Maximal graphs of path-width k or searching a partial k-caterpillar

Andrzej Proskurowski

CIS-TR-89-17
November 13, 1989

Abstract

We exhibit the fine structure of graphs of path-width k as partial graphs of restricted k-trees that are a generalization of caterpillars.

Department of Computer and Information Science
University of Oregon

Maximal graphs of path-width k or searching a partial k-caterpillar

Andrzej Proskurowski
Department of Computer and Information Science
University of Oregon, Eugene, Oregon 97403, USA

November 9, 1989

Abstract

We exhibit the fine structure of graphs of path-width k as partial graphs of restricted k-trees that are a generalization of caterpillars.

1 Motivation

For a fixed value of the integer parameter k, partial k-trees are exactly subgraphs of those chordal graphs that have at most $k+1$ completely interconnected vertices (see, for instance, [1,7,12]). Thus, partial 1-trees are the acyclic graphs (forests), and partial 2-trees are the series-parallel graphs (graphs with no K_4 minors or homeomorphs).

The class of partial k-trees is identical to the class of graphs of tree-width k [9]. Although replacing 'tree' by 'path' in the above statement does not carry through, we exhibit a relationship between the notion of k-path and path-width k. A k-path has exactly two k-leaves, and the path-width of a graph is defined by a similar modification of the tree-width concept. The path-width can be shown to be one less than the search number of the graph, defined through a variant of the graph searching game (*cf.* [6]).

Partial k-trees have been in the focus of attention in recent years because of their interesting algorithmic properties. Namely, for a large number of inherently difficult (on general graphs) discrete optimization problems, partial k-trees admit a linear time solution algorithm when the value of k is fixed and any partial k-tree is given with its k-tree embedding, [2,11].

2 Definitions and terminology

We will use standard graph theory terminology, as found, for instance, in Bondy and Murty [4]. First, we will define some basic concepts.

A *walk* is a sequence of vertices such that every two consecutive vertices are adjacent. If all the vertices are different, we have a *path*. A walk forms a *cycle* if only its first and last vertices are identical. A set of k vertices, every two of which are adjacent, is called a k-clique. A (minimal) subset of vertices of a graph such that their removal disconnects the graph is a (minimal) *separator*. A *k-tree* is a connected graph such that every minimal separator is a k-clique [10]. Equivalently, the complete graph on $k+1$ vertices (K_{k+1}) is a k-tree, and any k-tree with more vertices, say n, can be constructed from a k-tree with $n-1$ vertices by adding a new vertex adjacent to all vertices of a k-clique of that graph. In this new graph, the added vertex is a *k-leaf*. A *partial k-tree* is any subgraph of a k-tree.

A k-path is a k-tree that has exactly two k-leaves. A partial k-path is a partial subgraph of a k-path. A k-path has exactly k vertex-disjoint paths connecting the two k-leaves.

A *caterpillar* is a tree that can be partitioned into two subgraphs: the *body*, which is a path, and *hairs*, which are leaves adjacent to vertices of the body. Proskurowski [8] generalized this notion to k-trees by defining a *k-caterpillar* as a k-tree that can be likewise partitioned into the body, which is a k-path and hairs, which are k-leaves, each adjacent to all k vertices of a minimal separator of the body. We have the following relation between k-caterpillars and k-paths.

Theorem 1(Proskurowski [8]): The union of k-complete graphs induced by the minimal separators of a k-path is a k-caterpillar. ∎

The *tree-width* of a graph G is defined as the smallest width of a tree decomposition of G. A tree-decomposition of a graph $G = \langle V, E\rangle$ is given by a family of vertex subsets $N = \{V_1, \ldots, V_m\}$ and a tree $T = \langle N, E'\rangle$ such that end vertices of an edge of E are in a common subset and if $v \in V_i \cap V_j$ then v occurs in every set on the path from V_i to V_j in T. The width of the tree decomposition is one less than the maximum cardinality of the V_is. Path decomposition and *path-width* are defined similarly to the tree decomposition and tree-width, with T restricted to a path instead of a general tree. This parameter of a graph is closely related to the *search number* of the graph defined as follows: Markers ('searchers') are being placed on and removed from vertices of a graph. Initially, all edges are considered *contaminated*. An edge is *decontaminated* if the markers are placed on its end vertices. It remains decontaminated unless it becomes *recontaminated* when a marker is removed from a common end vertex of an adjacent contaminated edge. The *search number* of a graph is the minimum number of markers necessary to decontaminate all edges. This definition is one of several variants of the search game, see for instance [6].

3 The main result

In an alternative view of the graph searching process, we can restrict markers to be *moved*, coupling a removal of a marker with its immediately following placement. This view defines sets of vertices of the same size, so that all edges are covered in a manner prescribed by the path decomposition. Indeed, the following statement can be inferred from [3].

Theorem 2: The path-width of a graph is one less than its search number.

Proof: Given a path decomposition of a graph $G = \langle V, E \rangle$ with the vertex sets $V_i, |V_i| \leq k+1, 1 \leq i \leq m$ and path edges $(V_i, V_{i+1}), 1 \leq i < m$, define the following search strategy: Beginning with $i = 1$, place markers on vertices of V_i decontaminating the induced edges of G. Removing markers from vertices not in V_{i+1} does not recontaminate any edges; iterate this step incrementing i until all the edges are decontaminated. No more than $k+1$ markers (the size of the largest set V_i) will be used.

