Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 15, no. 2, pp. 253–268 (2011)

On the Threshold-Width of Graphs

Maw-Shang Chang[1] *Ling-Ju Hung*[1] *Ton Kloks*
Sheng-Lung Peng[2]

[1]Department of Computer Science and Information Engineering
National Chung Cheng University
Min-Hsiung, Chia-Yi 621, Taiwan
[2]Department of Computer Science and Information Engineering
National Dong Hwa University
Shoufeng, Hualien 97401, Taiwan

Abstract

For a graph class $\mathcal{G}$, a graph G has $\mathcal{G}$-width k if there are k independent sets $\mathbb{N}_1, \ldots, \mathbb{N}_k$ in G such that G can be embedded into a graph $H \in \mathcal{G}$ such that for every edge e in H which is not an edge in G, there exists an i such that both endpoints of e are in $\mathbb{N}_i$. For the class $\mathfrak{TH}$ of threshold graphs we show that $\mathfrak{TH}$-width is NP-complete and we present fixed-parameter algorithms. We also show that for each k, graphs of $\mathfrak{TH}$-width at most k are characterized by a finite collection of forbidden induced subgraphs.

Submitted:	Reviewed:	Revised:	Accepted:
September 2010	January 2011	March 2011	April 2011
	Final:	Published:	
	May 2011	July 2011	

Article type:
Regular paper

Communicated by:
D. Wagner

This research was supported by the National Science Council of Taiwan, under grants NSC 98–2218–E–194–004 and NSC 98–2811–E–194–006. Ton Kloks is supported by the National Science Council of Taiwan, under grants the National Science Council of Taiwan, under grants NSC 99–2218–E–007–016 and NSC 99–2811–E–007–044.

E-mail addresses: mschang@cs.ccu.edu.tw (Maw-Shang Chang) hunglc@cs.ccu.edu.tw (Ling-Ju Hung) kloks@cs.nthu.edu.tw (Ton Kloks) slpeng@mail.ndhu.edu.tw (Sheng-Lung Peng)

1 Introduction

Definition 1 *Let $\mathcal{G}$ be a class of graphs which contains all cliques. The $\mathcal{G}$*-width *of a graph G is the minimum number k of independent sets $\mathbb{N}_1, \ldots, \mathbb{N}_k$ in G such that there exists an embedding $H \in \mathcal{G}$ of G such that for every edge $e = (x, y)$ in H which is not an edge of G there exists an i with $x, y \in \mathbb{N}_i$.*

We restrict the $\mathcal{G}$-width parameter to classes of graphs that contain all cliques to ensure that it is well-defined for every (finite) graph.

In this paper we investigate the width-parameter for the class $\mathfrak{Th}$ of threshold graphs and henceforth we call it the *threshold-width* or *$\mathfrak{Th}$-width.* If a graph G has threshold-width k then we call G also a *k-probe threshold graph.* Note that graphs of threshold-width one are the graph class called probe threshold graphs (1-probe threshold graphs) introduced in [1]. A linear time algorithm was given in [1] for the recognition of those graphs of threshold-width one. Here we generalize this graph class of threshold-width one, we study the recognition problem of graphs having threshold-width k. We refer to the *partitioned case* of the problem when the collection of independent sets $\mathbb{N}_i$, $i = 1, \ldots, k$, which are not necessarily disjoint, is a part of the input. A collection of independent sets $\mathbb{N}_i$, $i = 1, \ldots, k$, is a *witness* for a partitioned graph. For historical reasons we call the set of vertices $\mathbb{P} = V - \bigcup_{i=1}^{k} \mathbb{N}_i$ the set of *probes* and the vertices of $\bigcup_{i=1}^{k} \mathbb{N}_i$ the set of *nonprobes*.

Threshold graphs are a well-known graph class. They play an important role on computer science, *e.g.,* graph theory and schedule theory. Threshold graphs were discovered independently by researchers working in different areas. Chvátal and Hammer coined the name 'threshold graphs' [4, 5]. They introduced threshold graphs using a concept called 'threshold dimension' and studied these graphs for their application in set packing problems [4]. It is known that many NP-hard problems are solvable in polynomial time on threshold graphs [16]. There is a lot of information about threshold graphs in the book [16], and there are chapters on threshold graphs in the book [7] and in the survey [2].

There are many ways to define threshold graphs. We choose the following way [4, 2]. An isolated vertex in a graph G is a vertex without neighbors. A universal vertex is a vertex that is adjacent to all other vertices.

Theorem 1 ([5, 9, 17, 20]) *A graph is a threshold graph if and only if every induced subgraph has an isolated vertex or a universal vertex.*

Remark 1 *Any threshold graph G has an elimination ordering $(v_1, v_2, \ldots, v_n)$ of the vertex set of G such that for all $i = 1, \ldots, n$, v_i is either an isolated vertex or a universal vertex of $G[\{v_i, \ldots, v_n\}]$, the induced graph of G by $\{v_i, \ldots, v_n\}$.*

We may take the following characterization as a definition [4, 2].

Definition 2 *A graph G is a* threshold graph *if G and its complement $\bar{G}$ are trivially perfect. Equivalently, G is a threshold graph if G has no induced P_4, C_4, nor $2K_2$.*

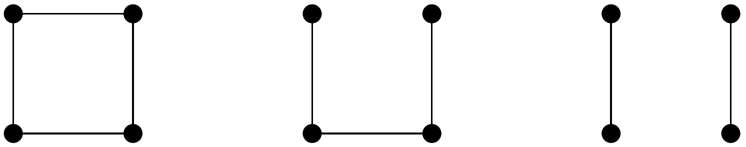

Figure 1: C_4, P_4 and $2K_2$

We end this section with some notational conventions. For two sets A and B we write $A+B$ and $A-B$ instead of $A \cup B$ and $A \setminus B$. We write $A \subseteq B$ if A is a subset of B with possible equality and we write $A \subset B$ if A is a subset of B and $A \neq B$. For a set A and an element x we write $A+x$ instead of $A+\{x\}$ and $A-x$ instead of $A-\{x\}$. In those cases we will make it clear in the context that x is an element and not a set. We use $\mathbf{0}$ to denote a vector of all zeros.

