML with degree two vertex v adjacent to vertices u and w, each of degree three or more, but not adjacent to each other. Lemma 7.1 and Corollary 5.9 guarantee that neither u nor w is adjacent to a pendant vertex. Let G' denote the minor of G obtained by contracting edge uv to u, and let M' denote a cost-three permutation for G'. Consider the overlap of the spans for u and w, and without loss of generality, assume the leftmost column is uw and that it contains the leftmost 1 in row w. If the overlap is uw, or if uw has cost two, adding row v and replacing uw with uv and vw produces a cost-three permutation for G, a contradiction. If uw has a fill-in in row x, it is straightforward to verify that some column of the overlap contains the rightmost 1 in row x, or that the overlap contains at most three columns one of which is ux. In either case, a cost-three permutation for G can be constructed from M', again contradicting the assumption that G has no three-track layout. Therefore, an obstruction for 3-GML contains a vertex of degree two adjacent to vertices of degree three or more, only if (as obstruction 6.4.1 in the appendix illustrates) the three vertices are pairwise adjacent. $\square$

Lemma 7.7. *K_4 and the graphs depicted in Fig.* 4 *are the only obstructions for* 3-GML *that are not outerplanar.*

Proof. Assume otherwise. Let G denote a nonouterplanar obstruction for 3-GML other than one of the five noted in the statement of the lemma. Thus, due to Lemma 7.5, there is at least one embedding of G in which every edge is adjacent to the exterior face. From the embeddings of G with this property, select one that maximizes the number of vertices on the exterior face, and let v denote a vertex that is not on this face. It must be that v has degree two, since otherwise $G \geqslant_m K_4$ due to the way the embedding was chosen. Let u and w denote the vertices adjacent to v. The maximality of the embedding ensures that G contains three edge-disjoint paths of length two or more between u and w. Moreover, Lemma 7.6 implies that $uw \in G$.

Consider this embedding restricted to $G' = G\backslash\{v\}$. There are faces F_1 and F_2 in G' such that $F_1 \cap F_2 = uw$. Let M' denote a cost-three permutation for G' in which, due to Lemma 7.4, the overlap of the face spans for F_1 and F_2 is uw.

If uw contains no fill-in, then we construct a new matrix M from M' by adding row v and placing columns uv and vw to the immediate left of uw.

If uw contains a fill-in in some row x, then it follows that ux and wx must exist, contain the only 1s in row x, and lie immediately to each side of uw. In this case, we construct a new matrix M from M' by adding row v and placing columns uv and vw to the immediate left of the column to the immediate left of uw.

In either case, M is a cost-three permutation for G, contradicting the assumption that G is an obstruction for 3-GML. □

7.3. Additional properties of three-track obstructions

We shall henceforth consider only outerplane obstructions and outerplanar embeddings in which all vertices lie on the exterior face. Thus the intersection of two faces is at most a single edge.

Lemma 7.8. *If an obstruction for* 3-GML *contains two faces that are adjacent at and only connected through a single vertex, then at least one of these faces is a triangle with two vertices of degree two.*

Proof. Let G denote an obstruction, with faces F_1 and F_2 adjacent at and only connected through v. Assume neither F_1 nor F_2 is a triangle with two vertices of degree two.

Let C_1 denote the (unique) connected component of $G\backslash\{v\}$ that contains an edge of $F_1\backslash\{v\}$. Let C_2 denote $(G\backslash\{v\})\backslash C_1$. Let u and w denote a pair of isolated vertices not in G. We define $G_1 = (G\backslash C_2) \cup \{u, w\} \cup \{uv, vw, uw\}$ and $G_2 = (G\backslash C_1) \cup \{u, w\} \cup \{uv, vw, uw\}$.

Observe that both G_1 and G_2 are proper minors of G and both, therefore, have cost-three permutations. It is straightforward to show that G_1 must possess an optimal permutation M_1 with the three columns of $\{u, v, w\}$ on the extreme right, else G properly contains an obstruction as described in Lemma 6.1. Similarly, G_2 must possess an optimal permutation M_2 with the three columns of $\{u, v, w\}$ on the extreme left.

But this means that we can construct a cost-three permutation for G by placing M_2 to the right of M_1 and removing the (six) columns of $\{u, v, w\}$. This contradicts the fact that G is an obstruction, however, and so the assumption that neither F_1 nor F_2 is a triangle with two vertices of degree two cannot hold. □

Let v denote a vertex on a face of an outerplane graph G. If the connected component of $G\backslash\{vw \mid w \text{ lies on a face}\}$ that contains v has at least one edge, then we term this component the *attachment* at v.

Lemma 7.9. *If an obstruction for* 3-GML *contains a face in which two or more vertices have attachments, then each attachment is a minor of* $S(K_{1,3})$.

Proof. Assume otherwise for obstruction G, in which vertices u and v of face F have attachments, with the attachment at v, $A(v)$, not a minor of $S(K_{1,3})$.

No vertex of $A(v)$ has degree greater than three unless $A(v)$ contains a cycle (Lemmas 5.4 and 7.1). No degree-three vertex of $A(v)$ is adjacent to both a vertex of degree two and a pendant vertex unless that vertex is v (Corollary 5.9). No degree-two vertex of $A(v)$ is adjacent to two vertices of degree three (Lemma 7.6). It follows that either $A(v)$ contains a cycle or $S(K_{1,3}) <_m A(v)$, and thus $A(v)$ has cost three.

Let $A^+ = A(v) \cup \{u, w\} \cup \{uv, vw, uw\}$. Let $A^- = A(v)\backslash\{v\}$. It is straightforward to show that A^+ possesses an optimal permutation M_1 in which uvw[7] is the rightmost column. Let $G' = (G\backslash A^-) \cup \{x, y\} \cup \{xv, vy, xy\}$. If $A(v)$ contains a cycle, then $G' <_m G$, and thus G' has cost three. If $A(v)$ is acyclic, then $S(K_{1,3}) <_m A(v)$, and thus (with the help of Lemma 5.16) again G' has cost three. Now it is straightforward to show that G' possesses an optimal permutation M_2 in which vxy is the leftmost column. But this means we can construct a cost-three permutation for G by placing M_2 to the right of M_1 and removing uvw and vxy, a contradiction. □

Lemma. 7.10. *If an obstruction for* 3-GML *contains two faces that are adjacent at and only connected through a single vertex, then there is an obstruction for* 3-GML *with one less face and with a vertex whose attachment is two or three pendant paths.*

Proof. Let G denote an obstruction with faces F_1 and F_2 adjacent at and only connected through v. Assume F_1 is a triangle in which only v has degree three or more (Lemma 7.8). Let H denote the graph obtained from G by deleting $F_1\backslash\{v\}$ and identifying the degree-three vertex of (a disjoint copy of) $S(K_{1,3})$ with v. H has cost four (Lemma 5.16). Let G' denote an obstruction contained in H. Observe that, in G', the attachment at v contains more than one pendant path, else $G' <_m G$. Thus, due to Corollary 5.17, either $G' = H$ or $G' = H\backslash\{vx, xy\}$ where x and y are vertices on a pendant path incident on v and the lemma holds. □

Corollary 7.11. *If an obstruction for* 3-GML *contains two faces that are adjacent at and only connected through a single vertex v, then v has no attachment.*

We say that two disjoint faces are *separated* if the removal of some edge places the faces in different connected components.

Lemma 7.12. *If an obstruction for* 3-GML *contains a pair of separated faces, then the obstruction is one obtained from Lemma* 6.1.

[7] As justified by Lemma 5.13, we shall from now on represent a triangular face by a column with three 1s rather than three columns each with two 1s.

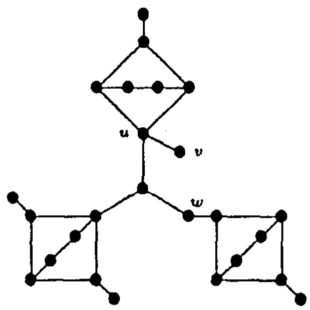


Fig. 5. A 4-GML obstruction.

Proof. Assume otherwise for obstruction G with separated faces F_1 and F_2. Let uv denote an edge of G whose removal places F_1 and F_2 in distinct connected components C_1 and C_2, respectively. Assume $u \in C_1$ and $v \in C_2$. C_1 must possess an optimal permutation M_1 in which every column to the right of the span for u has cost two, else $C_1 \backslash \{u\}$ contains two disjoint obstructions for 2-GML and the minimality of G ensures that it is obtained from Lemma 6.1. Similarly, C_2 must possess an optimal permutation M_2 in which every column to the left of the span for v has cost two. But now M_1, uv, M_2 is a cost-three permutation for G, a contradiction. □

7.4. Nonextendability of these results to four or more tracks

Unfortunately, the results of this section cannot be extended to values of $k > 3$. Consider, for example, the graph depicted in Fig. 5. We know from Lemma 6.1 that it is an obstruction for 4-GML.

Clearly, analogs of Lemmas 7.1 and 7.6 are ruled out by uv and w. Similarly, Lemma 6.2 quickly gives rise to obstructions for 4-GML that eliminate analogs for Lemmas 7.8 and 7.9. More complicated constructions [12] can be devised to rule out analogs for Lemmas 7.3, 7.4 and 7.12.

8. The complete three-track obstruction set

In this section, we shall complete the task of identifying all obstructions for 3-GML. Each is given a three-integer name, denoting its number of vertices, its number of interior faces and an index. For example, obstruction 8.2.3 is the third obstruction we list with eight vertices and two faces. For the reader's convenience, the entire set is displayed in an appendix to this paper.

8.1. Obstructions from previous constructions

Lemma 6.1 provides twenty obstructions: ten are trees (22.0.1–10); six have one face (18.1.1–6); four have separated faces (10.3.1 and 14.2.1–3).

Lemma 6.2 provides forty-three more obstructions (6.4.1, 8.3.1, 9.4.1–2, 11.2.1–2, 11.3.1, 12.3.1, 13.2.1, 13.3.1–6, 15.1.1–4, 15.2.1–7, 16.2.1, 16.2.5–6, 17.1.1–3, 17.2.1, 17.2.3–4, 18.1.7, 18.1.9–10, 19.1.1–3, 20.1.1 and 21.1.1).

Lemma 6.4 provides one additional obstruction (15.1.5).

Therefore, including the five nonouterplanar obstructions identified in Section 7, sixty-nine obstructions for 3-GML are known up to this point.

8.2. Conventions for describing new obstructions

We know from [5, 12] and Lemma 7.12 that no more tree or separated-face obstructions are possible. Moreover, those with vertex-adjacent faces can be obtained indirectly with Lemma 7.10. Thus we now consider only outerplane graphs with either a single face or with two or more edge-adjacent faces. Without loss of generality, we assume the outerplane embedding induces a left-to-right ordering of the faces, so that we can employ a simple (decimal) integer *pattern* to denote its face structure. In such a pattern, the number of digits equals the number of faces, and the value of each digit equals the number of vertices in the corresponding face. (As we shall see later, this easy scheme suffices, because we need only consider candidate obstructions in which no interior face has more than six vertices.)

If a face contains four or more vertices, then we assume each vertex of the face has degree at least three (Lemmas 5.5, 5.6 and 7.6). If a vertex has an attachment, then we assume this attachment is either a pendant edge or one, two or three pendant paths (Lemma 7.9). If the attachment consists of three pendant paths, then a minimality-preserving replacement is possible thanks to Lemmas 5.16 and 7.6. We term this a *type* 1 *replacement*. If the attachment is a pendant edge, then a minimality-preserving replacement is possible thanks to Lemma 5.6. We term this a *type* 2 *replacement*. Fig. 6 illustrates these two replacements, which we shall use to identify obstructions that might otherwise be missed due to the assumptions just stated.

