Information Processing Letters **98** (2006) 96-100

Vertex rankings of chordal graphs and weighted trees

Dariusz Dereniowski*, Adam Nadolski†
Department of Algorithms and System Modeling,
Gdańsk University of Technology, Poland
{deren,nadolski}@eti.pg.gda.pl

Abstract: In this paper we consider the vertex ranking problem of weighted trees. We show that this problem is strongly NP-hard. We also give a polynomial-time reduction from the problem of vertex ranking of weighted trees to the vertex ranking of (simple) chordal graphs, which proves that the latter problem is NP-hard. In this way we solve an open problem of Aspvall and Heggernes. We use this reduction and the algorithm of Bodlaender et al.'s for vertex ranking of partial k-trees to give an exact polynomial-time algorithm for vertex ranking of a tree with bounded and integer valued weight functions. This algorithm serves as a procedure in designing a PTAS for weighted vertex ranking problem of trees with bounded weight functions.

Keywords: chordal graph, computational complexity, graph algorithms, vertex ranking

1 Introduction

Let $G = (V, E)$ be a simple graph. A function $c\colon V(G) \to \{1, \ldots, k\}$ is a *vertex k-ranking* of G if every path connecting two vertices u, v satisfying $c(u) = c(v)$, contains a vertex x such that $c(x) > c(u)$. A vertex k-ranking is *optimal* if the integer k is as small as possible, and in that case k is equal to the *vertex ranking number* of G, denoted by $\chi_r(G)$. Define an *edge ranking* of G as a vertex ranking of the line graph $L(G)$ and the *edge ranking number*, denoted by $\chi'_r(G)$, is equal to $\chi_r(L(G))$.

In this paper we consider a generalization of the vertex ranking problem, where a graph $G = (V, E)$ with a weight function $w\colon V \to \mathbb{R}_+$ are given. If $A, B \subseteq \mathbb{R}$ then $A < B$ ($A > B$) iff for each $a \in A$ and $b \in B$, $a < b$ ($a > b$, resp.). Then, a function c assigning a finite union of intervals of $\mathbb{R}_+ \cup \{0\}$ to

*Partially supported by KBN grants 3 T11C 01127 and 4 T11C 04725
†Partially supported by KBN grant 4 T11C 04725

each vertex of G is a *weighted vertex ranking* of G if for each $v \in V$ we have $|c(v)| = w(v)$ and each path connecting two vertices u, v such that $c(u) \cap c(v) \neq \emptyset$ contains a vertex x such that $c(x) > c(u) \cap c(v)$. We say that c is *optimal* if $|c(V)|$ is as small as possible, where $c(X) = \bigcup_{v \in X} c(v)$ for $X \subseteq V$. Symbol $\chi_r(G, w)$ is used to denote the *weighted vertex ranking number* of G and w and is defined as $\chi_r(G, w) = |c(V)|$, where c is an optimal vertex ranking of G.

A *chord* is an edge connecting two nonconsecutive vertices of a cycle. A graph is *chordal* if every cycle of length at least four has a chord. The vertex ranking problem has application in the parallel Cholesky factorization of matrices [9] and chordal graphs are of special interest, because this is the class of graphs with perfect elimination orderings [8]. Aspvall and Heggernes [1] presented an exact and exponential algorithm for computing an optimal vertex ranking of a given chordal graph. Note that this problem is polynomially solvable for interval graphs [1, 5], a subclass of chordal graphs. An approximate algorithm for chordal graphs is given in [2]. Bodlaender et al. [4] gave an $O(\log^2 n)$-approximation algorithm for general graphs.

This paper is organized as follows. The next section gives the necessary definitions and some preliminary results. In Section 3 we show that the weighted vertex ranking problem is strongly NP-hard for trees. Then, we derive a polynomial-time reduction from this problem to the problem of vertex ranking of (simple) chordal graphs. This proves that the latter one is NP-complete, which answers an open question from [1]. This reduction is also used to derive an algorithm for finding optimal vertex ranking of a tree with a weight function which assigns bounded integers to vertices, which we describe in Section 4. Then, we use this algorithm to obtain a polynomial-time approximation scheme for vertex ranking of trees with a weight function w assigning the values of the same order to the vertices, i.e. $w(v) = \Theta(f(n))$, where f is an arbitrary function and n is the number of vertices of the colored tree. In the following (if the meaning is clear from context) we sometimes write vertex ranking and vertex ranking number instead of weighted vertex ranking and weighted vertex ranking number, respectively.