A graph G is a pair $G = (V, E)$ where the elements of V are called the vertices of G and where E is a set of two-element subsets of V, called the edges. We denote edges of a graph as (x, y) and we call x and y the endvertices of the edge. For a vertex x we write $N(x)$ for its set of neighbors and for $W \subseteq V$ we write $N(W) = \bigcup_{x \in W} N(x) - W$ for the neighbors of W. A *module* X is a vertex set of G such that $N(x) - X$ is the same for every $x \in X$. We write $N[x] = N(x) + x$ for the closed neighborhood of x. For a subset W we write $N[W] = N(W) + W$. Usually we will use $n = |V|$ to denote the number of vertices of G and we will use $m = |E|$ to denote the number of edges of G.

For a graph $G = (V, E)$ and a subset $S \subseteq V$ of vertices we write $G[S]$ for the subgraph *induced* by S, that is the graph with S as its set of vertices and with those edges of E that have both endvertices in S. For a subset $W \subseteq V$ we will write $G - W$ for the graph $G[V - W]$ and for a vertex x we will write $G - x$ rather than $G - \{x\}$. A vertex x of a graph G is *isolated* if its neighborhood is the empty set.

2 A finite characterization

In this section we show that the class of graphs with $\mathfrak{Th}$-width at most k is characterized by a finite collection of forbidden induced subgraphs.

Lemma 1 *A graph G is a threshold graph if and only if it has a binary tree-decomposition (T, f), where f is a bijection from the vertices of G to the leaves of T. Every internal node of T, including the root is labeled either as a join– or a union-node. For every internal node the right subtree consists of a single leaf. Two vertices are adjacent in G if and only if their lowest common ancestor in T is a join-node.*

Proof: According to Theorem 1 a graph is a threshold graph if and only if every induced subgraph has either an isolated vertex or a universal vertex. Given a threshold graph G, G has either an isolated vertex or a universal vertex.

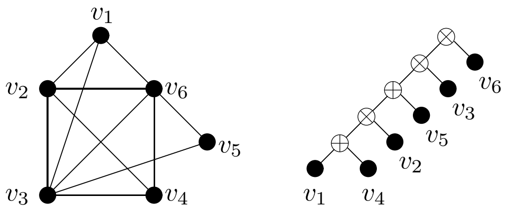

Figure 2: A threshold graph and its binary tree-decomposition where $\otimes$ denotes a join-node and $\oplus$ denotes a union-node.

Let the isolated vertex or the universal be v. We may recursively construct a decomposition tree (T, f) for G by the following approach. First create the root r of T, make a copy of v, say v' and attach v' as the right child of r. If v is an isolated vertex, label r as a union-node. Otherwise label r as a join-node. Let the left child of r be the root of the decomposition tree of $G - v$. By induction, this shows that if G is a threshold graphs, it has a binary tree-decomposition. Now suppose that a graph G has a binary tree-decomposition (T, f) defined as above. For any induced subgraph G' of G, we obtain tree-decomposition (T', f') of G' by the following steps. Remove all leaves of T that are not mapped from vertices in G', and then remove all internal nodes that have no right subtree and add an edge between its parent and its left child. It is easy to see that the vertex mapped to the leaf adjacent to the root of T' is either an isolated vertex or a universal vertex. This shows that if G has a tree-decomposition tree (T, f) as defined above, G is a threshold graph. Suppose that u and v are two vertices in a threshold graph G. Let (T, f) be a tree-decomposition of G. Let u' (resp. v') be a copy of u (resp. v) in T. Let r_u (resp. r_v) be the internal node in T that u' (resp. v') is adjacent to. Assume that r_u is an ancestor of r_v. We see that r_u is the lowest common ancestor of u' and v'. Node r_u is a join-node if and only if u is a universal vertex in the graph G' induced by the set of vertices mapped to the leaves in the right subtree of r_u, u and v are adjacent in G' and G. Similarly r_u is a union-node if and only if u is an isolated vertex in G', u and v are not adjacent in G' and G. This proves the lemma. $\square$

Remark 2 *Note that the tree-decomposition used here is not the same as those "tree-decomposition" defined for the well-known notion of treewidth.*

In the next theorem, we will prove that for every k the class of graphs with $\mathfrak{TH}$-width at most k has finite forbidden induced subgraphs. To prove it we first review the technique introduced by Pouzet [24].

Let $T_1, T_2, \ldots$ be a collection of rooted binary trees with nodes labeled from some finite set. We write $T_i \prec T_j$ if there exists an injective map h from the nodes of T_i to the nodes of T_j such that

1. the label of a node a in T_i is equal to the label of the node $h(a)$ in T_j, and
2. for every pair of nodes a and b in T_i, their lowest common ancestor is mapped to the lowest common ancestor of $h(a)$ and $h(b)$ in T_j, and

3. if a and b are nodes of T_i with lowest common ancestor c such that a is in the left subtree of c and b is in the right subtree of c then $h(a)$ is in the left subtree of $h(c)$ and $h(b)$ is in the right subtree of $h(c)$ in T_j.

Let $T_1, T_2, \ldots$ be an infinite sequence of rooted binary trees with nodes labeled from some finite set. Kruskal's theorem [15] states that there exist integers $i < j$ such that $T_i \prec T_j$.