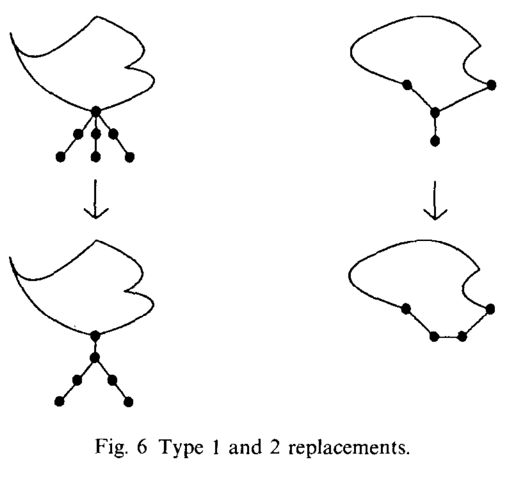


Fig. 6 Type 1 and 2 replacements.

We can thus use a succinct (character) *string* to denote a graph's attachment structure. We begin by visiting the vertices that lie on any internal face clockwise around the external face. If two or more (internal) faces are present, then we start with the vertex at the "top" of the edge shared by the leftmost two faces, otherwise we start at an arbitrary vertex. Letting v_i denote the ith vertex visited in this fashion, we represent the attachment at v_i with the ith character of the string. Such a character is either a 0 to denote that there is no attachment, the letter e to denote that it is a pendant edge, or an integer in the range [1, 3] to denote the number of pendant paths it contains.

New obstruction candidates are now uniquely (modulo rotations and reflections) describable in *pattern-string* form. For example, the graph denoted by 34-2e300 contains a triangle, edge adjacent to a square to its right. These two faces share the edge v_1v_4. The triangle's vertex set is $\{v_1, v_4, v_5\}$. The attachments at vertices v_1, v_2, and v_3 are, respectively, two pendant paths, a pendant edge and three pendant paths.

In describing permutations of graphs, we adopt the convention that u_i denotes the other vertex of an edge pendant at v_i. It is also helpful to use a shorthand for (complete and partial) permutations of more complicated attachments. See Fig. 7. For example, if three pendant paths are incident on v, then we use $A(v)$ in a permutation to indicate that the six edges of the attachment are to be placed in the order listed.

8.3. Obstructions with one face

Triangular face. If two vertices of the face have degree two, then it is straightforward to show that the graph can be obtained from Lemma 6.1. Otherwise, since the attachments at the vertices of the face are minors of $S(K_{1,3})$, the graph can be obtained from Lemma 6.2. Hereafter, we shall not consider any string that contains 333, 3321, 3312, 3213, 3123, 2133 or 1233, since the corresponding graph contains a minor whose pattern-string is 3-333 (known obstruction 21.1.1).

Square face. Pattern-string 4-2221 denotes new obstruction 18.1.8. Pattern-string 4-232e represents known obstruction 19.1.2. Any other graph with this pattern either contains one of these obstructions, or is a minor of a graph whose cost-three permutation resides in the following list.

4-3131 $A(v_1), C_1(v_2), v_1v_2, v_2v_3, v_1v_4, v_3v_4, C_2(v_4), A(v_3)$

4-323e $A(v_1), B_1(v_2), v_1v_2, v_1v_4, u_4v_4, v_3v_4, v_2v_3, B_2(v_2), A(v_3)$

4-3311 $A(v_1), C_1(v_4), v_1v_4, v_1v_2, v_3v_4, v_2v_3, C_2(v_3), A(v_2)$

Pentagonal face. Pattern-strings 5-11111 and 5-22e1e correspond to known obstructions 15.1.5 and 17.1.1, respectively. Any new obstruction with a pentagonal face contains at least one, and at most two pendant edges. If a string has a single e and a single 1, then the corresponding graph contains obstruction 18.1.8 (4-2221). Any

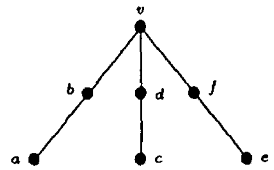
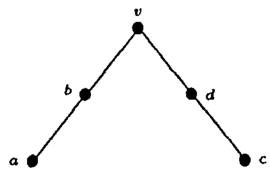
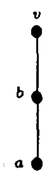
Attachment	Name	Notation		Permutation
	A	$A(v)$	v	0 1 * 1 1 0
			a	1 0 0 0 0 0
			b	1 1 0 0 0 0
			c	0 0 1 0 0 0
			d	0 0 1 1 0 0
			e	0 0 0 0 0 1
			f	0 0 0 0 1 1
	B	$B_1(v)$	v	0 1
			a	1 0
			b	1 1
		$B_2(v)$	v	1 0
			c	0 1
			d	1 1
	C	$C_1(v)$	v	0 1
			a	1 0
			b	1 1
		$C_2(v)$	v	1 0
			a	0 1
			b	1 1

Fig. 7. Shorthand used in permutations.

other candidate obstruction is a minor of a graph whose cost-three permutation resides in the following list.

5-3131e $A(v_1), C_1(v_2), v_1v_2, v_1v_5, u_5v_5, v_4v_5, v_3v_4, v_2v_3, C_2(v_4), A(v_3)$

5-3113e $A(v_1), C_1(v_2), v_1v_2, v_1v_5, u_5v_5, v_4v_5, v_3v_4, v_2v_3, C_2(v_3), A(v_4)$

5-1331e $A(v_2), C_1(v_1), v_2v_3, v_1v_2, v_1v_5, u_5v_5, v_4v_5, v_3v_4, C_2(v_4), A(v_3)$

Hereafter, no string with five or more entries from $\{1, 2, 3\}$ will be considered, because the corresponding graph contains known obstruction 15.1.5.

Hexagonal face. If three vertices of a graph with a hexagonal face have pendant edges incident on them, then the graph contains known obstruction 15.1.1 (6-1e1e1e). Thus we need only consider strings whose two e characters are in the third and sixth positions. The graph with pattern-string 6-22e11e contains known obstruction 17.1.1 (5-22e1e). All other possibilities are minors of a graph whose cost-three permutation resides in the following list.

6-31e31e $A(v_1), C_1(v_2), v_1v_2, v_2v_3, u_3v_3, v_1v_6, u_6v_6, v_5v_6, v_3v_4, v_4v_5, C_2(v_5), A(v_4)$

6-13e31e $A(v_2), C_1(v_1), v_1v_2, v_2v_3, u_3v_3, v_1v_6, u_6v_6, v_5v_6, v_3v_4, v_4v_5, C_2(v_5), A(v_4)$

Other faces. Any graph that contains a face with seven or more vertices, each with an attachment, must contain either known obstruction 15.1.1 (6-1e1e1e) or known obstruction 15.1.5 (5-11111). An obstruction whose face contains seven or more vertices must therefore have adjacent vertices of degree two on the face, in which case the obstruction can be obtained from a type 2 replacement and has already been considered.

Lemma 8.1. *There are exactly* 23 *obstructions for* 3-GML *that contain only one face.*

In summary, only one new one-faced obstruction exists, bringing the total number of known obstructions up to 70.

8.4. Obstructions with two faces

To identify obstructions with two vertex-adjacent faces, we apply the reverse of the replacement used in the proof of Lemma 7.10. Table 1 summarizes the two-faced obstructions thereby obtained. Other two-faced obstructions must contain edge-adjacent faces.

Two triangles. Pattern-string 33-0232 represents new obstruction 18.2.2, from which new obstruction 17.2.5 is obtained with a type 1 replacement. Pattern-string

Table 1
Two-faced obstructions from Lemma 7.10

Starting one-faced obstruction	Resultant two-faced obstruction(s)
17.1.1	15.2.4
17.1.2	15.2.5, 15.2.6
17.1.3	15.2.7
18.1.8	15.2.2, 16.2.4[a]
18.1.9	16.2.5
18.1.10	16.2.6
19.1.1	15.2.1
19.1.2	15.2.2, 17.2.3
19.1.3	15.2.3, 17.2.4
20.1.1	16.2.1
21.1.1	17.2.1

[a] New obstruction.

33-3230 denotes new obstruction 20.2.1, from which new obstructions 19.2.1 and 18.2.1 are obtained with type 1 replacements. The graph with pattern-string 33-2221 contains known obstruction 18.1.8 (4-2221). Pattern-string 33-2e22 represents new obstruction 17.2.6, from which new obstruction 17.2.7 is obtained with a type 2 replacement. All other possibilities are minors of a graph whose cost-three permutation resides in the following list.

33-0323	$A(v_2), B_1(v_3), v_1v_2v_3,\ v_1v_3v_4, B_2(v_3), A(v_4)$
33-1313	$A(v_2), C_1(v_1), v_1v_2v_3,\ v_1v_3v_4, C_2(v_3), A(v_4)$
33-3113	$A(v_1), C_1(v_2), v_1v_2v_3,\ v_1v_3v_4, C_2(v_3), A(v_4)$
33-3131	$A(v_1), C_1(v_2), v_1v_2v_3,\ v_1v_3v_4, C_2(v_4), A(v_3)$
33-3320	$A(v_2), B_1(v_3), v_1v_2v_3,\ v_1v_3v_4, B_2(v_3), A(v_1)$

Triangle and square. We assume the square is to the right of the triangle, so that both v_2 and v_3 must have attachments. If there is no e in the string, then there is at least one 0 in a position corresponding to a vertex of the triangle. Since known obstruction 13.2.1 has pattern-string 34-02200, we only consider graphs in which v_2 or v_3 has a pendant edge or a single pendant path as its attachment. A string with three 2s and a 1 corresponds to a graph that contains known obstruction 18.1.8 (4-2221). Pattern-string 34-21120 represents new obstruction 17.2.2. Pattern-string 34-2e102 denotes new obstruction 16.2.2, from which new obstruction 16.2.3 is obtained with a type 2 replacement. Pattern-string 34-111e denotes new obstruction 14.2.7, from which new obstruction 14.2.8 is obtained with a type 2 replacement. Graphs with pattern-strings 34-2e230 and 34-2e232 contain known obstruction 19.1.2 (4-232e). The graph with pattern-string 34-1e22e contains known obstruction 17.1.1 (5-22e1e). All other possibilities are minors of a graph whose cost-three permutation resides in the following list.

34-0e323	$A(v_3), B_1(v_4), v_3v_4, v_2v_3, u_2v_2, v_1v_2,\ v_1v_4v_5, B_2(v_4), A(v_5)$
34-01313	$A(v_3), C_1(v_2), v_2v_3, v_1v_2, v_3v_4, v_1v_4v_5, C_2(v_4), A(v_5)$
34-01331	$A(v_3), C_1(v_2), v_2v_3, v_1v_2, v_3v_4, v_1v_4v_5, C_2(v_5), A(v_4)$
34-03113	$A(v_2), C_1(v_3), v_2v_3, v_1v_2, v_3v_4, v_1v_4v_5, C_2(v_4), A(v_5)$
34-03131	$A(v_2), C_1(v_3), v_2v_3, v_1v_2, v_3v_4, v_1v_4v_5, C_2(v_5), A(v_4)$
34-1e133	$A(v_4), C_1(v_3), v_3v_4, v_2v_3, u_2v_2, v_1v_2,\ v_1v_4v_5, C_2(v_1), A(v_5)$
34-1e313	$A(v_3), C_1(v_4), v_3v_4, v_2v_3, u_2v_2, v_1v_2,\ v_1v_4v_5, C_2(v_1), A(v_5)$
34-11330	$A(v_3), C_1(v_2), v_3v_4, v_2v_3, v_1v_2, v_1v_4v_5, C_2(v_1), A(v_4)$
34-13130	$A(v_2), C_1(v_3), v_2v_3, v_3v_4, v_1v_2, v_1v_4v_5, C_2(v_1), A(v_4)$
34-3e131	$A(v_4), C_1(v_3), v_3v_4, v_2v_3, u_2v_2, v_1v_2,\ v_1v_4v_5, C_2(v_5), A(v_1)$
34-3e311	$A(v_3), C_1(v_4), v_3v_4, v_2v_3, u_2v_2, v_1v_2,\ v_1v_4v_5, C_2(v_5), A(v_1)$
34-3e320	$A(v_3), B_1(v_4), v_3v_4, v_2v_3, u_2v_2, v_1v_2,\ v_1v_4v_5, B_2(v_4), A(v_1)$

Two squares. Pattern-string 44-1e101e denotes new obstruction 14.2.4, from which new obstructions 14.2.5 and 14.2.6 are obtained with type 2 replacements. Graphs with pattern-strings 44-2e10e2 and 44-0e22e1 contain known obstruction 17.1.1 (5-22e1e). The graph with pattern-string 44-0e2320 contains known obstruction 19.1.2 (4-232e). If a string contains no e, then its first and fourth characters must both be 0 (Lemma 7.1 and avoidance of known obstruction 15.1.5 (5-11111)). Known obstruction 13.2.1 (34-02200) is a minor of any graph with pattern 44 in which both v_2 and v_3 (or both v_5 and v_6) have two or more pendant paths as attachments. All other possibilities are minors of a graph whose cost-three permutation resides in the following list.