2 Definitions and preliminaries

In this paper we consider connected graphs, since finding a vertex ranking of a graph $G = G_1 \cup \cdots \cup G_j$ reduces to finding rankings of all connected components independently. If $S \subseteq V(G)$ then $G - S = (V \setminus S, E \setminus \{\{u, v\} : u \in S \text{ or } v \in S\})$. *A connected component* of $G - S$ is any maximal subgraph which is connected, i.e. there exists a path between every pair of vertices.

Definition 1 Let a, b be any two nonadjacent vertices of a graph G. *An a, b-separator* is a subset S of vertices of G such that a and b belong to different connected components of $G - S$. An a, b-separator S is *minimal* if no proper subset of S is an a, b-separator. We say that S is a *minimal separator* of G if there exist $a, b \in V(G)$ such that S is a minimal a, b-separator.

Let $\mathcal{S}_G$ denote the set of all minimal separators of graph G and if $S \subseteq V$, then $\mathcal{C}_G(S)$ denotes the set of all connected components of $G - S$.

Theorem 1 ([5]) *For any simple graph G, which is not complete*

$$\chi_r(G) = \min \left\{ |S| + \max\{\chi_r(C) : C \in \mathcal{C}_G(S)\} : S \in \mathcal{S}_G \right\}.$$

□

We will also use the following generalization of the above theorem.

Theorem 2 *Let G and $w\colon V(G) \to \mathbb{R}_+$ be a graph and a weight function, respectively. There exists an optimal vertex ranking of G, which assigns an interval of size $w(v)$ to every vertex $v \in V(G)$. If G is complete, then $\chi_r(G, w) = \sum_{v \in V(G)} w(v)$ and if G is not complete then*

$$\chi_r(G, w) = \min \left\{ \sum_{v \in S} |w(v)| + \max\{\chi_r(C, w|_{V(C)}) : C \in \mathcal{C}_G(S)\} : S \in \mathcal{S}_G \right\}.$$

Proof: Let c be an optimal weighted vertex ranking of G. We prove the first part of this theorem by induction on $n = n(G)$. The claim clearly holds for complete graphs. Let

$$A = \bigcup_{u,v \in V(G), u \neq v} c(u) \cap c(v).$$

Consider the set of vertices $S = \{z_1, \ldots, z_j\}$ such that $c(z_i) > A$ for each $i = 1, \ldots, j$. If G is not complete, then there exist two nonadjacent vertices u, v in G such that $c(u) \cap c(v) \neq \emptyset$. Since G is connected, there exists a path P between u and v. By the definition of vertex ranking, there exists $x \in V(P)$ such that $c(x) > c(u) \cap c(v)$, which means that $S \neq \emptyset$. Note that $|c(S)| = \sum_{z \in S} w(z)$, because $c(z_i) \cap c(z_{i'}) = \emptyset$, for all $i \neq i'$. Thus, we can modify c so that

$$c(z_i) = \Big(\chi_r(G, w) - |c(S)| + \sum_{d < i} w(z_d), \chi_r(G, w) - |c(S)| + \sum_{d \leq i} w(z_d) \Big),$$

for $i = 1, \ldots, j$. If $C \in \mathcal{C}_G(S)$ then, by the induction hypothesis, we can modify $c|_{V(C)}$ so that $c(u)$ is an interval for each $u \in V(C)$, which proves the first part of the theorem.
To prove the second part, observe that if S (defined above) is not a minimal separator of G then there exists a minimal separator $S' \subseteq S$. Therefore

$$\begin{aligned} \chi_r(G, w) &= \sum_{v \in S'} w(v) + \max\{\chi_r(C, w|_{V(C)}) : C \in \mathcal{C}_G(S')\} \\ &\geq \min_{S'' \in \mathcal{S}_G} \Big\{ \sum_{v \in S''} |w(v)| + \max\{\chi_r(C, w|_{V(C)}) : C \in \mathcal{C}_G(S'')\} \Big\}. \end{aligned}$$

Similarly we can prove the reverse inequality, which completes the proof. □

On the basis of Theorem 2, we assume in the following that for each $v \in V(G)$ $c(v) = (a, b)$ for some $a, b \in \mathbb{R}_+ \cup \{0\}$, $a < b$.