Theorem 2 *For every k the class of graphs with $\mathfrak{Th}$-width at most k is characterized by a finite collection of forbidden induced subgraphs.*

Proof: The class of graphs with $\mathfrak{Th}$-width at most k is hereditary, *i.e.*, a graph has $\mathfrak{Th}$-width at most k, any its induced subgraph also has $\mathfrak{Th}$-width at most k. Let k be fixed. Assume that the class of $\mathfrak{Th}$-width at most k has an infinite collection of minimal forbidden induced subgraphs, say $G_1, G_2, \ldots$. In each G_i single out one vertex r_i and let $G'_i = G_i - r_i$. Then G'_i has $\mathfrak{Th}$-width at most k, thus there are independent sets $\mathbb{N}_1^{(i)}, \ldots, \mathbb{N}_k^{(i)}$ in G'_i such that G'_i can be embedded into a threshold graph H_i by adding certain edges between vertices that are pairwise contained in some $\mathbb{N}_\ell^{(i)}$. For each i consider a binary tree-decomposition (T_i, f_i) for H_i as stipulated in Lemma 1. Each leaf is labeled by a 0/1-vector with k entries. The j^{th} entry of this vector is equal to 0 or 1 according to whether the vertex is contained in $\mathbb{N}_j^{(i)}$ or not. Thus two vertices are adjacent in G'_i if and only if their lowest common ancestor is a join-node and their vectors are disjoint.

We give each leaf an additional 0/1-label that indicates whether the vertex that is mapped to that leaf is adjacent to r_i or not.

When we apply Kruskal's theorem [15] to the labeled binary trees T_i that represent the graphs G'_i we may conclude that there exist $i < j$ such that G'_i is an induced subgraph of G'_j. But then we must also have that G_i is an induced subgraph of G_j. This is a contradiction because we assume that the graphs G_i are minimal forbidden induced subgraphs. This proves the theorem. □

Remark 3 *Higman's lemma [10] preludes Kruskal's theorem. It deals with finite sequences over a finite alphabet instead of trees. Instead of Kruskal's theorem we could have used Higman's lemma to prove Theorem 2.*

3 $\mathfrak{Th}$-width is NP-complete

Let $\mathfrak{T}$ be the class of complete graphs (cliques). The following theorem is proved in [3] that $\mathfrak{T}$-width is NP-complete. We apply it to prove that $\mathfrak{Th}$-width is also NP-complete.

Theorem 3 ([3]) *$\mathfrak{T}$-Width is NP-complete.*

Theorem 4 *$\mathfrak{Th}$-width is NP-complete.*

Proof: Assume there is a polynomial-time algorithm to compute $\mathfrak{TH}$-width. We show that we can use that algorithm to compute $\mathfrak{T}$-width. Let G be a graph for which we wish to compute $\mathfrak{T}$-width. Construct a graph G' by adding a clique C with n^2 vertices. Make all vertices of C adjacent to all vertices of G. Add one more vertex ω and make ω also adjacent to all vertices of G. Consider two nonadjacent vertices x and y of G. We see that for any $z \in C$, $\{\omega, x, y, z\}$ induces a C_4 in G'. Note that C_4 is a forbidden induced subgraph of threshold graphs. In any embedding of G' into a threshold graph, either x and y are adjacent or ω is adjacent to all vertices of C. However, to make ω adjacent to all vertices of C, we need at least n^2 independent sets. Obviously, making a clique of G embeds G' into a threshold graph, namely the complement of a star and a collection of isolated vertices. This embedding needs less than n^2 independent sets. This proves the theorem. □

Remark 4 *The above theorem can be extended to that for the graph class $\mathcal{G}$ of graphs having no induced C_4, $\mathcal{G}$-width is also NP-complete.*

4 $\mathfrak{TH}$-width is fixed-parameter tractable

A problem is called *fixed-parameter tractable* (FPT) if it can be solved in $f(k) \cdot n^{O(1)}$ time, where n denotes the size of the instance and $f(k)$ is any function of the parameter k. We call algorithms which run in $f(k) \cdot n^{O(1)}$ time *fixed-parameter algorithms* [19]. In this section we show that for constant k, k-probe threshold graphs can be recognized in $O(n^3)$ time.

The most natural way to express and classify graph-theoretic problems is by means of logic. In monadic second order logic a (finite) sentence is a formula that uses quantifiers $\forall$ and $\exists$. The quantification is over vertices, edges, and subsets of vertices and edges. Relational symbols are $\neg$, $\in$, $=$, $\wedge$, $\vee$, $\subseteq$, $\cup$, $\cap$, and the logical implication $\Rightarrow$. Some of these are superfluous. Although the minimization or maximization of the cardinality of a subset is not part of the logic, one usually includes them.

A restricted form of this logic is where one does not allow quantification over subsets of edges. The C_2MS-logic is such a restricted monadic second-order logic where one can furthermore use a test whether the cardinality of a subset is even or odd.

Courcelle proved that problems that can be expressed in C_2MS-logic can be solved in $O(n^3)$ time for graphs of bounded rankwidth [6].

The following theorem is a monadic second-order characterization. We will show that k-probe threshold graphs have bounded rankwidth shortly.