44-0e1313 $A(v_4), C_1(v_3), v_3v_4, v_2v_3, u_2v_2, v_1v_2, v_1v_4, v_4v_5, v_1v_6, v_5v_6, C_2(v_5), A(v_6)$

44-0e1331 $A(v_4), C_1(v_3), v_3v_4, v_2v_3, u_2v_2, v_1v_2, v_1v_4, v_4v_5, v_1v_6, v_5v_6, C_2(v_6), A(v_5)$

44-0e3113 $A(v_3), C_1(v_4), v_3v_4, v_2v_3, u_2v_2, v_1v_2, v_1v_4, v_4v_5, v_1v_6, v_5v_6, C_2(v_5), A(v_6)$

44-0e3131 $A(v_3), C_1(v_4), v_3v_4, v_2v_3, u_2v_2, v_1v_2, v_1v_4, v_4v_5, v_1v_6, v_5v_6, C_2(v_6), A(v_5)$

44-0e323e $A(v_3), B_1(v_4), v_3v_4, v_2v_3, u_2v_2, v_1v_2, v_1v_4, v_1v_6, u_6v_6, v_5v_6, v_4v_5, B_2(v_4), A(v_5)$

44-0e331e $A(v_3), v_3v_4, v_2v_3, u_2v_2, v_1v_2, v_1v_4, v_1v_6, u_6v_6, v_5v_6, v_4v_5, C_2(v_5), A(v_4)$

44-031031 $A(v_2), C_1(v_3), v_2v_3, v_1v_2, v_3v_4, v_1v_4, v_1v_6, v_4v_5, v_5v_6, C_2(v_6), A(v_5)$

44-031013 $A(v_2), C_1(v_3), v_2v_3, v_1v_2, v_3v_4, v_1v_4, v_1v_6, v_4v_5, v_5v_6, C_2(v_5), A(v_6)$

44-1e31e3 $A(v_3), C_1(v_4), v_3v_4, v_2v_3, u_2v_2, v_1v_2, v_1v_4, v_4v_5, u_5v_5, v_5v_6, v_1v_6, C_2(v_1), A(v_6)$

44-3e13e1 $A(v_4), C_1(v_3), v_3v_4, v_2v_3, u_2v_2, v_1v_2, v_1v_4, v_4v_5, u_5v_5, v_5v_6, v_1v_6, C_2(v_6), A(v_1)$

44-3e31e1 $A(v_3), C_1(v_4), v_3v_4, v_2v_3, u_2v_2, v_1v_2, v_1v_4, v_4v_5, u_5v_5, v_5v_6, v_1v_6, C_2(v_6), A(v_1)$

Other patterns. The next result ensures that all two-faced obstructions with other patterns are already known (either by Lemma 6.2 or by type 2 replacements).

Lemma 8.2. *Obstruction* 11.2.1 *is the only two-faced outerplane obstruction for* 3-GML *with edge-adjacent faces in which one face has five or more vertices each with degree at least three.*

Proof. Assume otherwise for some obstruction G with faces F_1 and F_2, where $F_1 \cap F_2 = v_1v_m$, $m \geqslant 5$, and vertices $v_2, v_3, \ldots, v_{m-1}$ of F_2 each has an attachment. Since G does not by assumption contain obstruction 11.2.1 (35-01e100), the attachment at v_2 or v_4 must be a pendant edge, and the attachment at v_3 must be one or more pendant paths.

Suppose the attachment at v_2 is the pendant edge u_2v_2. Let $G' = G\backslash\{u_2v_2\}$. Thanks to Lemma 7.4, G' possesses a cost-three permutation M' in which the overlap of the face spans for F_1 and F_2 is column v_1v_m, the leftmost column of F_2. If the attachment at v_4 contains one or more pendant paths, then v_1v_2 must be the rightmost column in the span for v_1. But this means that a cost-three permutation for G can be constructed from M', a contradiction. Thus the attachment at v_4 is a pendant edge. It follows that F_2 must be a pentagon (else v_5 has an attachment with one or more pendant paths and G properly contains obstruction 11.2.1). Additionally, both v_1 and v_5 must have attachments, since otherwise M' can again be modified to produce a cost-three permutation for G. It is now clear that F_1 must be a triangle with vertex set $\{v_1, v_5, v_6\}$, and that v_6 must have degree two, else G properly contains obstruction 15.1.1 (6-1e1e1e). Also, the attachment at v_1 or v_5 must be a single pendant path, else G contains obstruction 17.1.1 (5-2e1e2). But this means that G is a minor of the graph with pattern-string 35-3e3e10, which has cost-three permutation $A(v_1)$, $C_1(v_5)$, $v_1v_5v_6$, v_1v_2, u_2v_2, v_4v_5, u_4v_4, v_3v_4, v_2v_3, $A(v_3)$, again a contradiction.

Suppose the attachment at v_2 is one or more pendant paths. The attachment at v_4 must be a pendant edge, from which it again follows that F_2 must be a pentagon, reducing this by symmetry to the previous case. □

Lemma 8.3. *There are* 39 *obstructions for* 3-GML *that have exactly two faces.*

In summary, sixteen new two-faced obstructions exist, bringing the total number of known obstructions up to 86.

8.5. Obstructions with three faces

To identify obstructions with three faces some of which are adjacent at and only connected through a single vertex, we again apply the reverse of the replacement used in the proof of Lemma 7.10. Table 2 summarizes the three-faced obstructions thereby obtained.

In any additional three-faced obstruction, each face must be edge adjacent to at least one other. Furthermore, the three faces cannot be mutually edge adjacent, else the graph contains K_4.

Lemma 8.4. *No outerplane obstruction for* 3-GML *contains faces,* F , F_2, *and* F_3, *where* $F_1 \cap F_3 = \emptyset$, *such that both* F_1 *and* F_3 *are edge adjacent to* F_2.

Proof. Assume otherwise for some obstruction G with faces F_1, F_2, and F_3 for which $F_1 \cap F_2 = v_1v_r$ and $F_2 \cap F_3 = v_iv_j$, where $1 < i < j < r$.

Suppose F_2 is a square with vertex set $\{v_1, v_2, v_{r-1}, v_r\}$. Let $G' = G \backslash \{v_1v_r\}$, and let F'_2 denote the (enlarged) face that results from the removal of v_1v_r from F_2. G' possesses a cost-three permutation M' in which the overlap of the spans for F'_2 and F_3 is column v_2v_{r-1}, the leftmost column of F_3. If both v_1 and v_r have attachments, then their spans must include the leftmost column of F'_2, and v_1v_2 can be placed to the immediate left of the span for F'_2 to obtain a cost-three permutation for G, a contradiction. Thus v_1 or v_r (and analogously v_2 or v_{r-1}) has no attachment. If neither v_2 nor v_r has an attachment, then v_1v_2 can be moved to the immediate left of v_2v_{r-1} and

Table 2
Three-faced obstructions from Lemma 7.10

Starting two-faced obstruction	Resultant three-faced obstruction(s)
13.2.1	11.3.1
15.2.2	13.3.5
15.2.3	13.3.6
15.2.4	13.3.2
15.2.5	13.3.3
15.2.6	13.3.3
15.2.7	13.3.4
16.2.1	12.3.1
16.2.2	14.3.3[a], 14.3.5[a]
16.2.3	14.3.4[a], 14.3.6[a]
16.2.4	13.3.1, 14.3.2[a]
16.2.5	12.3.1
16.2.6	12.3.1
17.2.1	13.3.1
17.2.2	15.3.1[a]
17.2.3	13.3.1, 13.3.5
17.2.4	13.3.1, 13.3.6
17.2.5	15.2.1
17.2.6	15.2.2, 15.3.3[a]
17.2.7	15.2.3, 15.3.4[a]
18.2.1	15.2.1
18.2.2	14.3.1[a], 16.2.1
19.2.1	15.3.2[a], 16.2.1
20.2.1	16.3.1[a], 17.2.1

[a] New obstruction.

$v_{r-1}v_r$ can be moved to the immediate left of v_1v_2, making it easy to construct a cost-three permutation for G, a contradiction. Thus v_2 or v_r (and analogously v_1 or v_{r-1}) has an attachment. So, without loss of generality, assume both v_1 and v_2 have attachments. Let G'' denote the graph obtained from G by contracting edge $v_{r-1}v_r$ to v_r, and let F'' denote the triangle with vertex set $\{v_1, v_2, v_r\}$. G'' possesses a cost-three permutation M'' in which v_1v_2 lies between v_1v_r, the rightmost column of F_1, and v_2v_r, the leftmost column of F_3. M'' can now be modified by adding row v_{r-1}, replacing v_2v_r by $v_{r-1}v_r$ and v_2v_{r-1}, and, in every column to the right of v_2v_{r-1}, interchanging the contents of rows v_r and v_{r-1}, thereby producing a cost-three permutation for G, a contradiction.

F_2 must therefore have five or more vertices. Without loss of generality, assume v_2 does not lie on F_3 and has degree three or more. The attachment at v_2 must be the pendant edge u_2v_2 and v_3 must lie on F_3, else G contains obstruction 8.3.1 (343-010000). But now it is a simple matter to modify a cost-three permutation for $G\backslash\{u_2v_2\}$ to obtain a cost-three permutation for G, again a contradiction. $\square$

Three triangles. Since known obstruction 13.2.1 has pattern-string 34-02200, and since removal of v_1v_4 (or v_2v_4) leaves an edge-adjacent triangle and square, we do not consider any graph in which both v_2 and v_3 (or both v_1 and v_5) have two or more pendant paths as attachments. Any graph with attachments at all five vertices contains new obstruction 13.3.7 denoted by pattern-string 333-11e1e, from which new obstructions 13.3.8 and 13.3.9 are obtained with type 2 replacements. Pattern-string 333-21e20 denotes new obstruction 16.3.2, from which new obstruction 16.3.3 is obtained with a type 2 replacement. Pattern-string 333-22030 denotes new obstruction 19.3.1, from which new obstruction 18.3.1 is obtained with a type 1 replacement. Graphs with pattern-strings 333-00232 and 333-02032 contain known obstruction 18.2.2 (33-0232). The graph with pattern-string 333-12021 contains known obstruction 17.2.2 (34-21120). All other possibilities are minors of a graph whose cost-three permutation resides in the following list.