3 Hard instances

3.1 NP-hardness of vertex ranking of weighted trees

For a given tree $T = (V, E)$ with an edge-weight function $w\colon E \to \mathbb{N}$, let $W = \sum_{e \in E} w(e)$. Define a tree $\widetilde{T} = (\widetilde{V}, \widetilde{E})$ with a vertex-weight function $\widetilde{w}\colon \widetilde{V} \to \mathbb{N}$ as follows

$$\begin{aligned}
\widetilde{V} &= V \cup \{v[e] : e \in E\}, \\
\widetilde{E} &= \{\{u, v[e]\} : u \in V, e \in E \text{ and } u \in e\}, \\
\widetilde{w}(u) &= W, \text{ for each } u \in V, \\
\widetilde{w}(v[e]) &= w(e), \text{ for each } e \in E.
\end{aligned}$$

Lemma 1 *If c is an optimal vertex ranking of $\widetilde{T}$, then for each $u \in V$, $c(u) < \bigcup_{e \in E : u \in e} c(v[e])$.*

Proof: Assume for a contradiction that there exists a vertex $u \in V$ and an edge $e \in E$, $u \in e$ such that $c(u) > c(v[e])$ ($c(u), c(v[e])$ are intervals which cannot overlap). Consider a vertex x such that $e = \{u, x\}$. Note that $u, v[e], x$ form a path in $\widetilde{T}$ and $|c(u)| = |c(x)| = W$. If $c(x) < c(v[e])$ then c uses at least $2W + \widetilde{w}(v[e])$ colors. If $c(x) > c(v[e])$ then $c(u)$ and $c(x)$ cannot overlap (by the definition of vertex ranking) which means that in that case we also have that c uses at least $2W + \widetilde{w}(v[e])$ labels. In both cases we have a contradiction with the optimality of c, because $\chi_r(\widetilde{T}, \widetilde{w}) \leq 2W$. □

Corollary 1 *W.l.o.g. we may assume that if c is an optimal vertex ranking of $\widetilde{T}$ then $c(v) = (0, W)$ for each $v \in V$.* □

Lemma 2 $\chi'_r(T, w) = \chi_r(\widetilde{T}, \widetilde{w}) - W$.

Proof: Add edges to $\widetilde{T}$ so that for each $v \in V$ the neighbors of v in $\widetilde{T}$ form a clique and then remove vertices (with incident edges) in V from $\widetilde{T}$. Graph $\widetilde{T}'$ obtained in this way is the line graph of T which means that $\chi'_r(T, w) = \chi_r(\widetilde{T}', \widetilde{w}|_{V(\widetilde{T}')})$. By Corollary 1, $\chi_r(\widetilde{T}', \widetilde{w}|_{V(\widetilde{T}')}) = \chi_r(\widetilde{T}, \widetilde{w}) - W$. □

The vertex ranking problem of weighted trees is clearly in NP and the size of $\widetilde{T}$ is polynomial in $n(T)$. Lemma 2 and the fact that edge ranking of weighted trees with weight function assigning positive integers bounded by a polynomial in n to the edges is NP-hard [6] give the following

Theorem 3 *Vertex ranking of trees with positive integral weights on vertices is strongly* NP*-hard.* □

3.2 Simple chordal graphs

Assume that a tree $T = (\{v_1, \dots, v_n\}, E)$ and a weight function $w\colon V(T) \to \mathbb{N}$ are given. Create a simple graph $G = G(T, w)$, $V(G) = V_1 \cup \dots \cup V_n$, where $V_i = \{v_i^1, \dots, v_i^{w(v_i)}\}$, and $\{v_i^k, v_j^l\} \in E(G)$ if $i = j$ or $\{v_i, v_j\} \in E(T)$ for $i, j \in \{1, \dots, n\}$, $k \in \{1, \dots, w(v_i)\}$ and $l \in \{1, \dots, w(v_j)\}$. Fig. 1($a$) shows some weighted tree T and Fig. 1(b) presents the corresponding graph $G(T, w)$.