Theorem 5 *A graph $G = (V, E)$ has threshold-width at most k if and only if there exist k independent sets $\mathbb{N}_i$, $i = 1, \ldots, k$, such that for every $W \subseteq V$, $G[W]$ has an isolated vertex or a vertex ω such that for every $y \in W - \omega$ either ω is adjacent to y or there exists $i \in \{1, \ldots, k\}$ with $\{\omega, y\} \subseteq \mathbb{N}_i$.*

Proof: This is inferred by Theorem 1 and Definition 1. □

Remark 5 *According to the above theorem, given a graph $G = (V, E)$, G has threshold-width at most k if and only if the following monadic second order logic formula is satisfied.*

$$\begin{array}{l}\exists \mathbb{N}_1, \ldots \mathbb{N}_k \big(\bigwedge_{1 \leq i \leq k} ((\mathbb{N}_i \subseteq V) \wedge (\forall u, v \in \mathbb{N}_i (\neg edg(u, v)))) \\ \forall W . W \subseteq V \wedge \exists w . w \in W \wedge \\ ((\forall y . y \in W \wedge ((y = w) \vee \neg edg(w, y)) \vee \\ (\forall y . y \in W \wedge ((y = w) \vee (edg(w, y) \vee (\bigvee_{1 \leq i \leq k} (w \in N_i \wedge y \in N_i)))))\big)\end{array}$$

where $edg(u, v)$ is true if and only if u and v are adjacent in G.

Definition 3 ([23, 21]) *A* rank-decomposition *of a graph $G = (V, E)$ is a pair (T, τ) where T is a ternary tree and τ a bijection from the leaves of T to the vertices of G. Let e be an edge in T and consider the two sets A and B of leaves of the two subtrees of $T - e$. Let M_e be the submatrix of the adjacency matrix of G with rows indexed by the vertices of A and columns indexed by the vertices of B. The* width of e *is the rank over $GF(2)$ of M_e. The* width of (T, τ) *is the maximum width over all edges e in T and the* rankwidth *of G is the minimum width over all rank-decompositions of G.*

Computing the rankwidth of a graph is NP-complete [11] but it is fixed-parameter tractable. This can be seen in various ways: Oum proved that there is a finite obstruction set for fixed-parameter rankwidth [22]. Now, note that Schrijver gave a general algorithm to minimize a class of submodular functions which uses the ellipsoid method [26, Chapter 45]. He turns this into a 'combinatorial algorithm' for a seemingly larger class of submodular functions, in [25]. Using this result, in [11] a combinatorial fixed-parameter algorithm was developed for computing the rankwidth of matroids.

Lemma 2 *Threshold graphs have rankwidth at most one.*

Proof: The class of graphs with rankwidth at most 1 is exactly the class of distance-hereditary graphs [21]. Every threshold graph is distance hereditary (see, *e.g.*, [2, 9, 17]). □

Theorem 6 *k-Probe threshold graphs have rankwidth at most 2^k.*

Proof: Suppose that G is a k-probe threshold graph and H is an embedding of G, *i.e.*, H is a threshold graph. Consider a rank-decomposition (T, τ) with width 1 for H. Consider an edge e in T and assume that M_e is an all-1s-matrix. Each independent set $\mathbb{N}_i$ creates a 0-submatrix in M_e. If $k = 1$ this proves that the rankwidth of G is at most 2. In general, for $k \geq 0$, note that there are at most 2^k different neighborhoods from one leaf-set of $T - e$ to the other. It follows that the rank of M_e is at most 2^k. □

Remark 6 *Note that the matrix M_e has indeed rank 2^k in the worst case.*

Theorem 7 *For each $k \geq 0$ there exists an $O(n^3)$ algorithm which checks whether a graph G with n vertices is a k-probe threshold graph. Thus $\mathfrak{TH}$-width $\in FPT$.*

Proof: k-Probe threshold graphs have bounded rankwidth. C_2MS-Problems can be solved in $O(n^3)$ time for graphs of bounded rankwidth [6, 12, 21]. By Theorem 5, the recognition of k-probe threshold graphs is such a problem.

Alternatively, the theorem is also proved by using the finite collection of forbidden induced subgraphs. Note however that this proof is non-constructive; Kruskal's theorem does not provide the forbidden induced subgraphs. □

A fortiori, Theorem 7 holds as well when the collection of independent sets $\mathbb{N}_1, \ldots, \mathbb{N}_k$ is a part of the input. Thus for each k there is an $O(n^3)$ algorithm that checks whether a graph G, given with k independent sets $\mathbb{N}_i$, can be embedded into a threshold graph.

There are a few drawbacks to this solution. First of all, Theorem 7 only shows the *existence* of an $O(n^3)$ recognition algorithm; *a priori*, it is unclear how to obtain an algorithm explicitly. Furthermore, the constants involved in the algorithm make the solution impractical; already there is an exponential blow-up when one moves from threshold-width to rankwidth.

5 Recognition of Partitioned k-probe threshold graphs

In this section we show that there exists an explicit, linear-time algorithm for the recognition of partitioned k-probe threshold graphs.

Let $(G, \mathcal{N})$ be a partitioned k-probe threshold graph, consisting of a graph G and a witness $\mathcal{N}$ with k independent sets $\mathbb{N}_1, \ldots, \mathbb{N}_k$ of G.

Lemma 3 *If G has an isolated vertex x then G is partitioned k-probe threshold if and only if $G - x$ is partitioned k-probe threshold with the same induced collection of independent sets. The same statement holds as well for the unpartitioned case.*

Proof: Assume G is k-probe threshold. Consider an embedding H of G. Then $H - x$ is an embedding of $G - x$. Thus $G - x$ is k-probe threshold.
Assume $G - x$ is k-probe threshold. Let H' be an embedding of $G - x$. Then we obtain an embedding of G by adding x as an isolated vertex to H'. □

Theorem 8 *For every k there exists a linear-time algorithm to check whether a pair $(G, \mathcal{N})$, where G is a graph and $\mathcal{N}$ a collection of k independent sets in G, is a partitioned k-probe threshold graph.*

Proof: Assume that $(G, \mathcal{N})$ is a partitioned graph and let H be an embedding of G. If H has an isolated vertex x, then x is also isolated in G since H is an embedding of G. By Lemma 3 any isolated vertex may be safely removed from G.