For the proof in the other direction we have to invoke the existence of a search strategy with $k+1$ markers without recontamination (*cf.* [3]). In this 'monotone' search, once a marker is removed from a vertex it will never be placed on this vertex again to decontaminate an edge. During such a search of a graph G with search number $k+1$, define time instances $i, 1 \leq i \leq m$, just after a marker is moved. Let the marked vertex set at time instance i be W_i. (W_0 is defined as the initially marked set of $k+1$ vertices.) The sets W_i define a path decomposition of G of width k. Indeed, each edge of G is between the vertices of some W_i and for the path with edges $(W_i, W_{i+1}), 0 \leq i < m$, the monotone search strategy ensures the intersection property. ∎

We are now able to use the above concepts to exactly characterize the structure of path-width k graphs (those searchable with $k+1$ guards).

Theorem 3: A graph has path-width $\leq k$ if and only if it is a partial k-caterpillar.

We will prove this statement with the help of two lemmata.

Lemma 1: Any k-caterpillar has path-width k.

Proof: Consider a k-caterpillar C and its $(k+1)$-cliques V_i in the order from one end of 'the body' to the other (the cliques involving hairs and sharing one k-clique ordered arbitrarily). The graph $T = \langle N, E' \rangle$ with $N = \{V_i\}$ and $E' = \{(V_i, V_{i+1})\}$ is a path and represents path decomposition of C of width k. ∎

Lemma 2: Maximal graphs of path-width k are exactly k-caterpillars.

Proof: Consider the path $T = \langle N, E' \rangle$ that represents a path decomposition of width k of a graph $G = \langle V, E \rangle$. *Wlog.*, we can assume that the vertex sets $V_i, 1 \le i \le m$ (the nodes of T) are all of order $k+1$. We claim that the graph $C = \langle V, E'' \rangle$ is a k-caterpillar, where $E'' = \{(u,v) : \exists i(u, v \in V_i)\}$.

Let us order the vertices of G by defining $\{v_i\} = V_i - V_{i+1}, 1 \le i < m$, and $\{v_n\} = V_m - V_{m-1}$. The remaining vertices of V_m can be numbered arbitrarily as $v_m \ldots, v_{n-1}$. Notice that the numbering is well defined since $(V_i - V_{i+1}) \cap V_j = \emptyset$ for $j > i+1, 1 \le i < m$. The construction of C from V_m by adding vertices in order $v_{m-1} \ldots, v_1$ ensures that C is a k-tree, since each new vertex v_i is adjacent to all vertices (call this set W_i) of a K_k subgraph of the existing k-tree. What more, W_i constitutes a minimal (v_1, v_n)-separator: By the intersection property of path decomposition, W_i contains some vertices of every path connecting vertices $\{v_n, \ldots, v_i\} - W_i$ with $\{v_{i-1}, \ldots, v_1\}$. Thus, C is a caterpillar. ∎

4 Conclusions

Analysis of minimal separating subgraphs in a k-path lead us to a characterization of graphs with bounded path-width. Since every such graph defines a $(k+1)$-path *via* its embedding in a k-caterpillar, we obtain an interesting relation between the parameters, possibly helpful in analysis of a graph's connectivity (alternative description of a set of vertex-disjoint connecting paths).

References

[1] S. ARNBORG AND A. PROSKUROWSKI, Characterization and Recognition of Partial 3-trees, *SIAM J. Alg. and Discr. Methods 7 (1986) 305-314*

[2] S. ARNBORG AND A. PROSKUROWSKI, Linear Time Algorithms for NP-hard Problems on Graphs Embedded in k-trees, *Discr. Appl. Math. 23 (1989) 11-24*

[3] D. BIENSTOCK and P.D. SEYMOUR, Monotonicity in graph searching, manuscript (May, 1988)

[4] J.A. BONDY AND U.S.R. MURTY, *Graph Theory with Applications*, North Holland (1976)

[5] M.R. GAREY AND D.S. JOHNSON, *Computers and Intractability*, W.H. Freeman and Company, San Francisco (1979)

[6] N. MEGGIDO, S.L. HAKIMI, M.R. GAREY, D.S. JOHNSON, AND C.H. PAPADIMITRIOU, The complexity of searching a graph, *J. ACM 35 (1988) 18-44*

[7] A. PROSKUROWSKI, Recursive graphs, recursive labelings and shortest paths, *SIAM J. Computing 10 (1981) 391-397*

[8] A. PROSKUROWSKI, Separating Subgraphs in k-trees: Cables and Caterpillars, *Discr. Math. 49 (1984), 275-285*

[9] N. ROBERTSON AND P.D. SEYMOUR, Graph minors II: algorithmic aspects of tree-width, *J. Algorithms 7 (1986) 309-322*

[10] D.J. ROSE, Triangulated Graphs and the Elimination Process, *J. Math. Anal. Appl. 32 (1970) 597-609*

[11] P. SCHEFFLER, Linear time algorithms for NP-complete problems for partial k-trees, *R-MATH-03/87* (1987)

[12] A. WALD AND C.J. COLBOURN, Steiner Trees, Partial 2-trees, and Minimum IFI Networks, *Networks 13 (1983), 159-167*