333-00323	$A(v_3), B_1(v_4), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, B_2(v_4), A(v_5)$
333-01313	$A(v_3), C_1(v_2), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, C_2(v_4), A(v_5)$
333-01331	$A(v_3), C_1(v_2), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, C_2(v_5), A(v_4)$
333-03023	$A(v_2), B_1(v_4), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, B_2(v_4), A(v_5)$
333-03113	$A(v_2), C_1(v_3), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, C_2(v_4), A(v_5)$
333-03131	$A(v_2), C_1(v_3), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, C_2(v_5), A(v_4)$
333-11033	$A(v_4), C_1(v_2), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, C_2(v_1), A(v_5)$
333-11303	$A(v_3), C_1(v_2), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, C_2(v_1), A(v_5)$
333-13013	$A(v_2), C_1(v_4), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, C_2(v_1), A(v_5)$
333-13103	$A(v_2), C_1(v_3), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, C_2(v_1), A(v_5)$
333-31031	$A(v_4), C_1(v_2), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, C_2(v_5), A(v_1)$
333-33011	$A(v_2), C_1(v_4), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, C_2(v_5), A(v_1)$
333-33020	$A(v_2), B_1(v_4), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, B_2(v_4), A(v_1)$
333-33101	$A(v_2), C_1(v_3), v_2v_3v_4, v_1v_2v_4, v_1v_4v_5, C_2(v_5), A(v_1)$

Other patterns. The next result ensures that all three-faced obstructions with other patterns are already known (either by Lemma 6.2 or by type 2 replacements).

Lemma 8.5. *Obstruction* 8.3.1 *is the only three-faced outerplane obstruction for* 3-GML *in which each face is edge adjacent to at least one other and one face has four or more vertices each with degree at least three.*

Proof. Assume otherwise for some obstruction G with faces F_1, F_2, and F_3 such that both F_1 and F_3 are edge adjacent to F_2. Thanks to lemma 8.4, we may assume $F_1 \cap F_2 = v_1v_r$, and $F_2 \cap F_3 = v_iv_r$.

Suppose F_2 is not a triangle. To avoid obstruction 8.3.1 (343-010000), F_2 must be a square with vertex set $\{v_1, v_2, v_3, v_r\}$, and v_2 must be adjacent to pendant vertex u_2. Let $G' = G\backslash\{u_2v_2\}$. G' possesses a cost-three permutation M' in which v_1v_2 and v_2v_3 lie between v_1v_r, the rightmost column of F_1, and v_3v_r, the leftmost column of F_3. It is straightforward to verify that v_1v_2 contains the rightmost 1 in row v_1, that v_2v_3 is to the immediate right of v_1v_2, and that u_2v_2 can be inserted in M' to produce a cost-three permutation for G, a contradiction.

Thus F_2 must be a triangle. Without loss of generality, assume F_3 has at least four vertices each with degree at least three. If v_2 has an attachment, then to avoid obstruction 11.2.1 (35-01e100) it follows that F_3 must be a square with vertex set $\{v_2, v_3, v_4, v_5\}$, the attachment at v_4 is the pendant edge u_4v_4, and the attachment at v_3 contains at least one pendant path. Let $G'' = G\backslash\{u_4v_4\}$ and let M'' denote a cost-three permutation for G'' in which the span for F_3 is to the right of column $v_1v_2v_5$. Since v_4v_5 must be the rightmost column in the span for v_5, it is straightforward to construct a cost-three permutation for G, a contradiction. Thus v_2 can have no attachment. Let G''' denote the graph obtained from G by contracting edge v_2v_3 to v_2, and let F_3''' denote the (shrunken) face that results from this contraction in F_3. Using a cost-three permutation for G''' in which the span for F_3''' is to the right of column $v_1v_2v_r$, it is again straightforward to construct a cost-three permutation for G, a contradiction. □

Lemma 8.6. *There are* 29 *obstructions for* 3-GML *that have exactly three faces.*

In summary, eighteen new three-faced obstructions exist, bringing the total number of known obstructions up to 104.

8.6. Obstructions with four faces

To identify obstructions with four faces some of which are adjacent at and only connected through a single vertex, we again apply the reverse of the replacement used in the proof of Lemma 7.10. Table 3 summarizes the four-faced obstructions thereby obtained.

In any additional four-faced obstruction, each face must be edge adjacent to at least one other. One face cannot be edge adjacent to the other three, else the graph contains known obstruction 6.4.1. Furthermore, to avoid K_4, at least two faces must be edge adjacent to exactly one other face. Our next result ensures that all four-faced obstructions are already known.

A *chain* in an outerplane graph is a sequence of faces $F_1, F_2, \ldots, F_h$ such that F_i and F_j intersect at a single edge if $|i-j| = 1$, and are either disjoint or intersect at a single vertex otherwise. The length of a chain is the number of faces it contains. Fig. 8 illustrates four different four-faced chains.

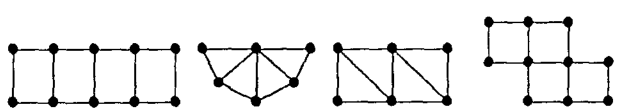


Fig. 8. Sample four-faced chains.

Table 3
Four-faced obstructions from Lemma 7.10

Starting three-faced obstruction	Resultant four-faced obstruction(s)
11.3.1	9.4.2
13.3.1	9.4.1
13.3.5	9.4.1
13.3.6	9.4.1
14.3.1	12.3.1
14.3.2	9.4.1
14.3.3	12.3.1
14.3.4	12.3.1
14.3.5	12.3.1
14.3.6	12.3.1
15.3.1	12.4.1[a]
15.3.2	12.3.1
15.3.3	12.4.1[a], 13.3.1
15.3.4	12.4.1[a], 13.3.1
16.3.1	12.4.1[a], 13.3.1
16.3.2	14.4.1[a], 14.4.3[a]
16.3.3	14.4.2[a], 14.4.4[a]
18.3.1	15.3.2
19.3.1	15.4.1[a], 16.3.1

[a] New obstruction.

Lemma 8.7. *No obstruction for* 3-GML *contains a chain whose length exceeds three.*

Proof. Assume otherwise for some obstruction G with chain $F_1, F_2, \ldots, F_h$ where $h \geqslant 4$ and $F_i \cap F_{i+1} = v_i w_i$ for $1 \leqslant i < h$.

Thanks to Lemma 8.4, we assume without loss of generality that $w_1 = w_2$. To avoid obstruction 8.3.1, G must contain either $v_1 v_2$ or a degree-three vertex x adjacent to v_1, v_2 and pendant vertex y.

Let $G' = G \backslash \{v_2 w_2\}$, and let F'_2 denote the (enlarged) face that results from the removal of $v_2 w_2$ from F_2. G' possesses a cost-three permutation in which the overlap of the spans for F_1 and F'_2 is $v_1 w_1$, the leftmost column of F'_2. Since any attachment at v_1 must lie to the left of the span for F_1, since the span for F_4 must be to the right of $v_1 w_1$, and since outerplanarity ensures $v_1 \notin F_4$, column $v_1 v_2$ (or column $v_1 x$) must contain the rightmost 1 in row v_1. Thus, with no increase in cost, column $v_1 v_2$ (or the set of columns a $v_1 x, xy, xv_2$) may be moved to the immediate right of $v_1 w_1$, from which it is straightforward to construct a cost-three permutation for G, a contradiction. □

Lemma 8.8. *There are nine obstructions for* 3-GML *that have exactly four faces.*

In summary, six new four-faced obstructions exist, bringing the total number of known obstructions up to 110. We shall now show that there are no more.

Table 4
A review of the 3-GML obstruction set

Number of faces	Number of obstructions
none	10
one	23
two	39
three	29
four	9
five or more	0

8.7. Obstructions with five or more faces

Lemma 8.9. *No obstruction for* 3-GML *contains five or more faces.*

Proof. The reverse of the replacement used in the proof of Lemma 7.10 generates only known obstructions 9.4.1 and 12.4.1. Thus there can be no obstruction with five or more faces some of which are adjacent at and only connected through a single vertex. Thanks to Lemmas 7.8 and 8.7, no obstruction can contain either separated faces or a chain whose length exceeds three. □

9. Main result

All elements of the 3-GML obstruction set are now known. The structure of this set is reviewed in Table 4.

Theorem 9.1. *There are exactly* 110 *obstructions for* 3-GML, *namely, those identified in preceding results and depicted in the appendix.*

10. Conclusions

Gate matrix layout is a well-known but notoriously difficult problem. Each of its fixed-parameter variants, however, possesses a finite-basis characterization that provides a polynomial-time recognition algorithm. In this paper, we have isolated the basis for parameter value three. In order to accomplish this, we have also derived a number of more general results to bound and identify basis elements for any parameter value.

We conjecture that the trees are the largest elements in each basis. A proof of this, if it is indeed true, would be particularly interesting, because it would automatically mean that every basis is computable. (Exhaustive computation could, at least in principle, be applied until the trees were reached, after which it would be pointless to look further.)

Lemma 6.1 makes it easy to see that basis size grows monotonically. This and the fact that the basis for parameter value four contains at least 122 million elements [12] suggest that no other bases for this problem are likely to be isolated in the foreseeable future.

References

[1] J.A. Bondy and U.S.R. Murty, Graph Theory with Applications (North-Holland, New York, 1976).

[2] R.L. Bryant, M.R. Fellows, N.G. Kinnersley and M.A. Langston, On finding obstruction sets and polynomial-time algorithms for gate matrix layout, in: Proceedings of the 25th Allerton Conference on Communication, Control, and Computing (1987) 397-398.

[3] D.J. Brown, M.R. Fellows and M.A. Langston, Polynomial-time self-reducibility: theoretical motivations and practical results, Internat. J. Comput. Math. 31 (1989) 1-9.

[4] N. Deo, M.S. Krishnamoorthy and M.A. Langston, Exact and approximate solutions for the gate matrix layout problem, IEEE Trans. Computer-Aided Design 6 (1987) 79-84.

[5] J.A. Ellis, I.H. Sudborough and J.S. Turner, The vertex separation and search number of a graph, Inform. and Comput., to appear.

[6] M.R. Fellows and M.A. Langston, Nonconstructive advances in polynomial-time complexity, Inform. Process. Lett. 26 (1987) 157-162.

[7] M.R. Fellows and M.A. Langston, Nonconstructive tools for providing polynomial-time decidability, J. ACM 35 (1988) 727-739.

[8] M.R. Fellows and M.A. Langston, On well-partial-order theory and its application to combinatorial problems of VLSI design, SIAM J. Discrete Math. 5 (1992) 117-126.

[9] M.R. Fellows and M.A. Langston, Fast search algorithms for layout permutation problems, Internat. J. Comput. Aided VLSI Design 3 (1991) 325-342.

[10] M.R. Fellows and M.A. Langston, On search, decision and the efficiency of polynomial-time algorithms, in: Proceedings of the 21st ACM Symposium on the Theory of Computing (1989) 501-512.

[11] F. Harary, Graph Theory (Addison-Wesley, Reading, MA, 1969).

[12] N.G. Kinnersley, Obstruction set isolation for layout permutation problems, Ph.D. Thesis, Washington State University (1989).

[13] C. Kuratowski, Sur le probleme des curbes gauches en topologie, Fund. Math. 15 (1930) 271-283.

[14] N. Robertson and P.D. Seymour, Graph minors I. Excluding a forest, J. Combin. Theory Ser. B 35 (1983) 39-61.

[15] N. Robertson and P.D. Seymour, Graph minors V. Excluding a planar graph, J. Combin. Theory Ser. B 41 (1986) 92-114.

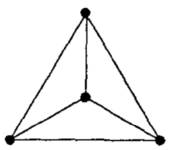
[16] N. Robertson and P.D. Seymour, Graph minors XIII. The disjoint paths problems, to appear.

[17] N. Robertson and P.D. Seymour, Graph minors XVI. Wagner's conjecture, to appear.

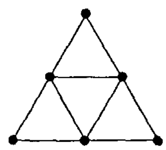
[18] P.D. Seymour, private communication.