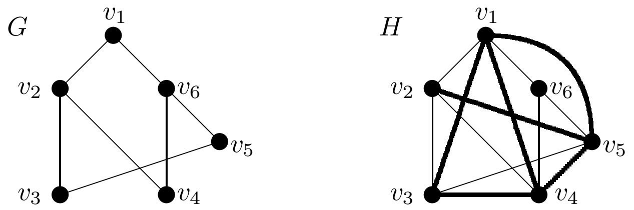

Figure 3: A partitioned 3-probe threshold graph $(G,\mathcal{N})$ where $\mathcal{N} = \{\mathbb{N}_1,\mathbb{N}_2,\mathbb{N}_3\}$, $\mathbb{N}_1 = \{v_1, v_3, v_4\}$, $\mathbb{N}_2 = \{v_1, v_4, v_5\}$, and $\mathbb{N}_3 = \{v_2, v_5\}$. H is a threshold embedding of G where the thin lines in H are those edges in G and those thick lines in H are new edges being added.

Now we may assume that any embedding H has no isolated vertices. Since H is a threshold graph, it has no induced $2K_2$, H is connected. By Theorem 1 H has a universal vertex ω. We call ω a 'probe universal vertex' of $(G,\mathcal{N})$ if for every nonneighbor z there is an independent set in $\mathcal{N}$ which contains both ω and z. Thus any partitioned k-probe threshold graph has an isolated vertex or a probe universal vertex. Finally, observe the following: if ω is a probe universal vertex then G is k-probe threshold if and only if $G-\omega$ is k-probe threshold, since we may add ω as a universal vertex to any embedding of $G-\omega$ and obtain an embedding of G. Let $(\omega_1, \omega_2, \ldots, \omega_n)$ be an elimination ordering such that ω_i is either an isolated vertex or a probe universal vertex of $G[\{\omega_i, \omega_{i+1}, \ldots, \omega_n\}]$. Since k is a constant, such an elimination ordering can be obtained in linear time. □

Remark 7 *Note that the algorithm described in Theorem 8 is fully polynomial. This proves that the 'sandwich problem,' studied by Golumbic* et al.*, in [8], is polynomial for threshold graphs. Given $G^1 = (V, E^1)$ and $G^2 = (V, E^2)$ where $E^1 \subseteq E^2$, the* sandwich problem *asks whether there exists a graph $G = (V, E)$, $E^1 \subseteq E \subseteq E^2$, where G is in a specific graph class $\mathcal{G}$.*

6 A fixed-parameter algorithm to compute $\mathfrak{Th}$-width

Assume that $(G,\mathcal{N})$ is a connected partitioned k-probe threshold graph with witness

$$\mathcal{N} = \{\mathbb{N}_i \mid i = 1\ldots,k\}$$

and let H be an embedding. The *label* $L(x)$ of a vertex x is the 0/1-vector of length k with the i^{th} entry $L(x)[i]$ equal to 1 if and only if $x \in \mathbb{N}_i$. We write $L(x) \leq L(y)$ if $L(x)[i] \leq L(y)[i]$ for all $i = 1, \ldots, k$. We write $L(x) \perp L(y)$ if there is no i with $L(x)[i] = L(y)[i] = 1$.

Definition 4 *A witness $\mathcal{N}$ is* well-linked *if for every $i = 1, \ldots, k$, every vertex $x \notin \mathbb{N}_i$ has a neighbor in $\mathbb{N}_i$.*

Lemma 4 *Every k-probe threshold graph has a witness with k independent sets which is well-linked.*

Proof: Starting with any witness, repeatedly add a vertex x to an independent set $\mathbb{N}_i$ if it has no neighbor in that set. □

Consider the equivalence relation $\equiv$ defined by $x \equiv y$ if $N(x) = N(y)$. Denote the equivalence class of a vertex x by (x). Define the partial order $\preceq$ by:

$$(x) \preceq (y) \quad \text{if} \quad N(x) \subseteq N(y).$$

Likewise, we consider the equivalence relation $\equiv'$ defined by $x \equiv' y$ if $N[x] = N[y]$. The equivalence class of a vertex x under this relation is denoted by $[x]$. We consider the partial order defined by:

$$[x] \preceq [y] \quad \text{if} \quad N[x] \subseteq N[y].$$

Lemma 5 *Assume $(G, \mathcal{N})$ is a k-probe clique with a well-linked witness $\mathcal{N}$. Then*

$$(x) \preceq (y) \quad \Leftrightarrow \quad L(x) \geq L(y) \neq \mathbf{0}.$$

Proof: Since $(G, \mathcal{N})$ is a k-probe clique, there exists an embedding H of G with respect to the witness $\mathcal{N}$, H is a clique. Note that any probe w in a k-probe clique must be with $L(w) = \mathbf{0}$.

Assume $L(x) \geq L(y) \neq \mathbf{0}$. Both x and y are nonprobes and if y is in $\mathbb{N}_i$ for some i, so is x, that is, they are not adjacent in G. Let $z \in N_G(x)$. Since x and y are not adjacent, $z \neq y$. If $z \notin N_G(y)$, there exists a j with $\{z, y\} \subseteq \mathbb{N}_j$. Now $L(x) \geq L(y)$ implies $x \in \mathbb{N}_j$, which contradicts that z is adjacent to x. Hence $(x) \preceq (y)$.