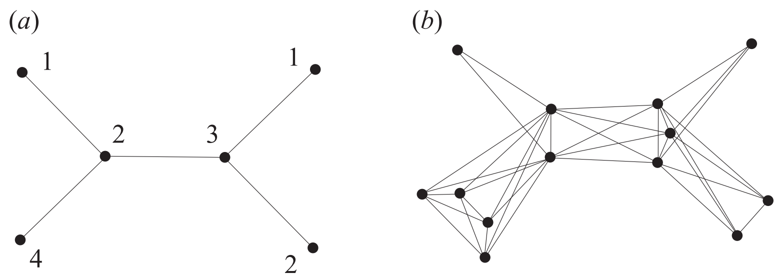

Figure 1: (a) a weighted tree T; (b) the corresponding graph $G(T, w)$;

Lemma 3 *If S is a minimal separator of G then $S = V_i$ for some $i \in \{1, \dots, n\}$.*

Proof: Assume that S separates vertices $a, b \in V(G)$. We have $a \in V_i$ and $b \in V_j$ for some $i \neq j$. Let $v_i, v_{p_1}, \dots, v_{p_l}, v_j$ be the path joining v_i and v_j in T. If for each $k = 1, \dots, l$ there exists a vertex in V_{p_k} which does not belong to S then there exists a path connecting a and b in $G - S$. So, $V_{p_k} \subseteq S$ for some $k \in \{1, \dots, l\}$. Note that the set V_{p_k} separates a and b in G. By the minimality of S, $S = V_{p_k}$. □

Lemma 4 $\chi_r(G) = \chi_r(T, w)$.

Proof: We prove this lemma by induction on the number of vertices of T. If $V(T) = \{v\}$ then $\chi_r(T, w) = w(v)$ and G is a clique with $w(v)$ vertices. Assume that the induction hypothesis holds for all trees with at most n vertices and let T have $n + 1$ nodes.

Consider an optimal vertex ranking c of G. By Theorem 1 and Lemma 3, some set V_i contains vertices which get unique and the biggest colors under c. If $G' \in \mathcal{C}_G(V_i)$ and $T' \in \mathcal{C}_T(\{v_i\})$ are the corresponding connected components of $G - V_i$ and $T - \{v_i\}$, respectively, then by the induction hypothesis $\chi_r(G') = \chi_r(T', w|_{V(T')})$. Thus, we can create a vertex ranking c' of T which uses at most $|V_i| + \max\{\chi_r(G') : G' \in \mathcal{C}_G(V_i)\}$ colors. This implies that $\chi_r(T, w) \leq \chi_r(G)$. Similarly (using Theorem 2) we can prove that $\chi_r(G) \leq \chi_r(T, w)$, which completes the proof. □

Theorem 4 *The problem of deciding whether $\chi_r(G) \leq k$ for a given $k \in \mathbb{N}$ and chordal graph G is* NP*-complete.*

Proof: The vertex ranking problem is in NP. A graph is chordal if and only if its every minimal separator is a complete graph [7]. This, and Lemma 3 imply that graph G defined above is chordal. Theorem 3 and Lemma 4 complete the proof. $\square$

4 PTAS for vertex ranking with bounded weight function

First, we will give a definition and some necessary facts concerning partial k-trees.

Definition 2 Let $k \in \mathbb{N}$. A complete graph on k vertices is a *k-tree*. If G is a k-tree with $n \geq k$ nodes then a graph obtained from G by adding new vertex v to G in such a way that neighbors of v form a complete graph on k vertices, is also a *k-tree*. A *partial k-tree* is a subgraph of some k-tree.