Appendix

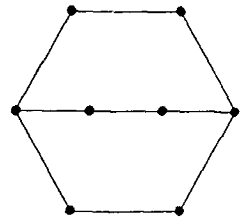
The 3-GML Obstruction Set



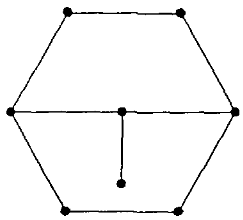
4.3.1



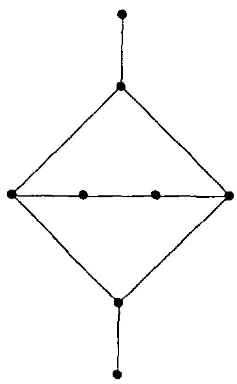
6.4.1



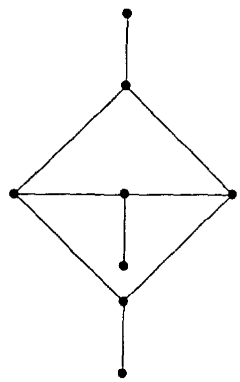
8.2.1



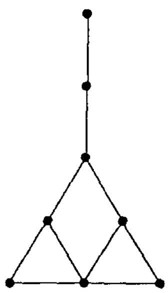
8.2.2



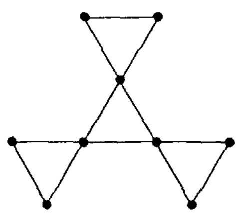
8.2.3



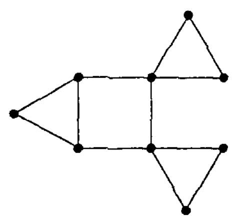
8.2.4



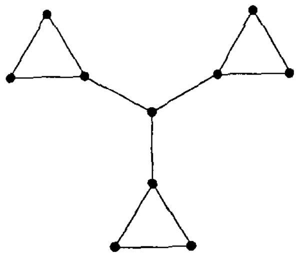
8.3.1



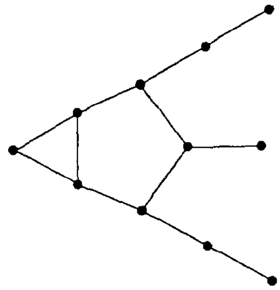
9.4.1



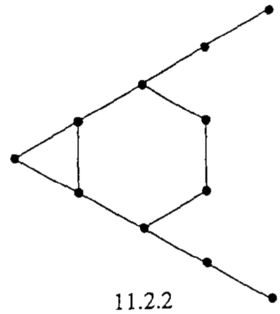
9.4.2



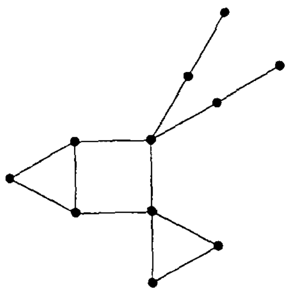
10.3.1



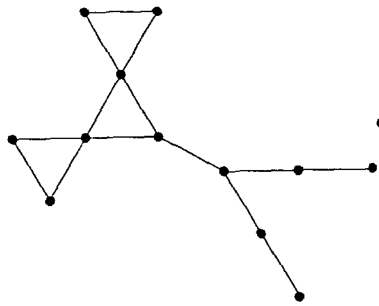
11.2.1



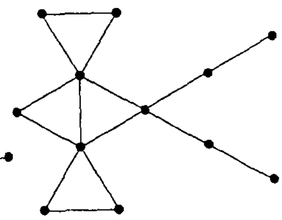
11.2.2



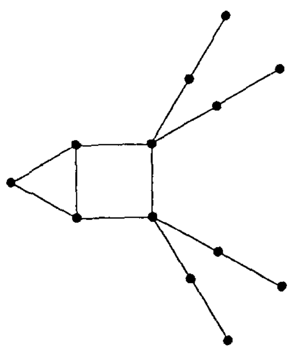
11.3.1



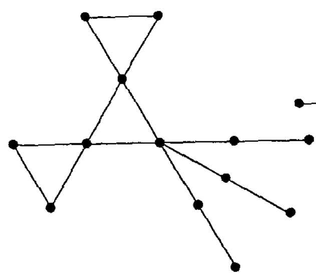
12.3.1



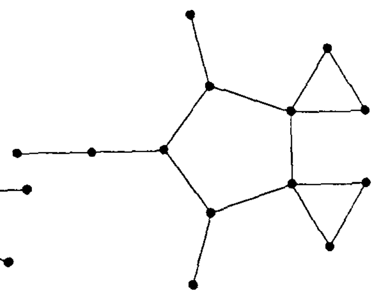
12.4.1



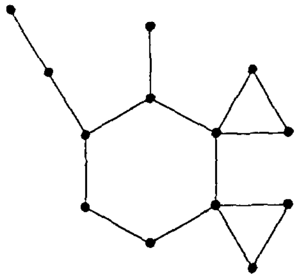
13.2.1



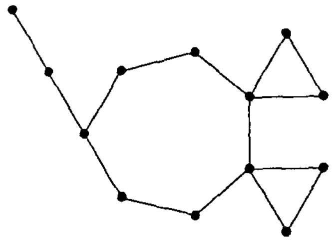
13.3.1



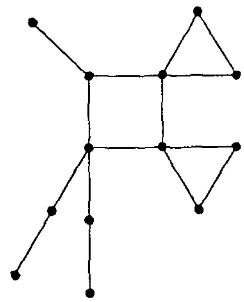
13.3.2



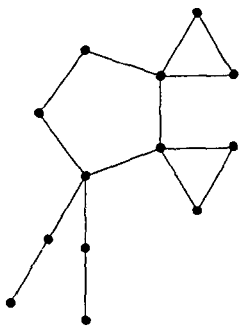
13.3.3



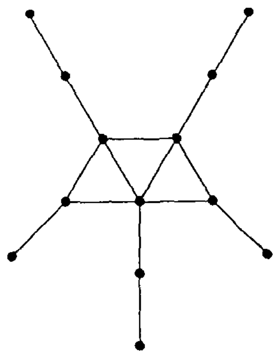
13.3.4



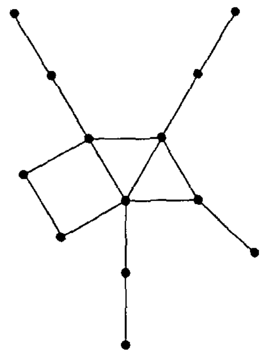
13.3.5



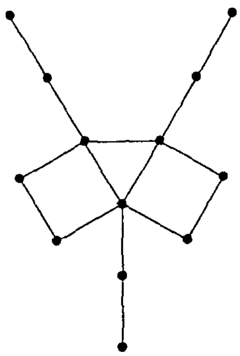
13.3.6



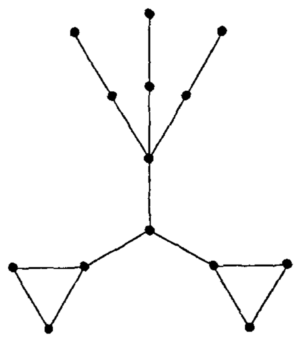
13.3.7



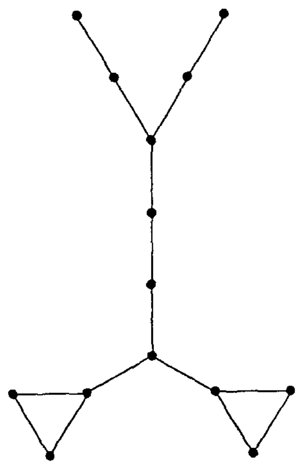
13.3.8