Assume $(x) \preceq (y)$ in G. Thus $N_G(x) \subseteq N_G(y)$ and x and y are not adjacent in G. Since H is a clique, x and y are adjacent in H, both x and y are nonprobes. Assume that $\neg(L(x) \geq L(y))$. There exists an i with $y \in \mathbb{N}_i$ and $x \notin \mathbb{N}_i$. Since $\mathcal{N}$ is well-linked there exists a vertex $z \in N_G(x) \cap \mathbb{N}_i$. Since $(x) \preceq (y)$, $z \in N_G(y)$. But this contradicts $\{y, z\} \subseteq \mathbb{N}_i$. □

Note that Definition 2 is equivalent to the following characterization.

Theorem 9 ([5, 16]) *A graph H is a threshold graph if and only if for every pair of vertices x and y, $N(x) \subseteq N[y]$ or $N(y) \subseteq N[x]$.*

In other words, a graph $G = (V, E)$ is a threshold graph if and only if there is a *total order* of the vertices $[x_1, \ldots, x_n]$, *i.e.*, a *chain*, such that:

$$1 \leq i < j \leq n \quad \Rightarrow \quad N(x_i) \subseteq N[x_j].$$

Theorem 10 *Let $(G,\mathcal{N})$ be a k-probe threshold graph with a well-linked witness $\mathcal{N}$ and let H be an embedding of G with respect to the witness $\mathcal{N}$, H is a threshold graph. For every nonadjacent pair x and y in G with $N_H(x) \subseteq N_H[y]$:*

$$(x) \preceq (y) \quad \Leftrightarrow \quad L(x) \geq L(y).$$

Proof: Assume $L(x) \geq L(y)$. Let $z \in N_G(x)$. Then $z \in N_H[y]$. If $L(y) = \mathbf{0}$, y is a probe, $N_G[y] = N_H[y]$, $z \in N_G[y]$. Since $N_G(x) \subseteq N_H(x) \subseteq N_H[y] = N_G[y]$, $(x) \preceq (y)$. Now assume that $L(y) \neq \mathbf{0}$, both x and y are nonprobes. Since x and y are not adjacent, $z \neq y$. Thus $z \in N_H(y)$. If $z \notin N_G(y)$, then there exists an i with $\{z, y\} \subseteq \mathbb{N}_i$. Now $L(x) \geq L(y)$ implies that also $x \in \mathbb{N}_i$, which contradicts that z is adjacent to x. Hence $(x) \preceq (y)$.
Assume $(x) \preceq (y)$, that is, $N_G(x) \subseteq N_G(y)$. If y is a probe, $L(y) = \mathbf{0}$. It is true that $L(x) \geq L(y)$. Suppose that y is a nonprobe. *A fortiori*, x and y are not adjacent in G. Assume $\neg(L(x) \geq L(y))$. Then there exists an i with $y \in \mathbb{N}_i$ and $x \notin \mathbb{N}_i$. Since $\mathcal{N}$ is well-linked, there exists a vertex $z \in N_G(x) \cap \mathbb{N}_i$. Since $(x) \preceq (y)$, $z \in N_G(y)$, contradicting that z and y are both in $\mathbb{N}_i$. □

For completeness sake we note the following.

Lemma 6 *Let $(G,\mathcal{N})$ be a k-probe threshold graph with a well-linked witness $\mathcal{N}$ and let H be an embedding of G with respect to the witness $\mathcal{N}$, H is a threshold graph. For every adjacent pair x and y in G with $N_H(x) \subseteq N_H[y]$, if $[x] \preceq [y]$ and $x \neq y$, then*

$$\forall i : L(y)[i] = 1 \Rightarrow N_G(x) \cap \mathbb{N}_i = \{y\}.$$

Proof: Since x and y are adjacent, we have that $L(x) \perp L(y)$. Assume that $y \in \mathbb{N}_i$ for some $i \in \{1, \ldots, k\}$. Thus $x \notin \mathbb{N}_i$. Since $\mathcal{N}$ is well-linked, there exists a vertex $z \in N(x) \cap \mathbb{N}_i$. Since $N_G[x] \subseteq N_G[y]$, $z \in N_G[y]$. But then we must have $z = y$, otherwise z and y are nonadjacent. □

Definition 5 *A* true − *or* false module *is a set of vertices such that every pair is a true – or false twin, respectively.*[1] *A* k-probe module *is either a false module with at least 3 vertices or a true module with at least $k + 3$ vertices.*

Lemma 7 *Let S be a k-probe module. Then G has $\mathfrak{TH}$-width at most k if and only if $G - x$ has $\mathfrak{TH}$-width at most k for any $x \in S$.*

Proof: If G is k-probe threshold then so is $G - x$ for any vertex x. Let $x \in S$ and assume that $G - x$ is a k-probe threshold graph. Let H be an embedding of $G - x$. First assume that S is a false module with at least three vertices. Let $y \in S - x$. If y is in the independent set, then we can let x be a copy of y. Assume that all vertices of $S - x$ are in the clique of H. Since $S - x$ has at least two vertices, they must be nonprobes. We can let x be a copy of either of them.

[1] A *true twin* is a pair of vertices x and y with $N[x] = N[y]$. A *false twin* is a pair of vertices x and y with $N(x) = N(y)$.