Theorem 5 ([3]) *Optimal vertex ranking of a partial k-tree can be computed in $n^{O(k)}$ time.* $\square$

Let a tree T with a weight function w be given. Assume that there exists a constant $b \in \mathbb{N}$ such that $w(v) \in \{1, \ldots, b\}$ for each $v \in V$. Consider a simple chordal graph $G(T, w)$ defined in Section 3.2, obtained on the basis of T and w. Clearly, G is a partial $(2b)$-tree, which gives the following.

Corollary 2 *If T is a tree and w is a weight function such that $w(v) \in \{1, \ldots, b\}$ for every vertex $v \in V$, then $\chi_r(T, w)$ can be computed in $n^{O(b)}$ time.* $\square$

We will use the above optimal algorithm to derive a PTAS for a more general case, where the weights of vertices are bounded real numbers, i.e. $w(v) \in [1, b]$. Assume that an error parameter $\epsilon > 0$ is given. Let l be an integer satisfying $1/2^l \leq \epsilon < 1/2^{l-1}$.

Consider the function $\lceil 2^l \cdot w \rceil$. Clearly $\lceil 2^l \cdot w \rceil = O(b/\epsilon)$. This means that the number of vertices of the corresponding chordal graph $G(T, \lceil 2^l \cdot w \rceil)$ is $O(bn(T)/\epsilon)$ and G is a partial $O(b/\epsilon)$-tree. Consider an algorithm $A_\epsilon(T, w)$ which computes an approximate vertex ranking c_T of T and returns the number of colors used. The algorithm $A_\epsilon(T, w)$ is as follows.

Step 1: compute the chordal graph $G = G(T, \lceil 2^l \cdot w \rceil)$;

Step 2: use the algorithm from Theorem 5 to find an optimal vertex ranking c_G of G;

Step 3: for each v_i^j, $j = 2, \ldots, \lceil 2^l \cdot w(v_i) \rceil$ add the interval $(\frac{c_G(v_i^j)-1}{2^l}, \frac{c_G(v_i^j)}{2^l})$ to $c_T(v_i)$ and finally add $(\frac{c_G(v_i^1)-(\lceil 2^l \cdot w(v_i)\rceil - 2^l \cdot w(v_i))}{2^l}, \frac{c_G(v_i^1)}{2^l})$ to $c_T(v_i)$;

Step 4: return $|c_T(V(T))|$;

To prove the correctness of c_T assume that there are two vertices v_i, v_j in T such that $c_T(v_i) \cap c_T(v_j) \neq \emptyset$, and there exists a path between v_i, v_j in T violating the definition of ranking. There exist $r \in \mathbb{R}_+$ and an integer s such that $\emptyset \neq (r, \frac{s}{2^l}) \subseteq c_T(v_i) \cap c_T(v_j)$. By the definition of c_T, $s \in c_G(V_i) \cap c_G(V_j)$ and there exists a path in G connecting cliques V_i, V_j, which contains no vertex with color greater than s – a contradiction. By Lemma 4 we have $\chi_r(G) = \chi_r(T, \lceil 2^l \cdot w \rceil)$, which implies

$$A_\epsilon(T, w) \leq \frac{\chi_r(T, \lceil 2^l \cdot w \rceil)}{2^l}. \tag{1}$$

Lemma 5 *If $w(v) \geq 1$ for each $v \in V(T)$ then $\chi_r(T, \lceil 2^l \cdot w \rceil) \leq (1 + 1/2^l) \cdot \chi_r(T, 2^l \cdot w)$.*