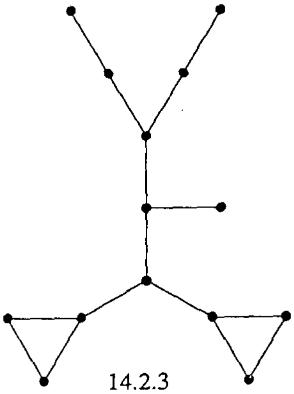
13.3.9



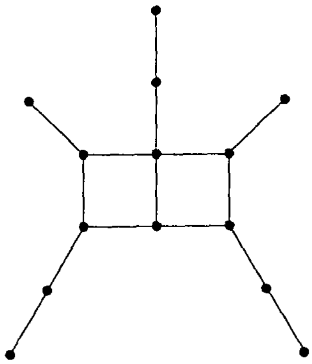
14.2.1



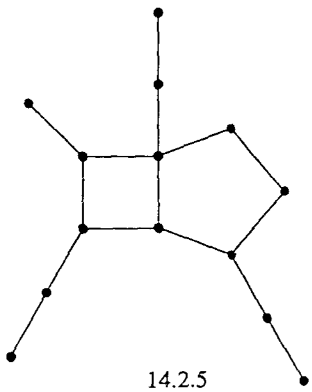
14.2.2



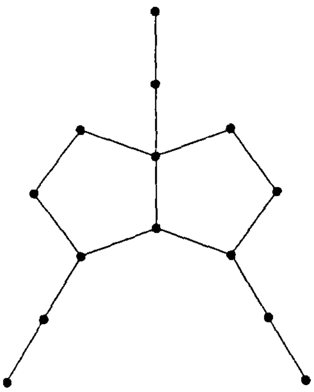
14.2.3



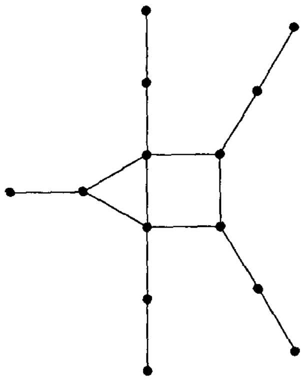
14.2.4



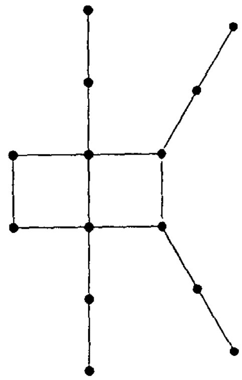
14.2.5



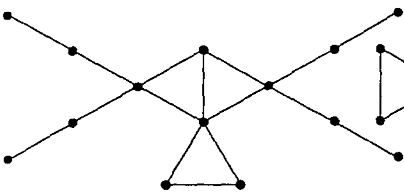
14.2.6



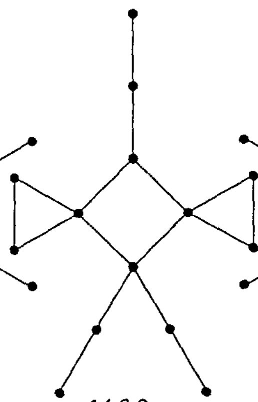
14.2.7



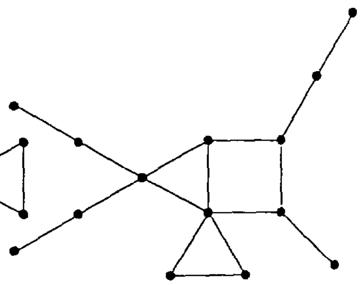
14.2.8



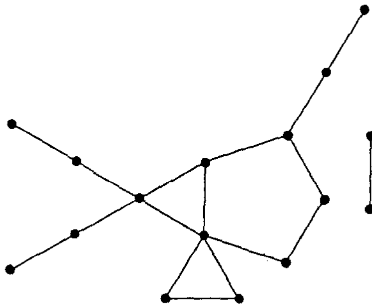
14.3.1



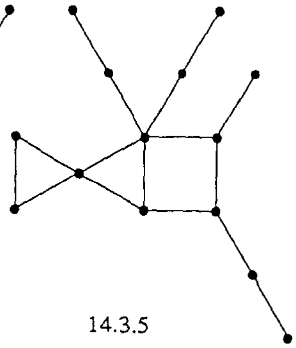
14.3.2



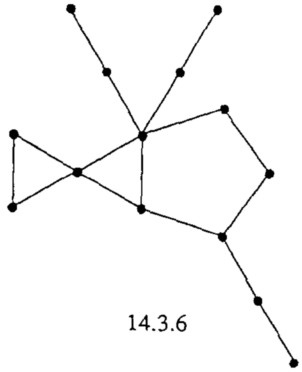
14.3.3



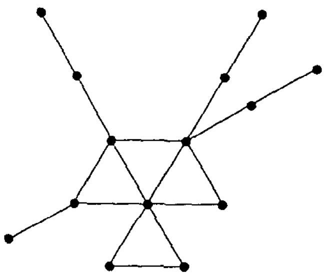
14.3.4



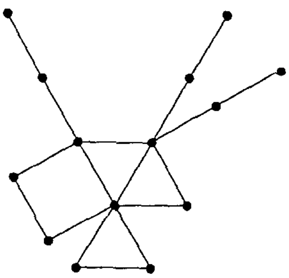
14.3.5



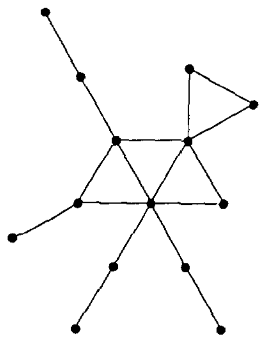
14.3.6



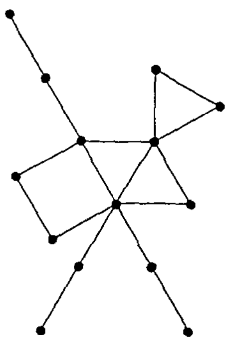
14.4.1



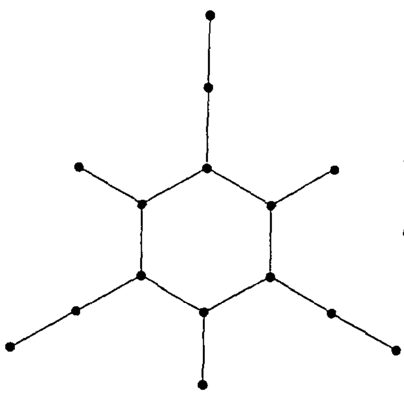
14.4.2



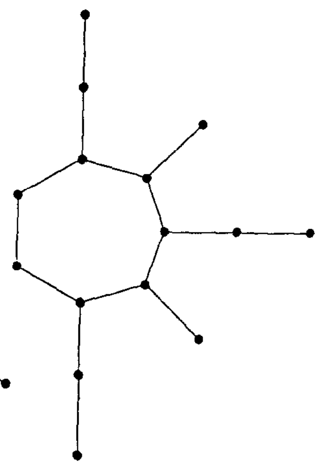
14.4.3



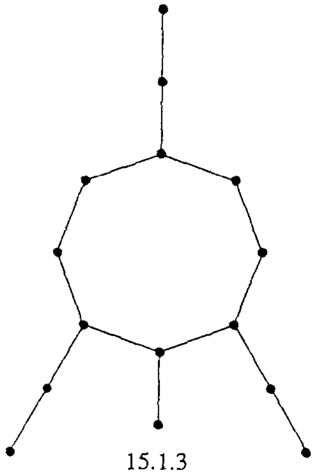
14.4.4

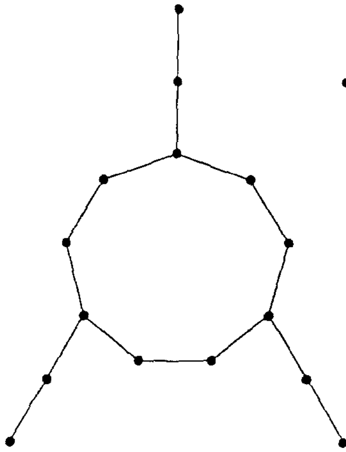


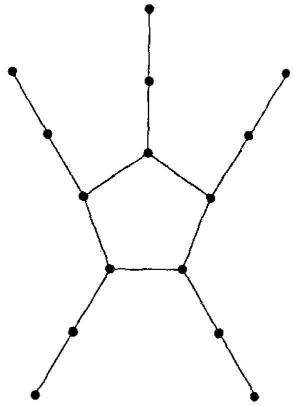
15.1.1

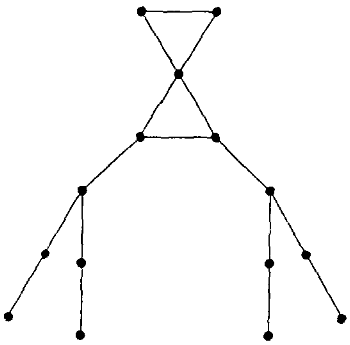


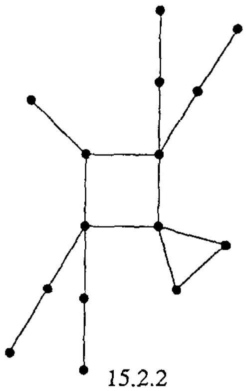
15.1.2

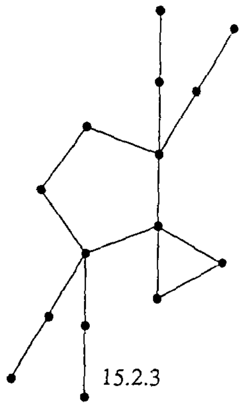

15.1.3

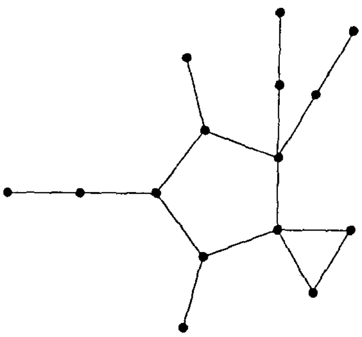

15.1.4

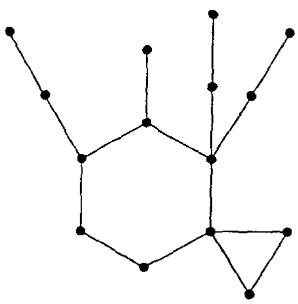

15.1.5