Assume S is a true module with at least $k+3$ vertices. Then at least $k+1$ vertices are in the clique C of H. Let z be a vertex of $S \cap C$ with a minimal closed neighborhood in H. Assume that z has a neighbor u in H which is not a neighbor of z in G. Then u is a neighbor of every vertex of $S \cap C$ in H, but not in G. Since every pair of vertices $a, b \in S$ is adjacent in G, $L(a) \perp L(b)$. It follows that u must be in at least $k+1$ independent sets, which is a contradiction. Thus $N_H(z) = N_G(z)$, and we can let x be a copy of z. □

Definition 6 *A vertex x is* maximal *if there exists no $(y) \neq (x)$ with $(x) \preceq (y)$ and there exists no $[y] \neq [x]$ with $[x] \preceq [y]$.*

Lemma 8 *Assume that G is a k-probe threshold graph without k-probe module. Then there are at most $2^{k+1} + k$ maximal vertices.*

Proof: Consider a well-linked embedding H. By Theorem 9 there is a chain order of its vertices. Let $M_0, M_1, \ldots$ be the equivalence classes in H of vertices with the same open or closed neighborhoods. Assume they are ordered such that $N[x_i] \supseteq N(x_{i+1})$ for each $x_i \in M_i$ and $x_{i+1} \in M_{i+1}$, for $i = 0, 1, \ldots$. Thus if H is connected, M_0 is the set of universal vertices in H. We call these equivalence classes $M_0, M_1, \ldots$ the *levels* of the embedding. Thus a level contained in the clique induces a k-probe clique in G and a level contained in the independent set induces an independent set in G.
Consider the partition of each level M_s into sets of vertices with the same label. We call the sets of the partition of a level M_s the *label-sets* of M_s. Notice that each label-set is a module in G. Since there is no k-probe module, each label-set of nonprobes has at most 2 vertices and the label-set of probes has at most $k+2$ vertices. Thus

$$|M_s| \leq 2(2^k - 1) + (k+2) = 2^{k+1} + k.$$

By Theorem 10 a vertex $x \in M_s$ is maximal if it has a label $L(x)$ such that all other label-sets $L' \leq L(x)$ in $M_0, \ldots, M_s$ are empty. It follows that there are at most $\sum_{i=0}^{k} \binom{k}{i} = 2^k$ label-sets of maximal vertices, at most $2^k - 1$ of maximal nonprobes, each containing at most 2 elements, and at most one label-set of maximal probes, containing at most $k+2$ elements. Thus the number of maximal elements is bounded by $2^{k+1} + k$. □

Lemma 9 *Assume G is a k-probe threshold graph without isolated vertices and without k-probe module. There exists a set Υ, of size $|\Upsilon| \leq 2^{2(k+1)}$ such that any well-linked embedding of G has its set of universal vertices $M_0 \subseteq \Upsilon$. This set Υ can be computed in linear time.*

Proof: Since G has no isolated vertices, H has a set of universal vertices M_0. Start with $\Upsilon = \varnothing$. Repeatedly compute the set of maximal vertices in G, add them to Υ, and delete them from the graph. After at most 2^k repetitions, each label-set of M_0 is contained in Υ. Since each set of maximal elements has at most $2^{k+1} + k$ vertices,

$$|\Upsilon| \leq 2^k(2^{k+1} + k) \leq 2^{2k+1} + 2^{2k} \leq 2^{2(k+1)}.$$

□

Definition 7 *Let $(G, \mathcal{N})$ be a partitioned k-probe graph where $G = (V, E)$ and $\mathcal{N} = \{\mathbb{N}_1, \mathbb{N}_2, \ldots, \mathbb{N}_k\}$. A vertex subset $W \subseteq V$ is called a* partitioned probe clique *if for every pair of $x, y \in W$, $x \neq y$, either $(x, y) \in E$ or $x, y \in \mathbb{N}_i$ for some $i \in \{1, 2, \ldots, k\}$.*

Definition 8 *A* probe universal set *is a set U of labeled vertices such that for every vertex $x \notin U$, there is a label for x such that $U + x$ is a partitioned probe clique.*

Lemma 10 *Let U be a probe universal set and let $x \notin U$ be a vertex with minimal neighborhood such that $U' = U + N(x)$ is probe universal with the same number of nonempty label-sets as U. Then there exists an embedding such that U is universal if and only if there exists an embedding such that U' is universal.*

Proof: By definition, the label-sets of U' are modules that extend the label-sets of U. This proves the lemma. □

Theorem 11 *For each k, there exists an $O(n^2)$-time algorithm for the recognition of k-probe threshold graphs.*

Proof: We may assume that G has no k-probe module. By Lemma 9 there exists a constant number of feasible probe universal sets. By Lemma 10, if there exists a vertex x that can be labeled such that $N(x)$ extends the probe universal set in a way that does not increase the number of nonempty label-sets in the probe universal set, then the algorithm can greedily extend the probe universal set with $N(x)$. Next the algorithm removes the vertex x and tries to find another greedy extension.

If there are no more greedy extensions, the algorithm computes the set Υ as in Lemma 9 in the graph minus the probe universal set, and chooses one of the constant number of subsets as an extension of the probe universal set. Notice that there can be at most 2^k extensions that increase the number of label-sets.

Since the computation of maximal vertices can be done in $O(n^2)$ time, the algorithm can be implemented to run in $O(n^2)$ time. □

Remark 8 *Perhaps it is a bit surprising that we do not have to treat the different components of the graph separately.*

7 Concluding remarks

The recognition problem of probe interval graphs was introduced by Zhang *et al.* [27, 18]. This problem stems from the physical mapping of chromosomal DNA of humans and other species. Since then probe graphs of many other graph classes have been investigated by various authors. In this paper we generalized the concept to the graph-class-width parameters. So far, we have limited our research to classes of graphs that have bounded rankwidth.

In [14], we derived a fixed-parameter algorithm that solves a similar problem for the class of trivially perfect graphs. It is well-known that every threshold graph is trivially perfect. Obviously, this does not imply that the algorithm for trivially perfect graphs can be used for threshold graphs. In fact, a similar, elegant solution as the one that we obtained in this paper cannot work for threshold graphs.