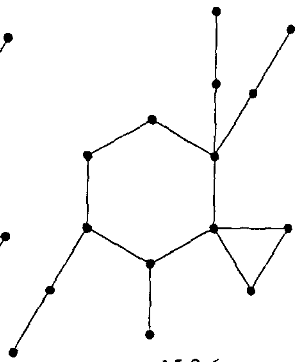

15.2.1

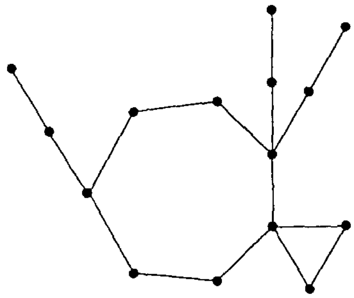

15.2.2


15.2.3

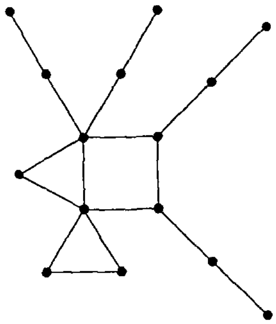

15.2.4


15.2.5

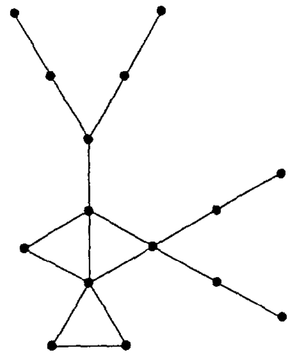

15.2.6



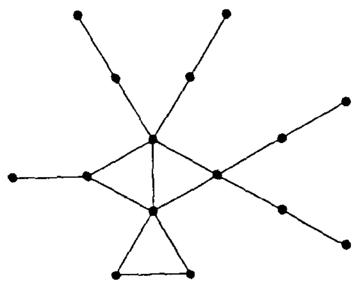
15.2.7



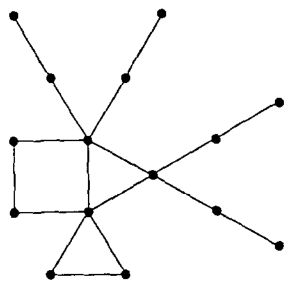
15.3.1



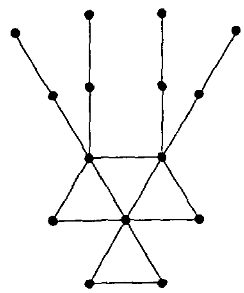
15.3.2



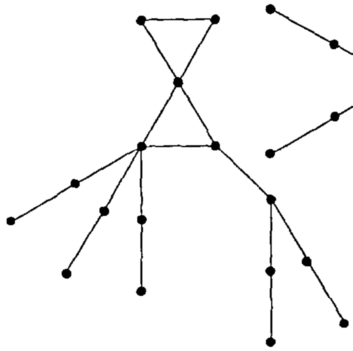
15.3.3



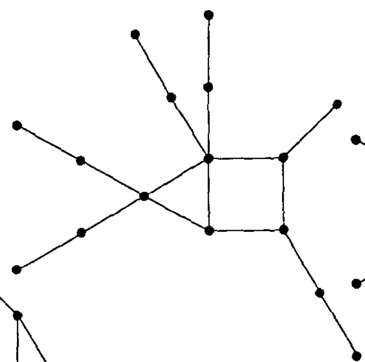
15.3.4



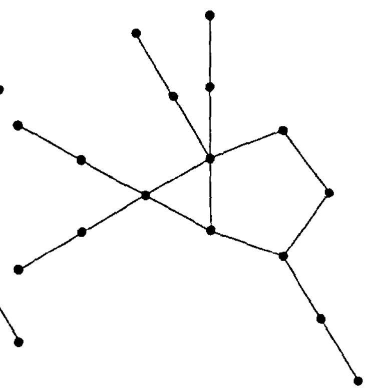
15.4.1



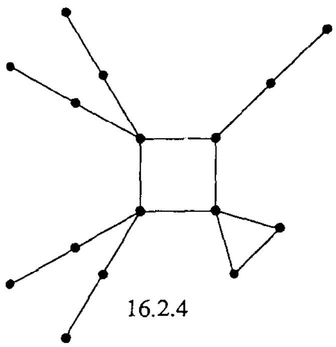
16.2.1



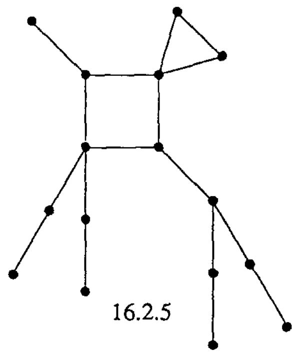
16.2.2



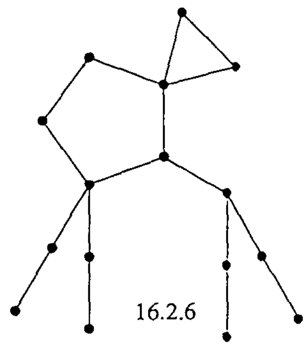
16.2.3



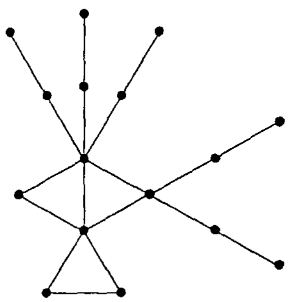
16.2.4



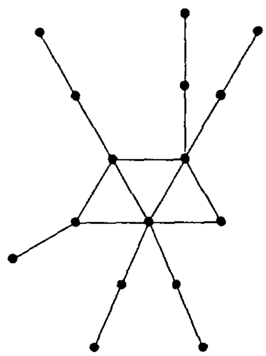
16.2.5



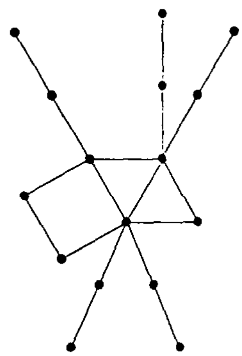
16.2.6



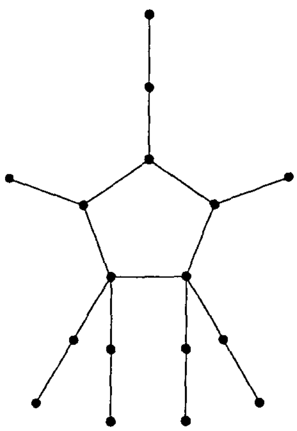
16.3.1



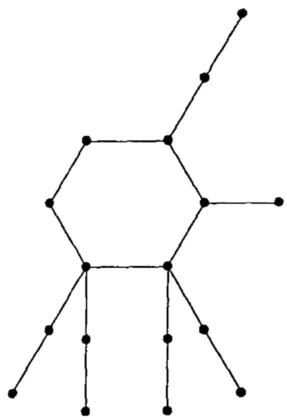
16.3.2



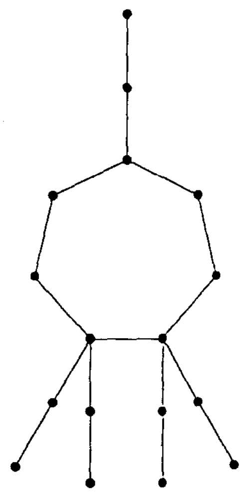
16.3.3



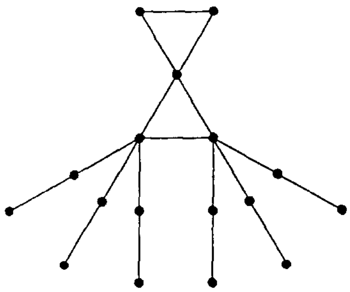
17.1.1



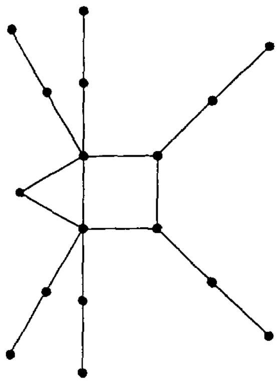
17.1.2



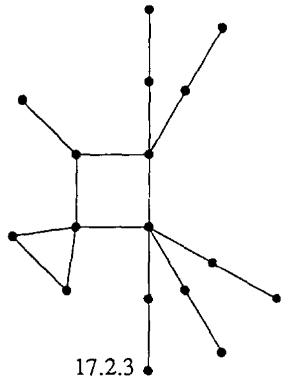
17.1.3



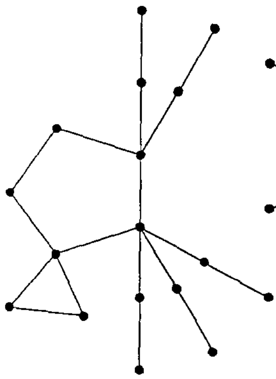
17.2.1



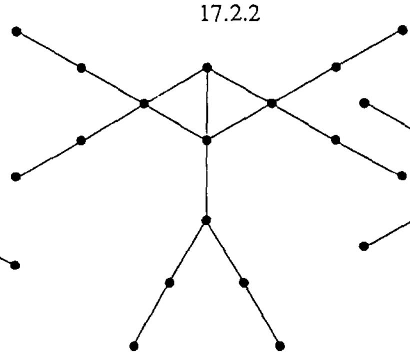
17.2.2



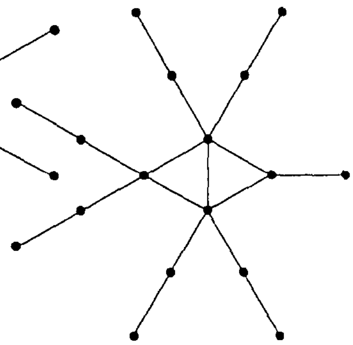
17.2.3