For the classes of blockgraphs, threshold graphs, trivially perfect graphs, and cographs we were able to show that the width parameter is fixed-parameter tractable [3, 14, 13]. One of the classes for which this is still open is the class of distance-hereditary graphs. We are unaware of a monadic second-order formulation that describes the distance-hereditary width. Consider a decomposition tree of bounded rankwidth. The 'twinset' of a branch is defined as the subset of vertices that are mapped to the leaves of that branch, and that have neighbors in the rest of the graph (outside the branch). It is not difficult to show that for bounded rankwidth, the graphs that arise as twinsets constitute a class of graphs that is characterized by a finite collection of forbidden induced subgraphs. (For rankwidth one this is the class of cographs.) The same holds true for graphs of bounded DH-width. So far, we have not been able to describe the class of graphs as tree-extensions of these twinsets.

References

[1] D. Bayer, V. B. Le, and H. N. de Ridder. Probe threshold and probe trivially perfect graphs. *Theoretical Computer Science*, 410:4812–4822, 2009.

[2] A. Brandstädt, V. B. Le, and J. P. Spinrad. *Graph Classes: A Survey.* SIAM Monographs on Discrete Mathematics and Applications. SIAM, Philadelphia, 1999.

[3] M.-S. Chang, L.-J. Hung, T. Kloks, and S.-L. Peng. Block-graph width. *Theoretical Computer Science*, 412:2496–2502, 2011.

[4] V. Chvátal and P. L. Hammer. Set-packing and threshold graphs. Technical Report CORR 73-21, Computer Science Department, University of Waterloo, Canada, 1973.

[5] V. Chvátal and P. L. Hammer. Aggregation of inequalities in integer programming. In P. L. Hammer, E. L. Johnson, B. H. Korte, and G. L. Nemhouser, editors, *Studies in Integer Programming*, volume 1 of *Annals of Discrete Mathematics.* North-Holland, Amsterdam, 1977.

[6] B. Courcelle and S. Oum. Vertex minors, monadic second-order logic, and a conjecture by Seese. *Journal of Combinatorial Theory, Series B*, 97:91–126, 2007.

[7] M. C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs.* Academic Press, New York, 1980.

[8] M. C. Golumbic, H. Kaplan, and R. Shamir. Graph sandwich problems. *Journal of Algorithms*, 19:449–473, 1995.

[9] F. Harary. The structure of threshold graphs. *Decisions in Economics and Finance*, 2:169–172, 1979.

[10] G. Higman. Ordering by divisibility in abstract algebras. In *Proceedings of the London Mathematical Society*, volume s3-2, pages 326–336, 1952.

[11] P. Hliněný and S. Oum. Finding branch-decompositions and rank-decompositions. *SIAM Journal on Computing*, 38:1012–1032, 2008.

[12] P. Hliněný, S. Oum, and D. Seese. Width parameters beyond tree-width and their applications. *The Computer Journal*, 51:326–362, 2008.

[13] L.-J. Hung and T. Kloks. On some simple widths. In *the 4th International Workshop of Algorithms and Computation (WALCOM 2010)*, volume 5942 of *Lecture Notes in Computer Science*, pages 204–215. Springer, 2010.

[14] L.-J. Hung, T. Kloks, and C. M. Lee. Trivially-perfect width. In *the 20th International Workshop of Combinatorial Algorithms (IWOCA 2009)*, volume 5874 of *Lecture Notes in Computer Science*, pages 301–311. Springer, 2009.

[15] J. B. Kruskal. Well-quasi-ordering, the tree theorem, and Vazsonyi's conjecture. *Transactions of the American Mathematical Society*, 95:210–225, 1960.

[16] N. V. R. Mahadev and U. N. Peled. *Threshold Graphs and Related Topics*, volume 56 of *Annals of Discrete Mathematics.* Elsevier, North-Holland, Amsterdam, 1995.

[17] P. Manca. On a simple characterization of threshold graphs. *Rivista di Matematica per le Scienze Economiche e Sociali*, 2:3–8, 1979.

[18] F. R. McMorris, C. Wang, and P. Zhang. On probe interval graphs. *Discrete Applied Mathematics*, 88:315–324, 1998.

[19] R. Niedermeier. *Invitation to Fixed-Parameter Algorithms.* Oxford Lecture Series in Mathematics and its Applications. Oxford University Press, 2006.

[20] J. Orlin. The minimal integral separator of a threshold graph. In P. L. Hammer, E. L. Johnson, B. H. Korte, and G. L. Nemhouser, editors, *Studies in Integer Programming*, volume 1 of *Annals of Discrete Mathematics*, pages 415–419. North-Holland, Amsterdam, 1977.

[21] S. Oum. *Graphs of bounded rank-width.* PhD thesis, Princeton University, 2005.

[22] S. Oum. Rank-width and well-quasi-ordering. *SIAM Journal on Discrete Mathematics*, 22:666–682, 2008.

[23] S. Oum and P. Seymour. Approximating clique-width and branch-width. *Journal of Combinatorial Theory, Series B*, 96:514–528, 2006.

[24] M. Pouzet. Applications of well-quasi-ordering and better-quasi-ordering. In I. Rival, editor, *Graphs and orders*, pages 503–519. D. Reidel Publishing Company, 1985.

[25] A. Schrijver. A combinatorial algorithm minimizing submodular functions in strongly polynomial time. *Journal of Combinatorial Theory, Series B*, 80:346–355, 2000.

[26] A. Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*, volume 24 of *Algorithms and Combinatorics.* Springer, 2003.

[27] P. Zhang, E. A. Schon, S. G. Fischer, E. Cayanis, J. Weiss, S. Kistler, and P. E. Bourne. An algorithm based on graph theory for the assembly of contigs in physical mapping of DNA. *CABIOS*, 10:309–317, 1994.