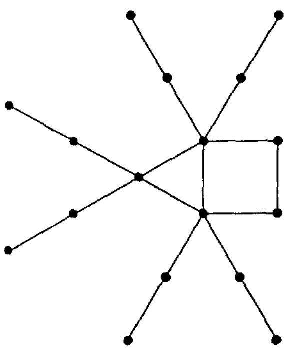
17.2.4



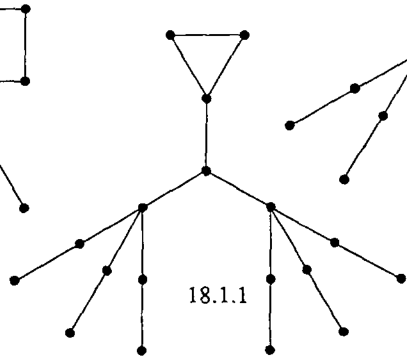
17.2.5



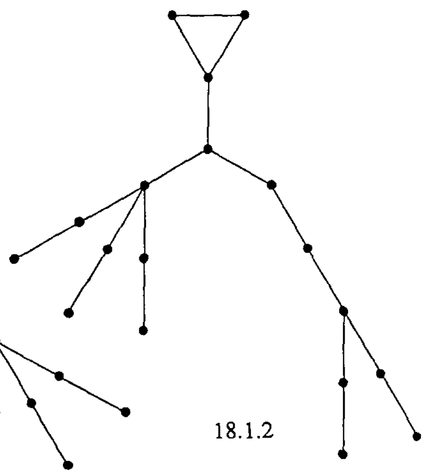
17.2.6



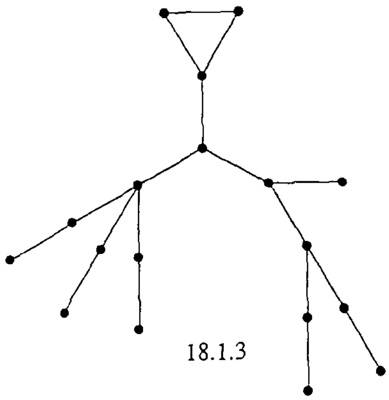
17.2.7



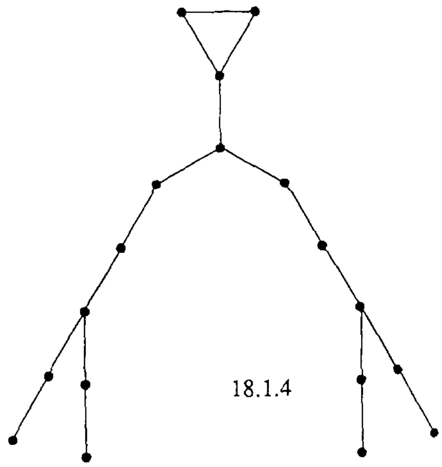
18.1.1



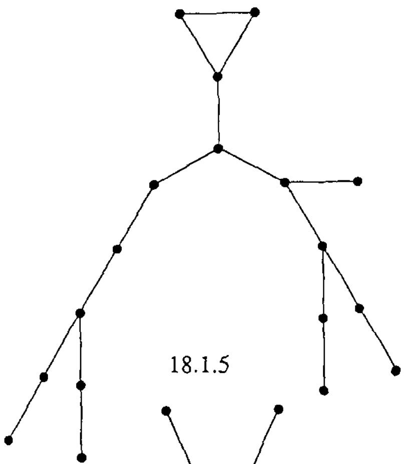
18.1.2



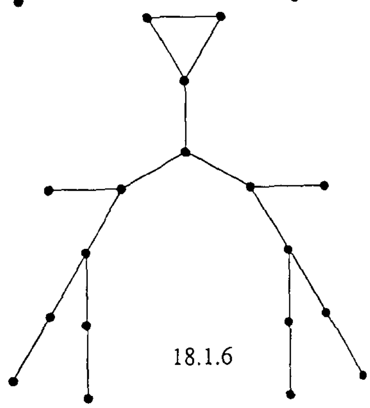
18.1.3



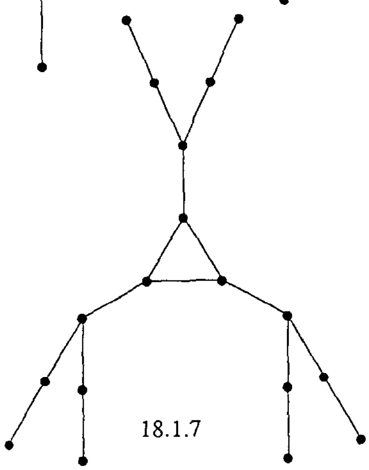
18.1.4



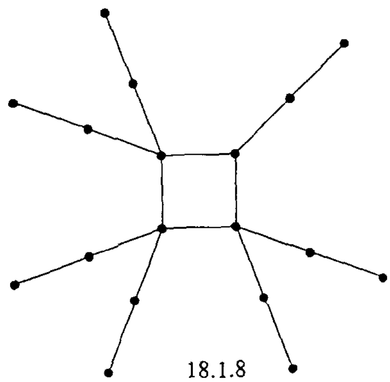
18.1.5



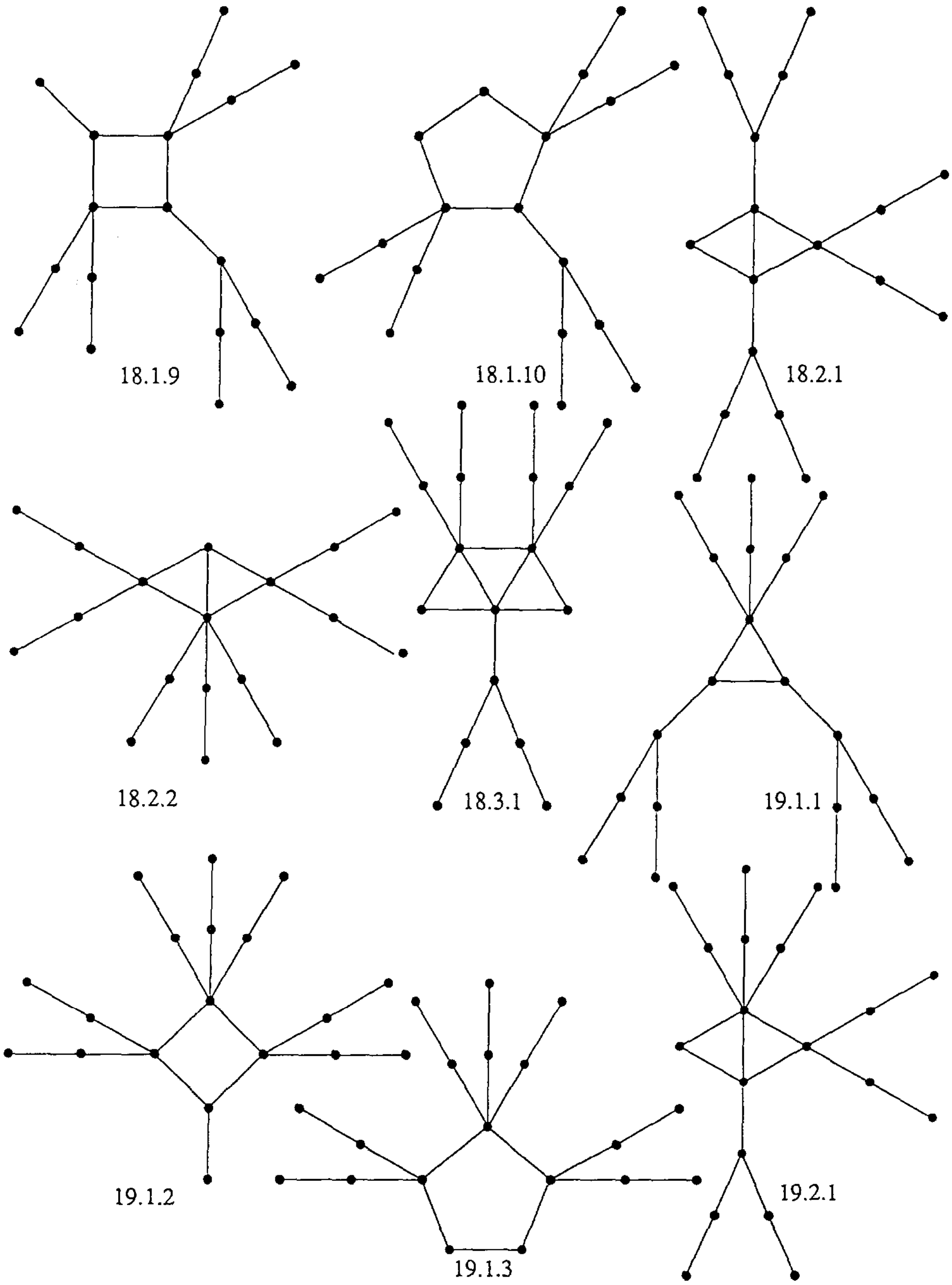
18.1.6

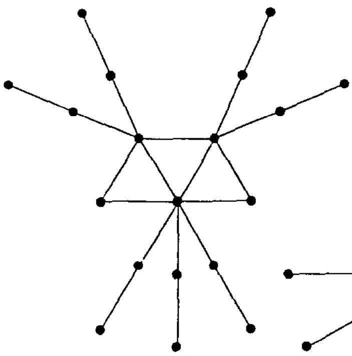


18.1.7

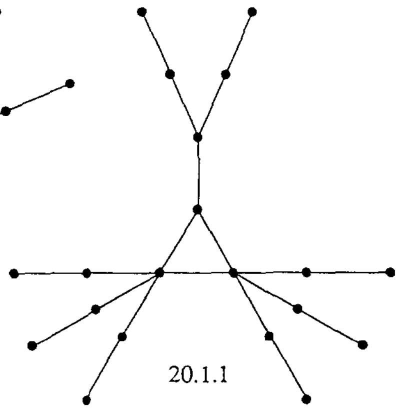


18.1.8

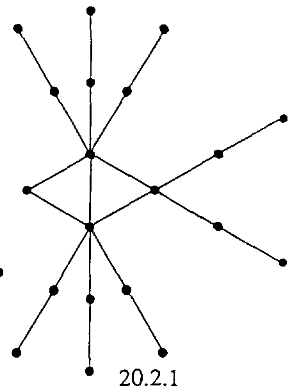

18.1.9
18.1.10
18.2.1
18.2.2
18.3.1
19.1.1
19.1.2
19.1.3
19.2.1



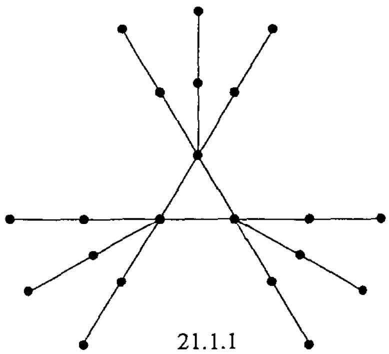
19.3.1



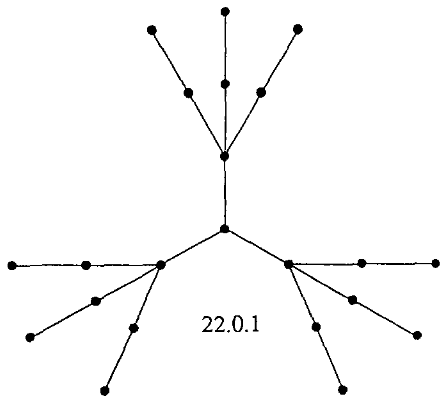
20.1.1



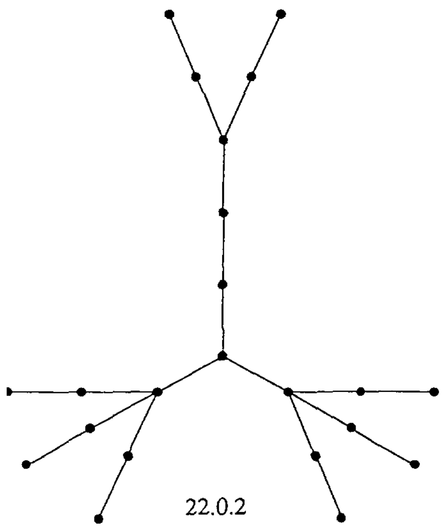
20.2.1



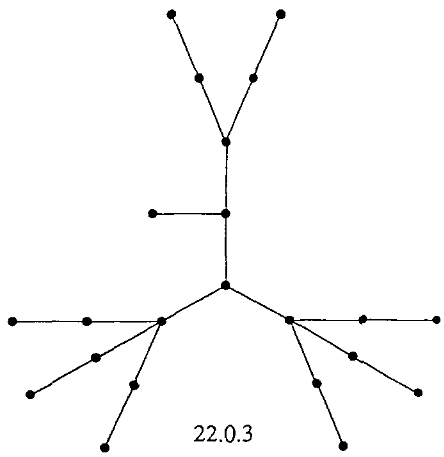
21.1.1



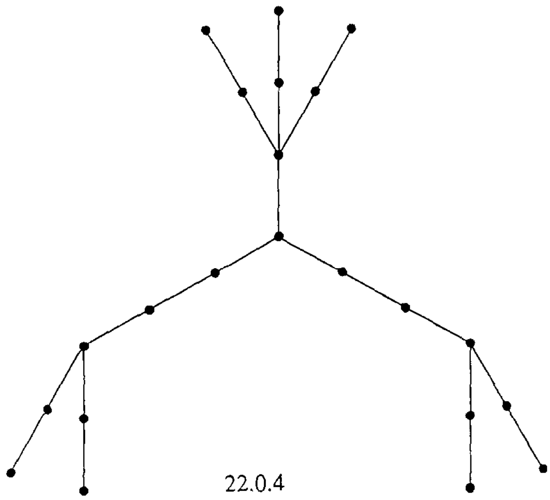
22.0.1



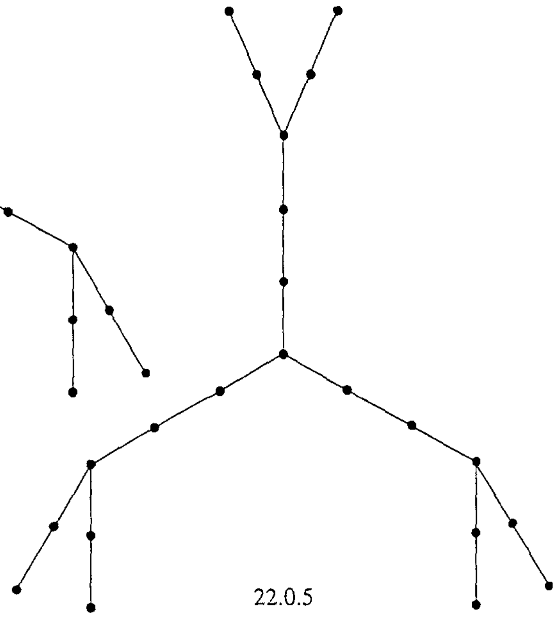
22.0.2



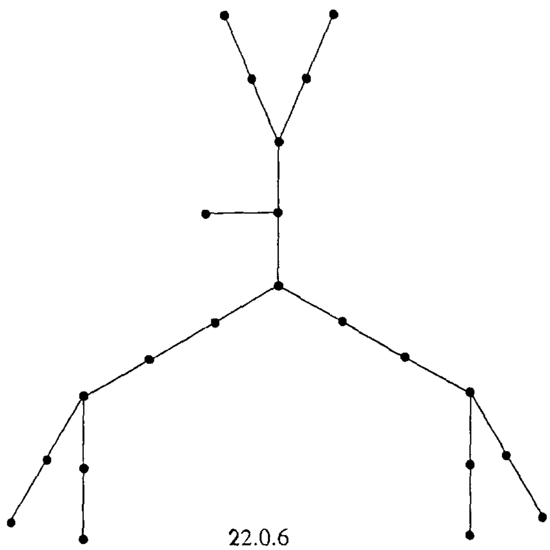
22.0.3



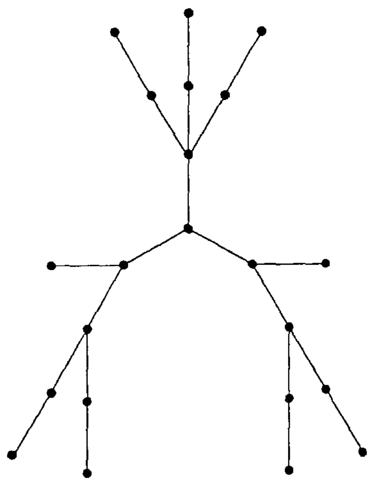
22.0.4



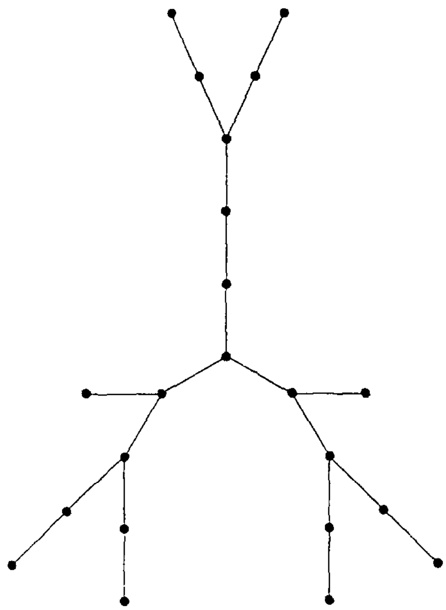
22.0.5



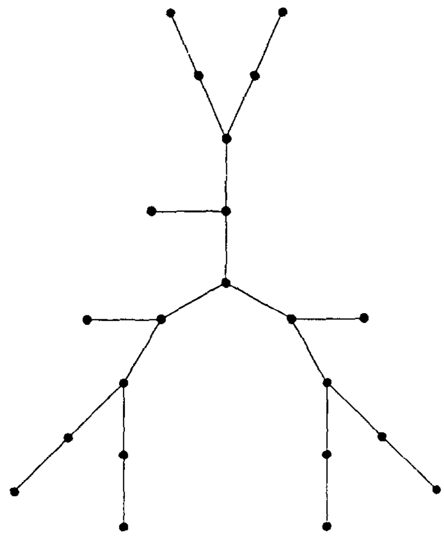
22.0.6



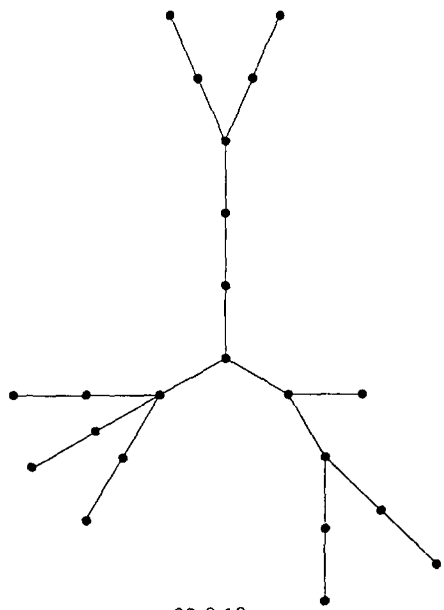
22.0.7



22.0.8



22.0.9



22.0.10