Proof: We prove this theorem by induction on n. Note that $\lceil 2^l \cdot w(v) \rceil - 2^l \cdot w(v) < 1$. By assumption, $1 \leq w(v)$, which proves the case when $n = 1$. Now, assume that $n > 1$. Suppose that c is an optimal vertex ranking of T with weight $2^l \cdot w$. By Theorem 2, $c(v)$ is an interval for each vertex $v \in V(T)$ and there exists a vertex $v \in V(T)$ such that $c(v) = (\chi_r(T, 2^l \cdot w) - 2^l \cdot w(v), \chi_r(T, 2^l \cdot w))$. As before, $\lceil 2^l \cdot w(v) \rceil < (1 + 1/2^l) 2^l \cdot w(v)$. By the induction hypothesis, $\chi_r(C, \lceil 2^l \cdot w|_{V(C)} \rceil) \leq (1 + 1/2^l) \chi_r(C, 2^l \cdot w|_{V(C)})$ for each $C \in \mathcal{C}_T(\{v\})$. By Theorem 2,

$$\begin{aligned} \chi_r(T, \lceil 2^l \cdot w \rceil) &\leq \lceil 2^l \cdot w(v) \rceil + \max\{\chi_r(C, \lceil 2^l \cdot w|_{V(C)} \rceil) : C \in \mathcal{C}_T(\{v\})\} \\ &\leq (1 + 1/2^l)\Big(2^l \cdot w(v) + \max\{\chi_r(C, 2^l \cdot w|_{V(C)}) : C \in \mathcal{C}_T(\{v\})\}\Big) \\ &= (1 + 1/2^l) \chi_r(T, 2^l \cdot w). \end{aligned}$$

□

If $p \in \mathbb{R}_+$ then by Theorem 2 for any weight function w it holds

$$\chi_r(T, p \cdot w) = p \cdot \chi_r(T, w). \tag{2}$$

Using Lemma 5 and equations (1) and (2) we obtain

$$\frac{A_\epsilon(T, w)}{\chi_r(T, w)} \leq \frac{\chi_r(T, \lceil 2^l \cdot w \rceil)}{2^l \cdot \chi_r(T, w)} = \frac{\chi_r(T, \lceil 2^l \cdot w \rceil)}{\chi_r(T, 2^l \cdot w)} \leq 1 + \epsilon.$$

By the discussion above, the running time of A_ϵ is polynomial in $n(T)$, which proves the following theorem.

Theorem 6 *Let $\epsilon > 0$ be fixed. If T is any tree and w is a weight for vertices of T, such that $w(v) \in [1, b]$ for some fixed b and each $v \in V(T)$, then there exists a $(bn(T)/\epsilon)^{O(b/\epsilon)}$–algorithm A_ϵ for weighted vertex ranking of T. Furthermore,*

$$\frac{A_\epsilon(T, w)}{\chi_r(T, w)} \leq 1 + \epsilon.$$

□

Note that if $w(v) = \Theta(f(n))$ for any function f, then we can also use the above algorithm. If $b_1 f(n) \leq w(v) \leq b_2 f(n)$ for some fixed b_1 and b_2, $0 < b_1 < b_2$, then the weight function $w'(v) := \frac{w(v)}{b_1 f(n)}$ satisfies the assumptions of Theorem 6 with $b = b_2/b_1$.

References

[1] B. Aspvall, P. Heggernes, Finding minimum height elimination trees for interval graphs in polynomial time, *BIT* **34** (1994) 484-509.

[2] C. Bornstein, B. Maggs, G. Miller, R. Ravi, Parallelizing elimination orders with linear fill, *FOCS* 1997 274-283.

[3] H. Bodlaender, J.S. Deogun, K. Jansen, T. Kloks, D. Kratsch, H. Müller, Z. Tuza, Rankings of graphs, *SIAM J. Discrete Math.* **11** (1998) 168-181.

[4] H. Bodlaender, R. J. Gilbert, H. Hafsteinson, T. Kloks, Approximating treewidth, pathwidth, frontsize, and shortest elimination tree, *J. Algorithms* **18** (1995) 238-255.

[5] J.S. Deogun, T. Kloks, D. Kratsch, H. Müller, On the vertex ranking problem for trapezoid, circular-arc and other graphs, *Discrete Appl. Math.* **98** (1999) 39-63.

[6] D. Dereniowski, Edge ranking of weighted trees, *Discrete Appl. Math.* (accepted).

[7] G.A. Dirac, On rigid circuit graphs, *Abh. Math. Sem. Univ. Hamburg* **25** (1961) 71-76.

[8] D.R. Fulkerson, O.A. Gross, Incidence matrices and interval graphs, *Pacific J. Math.* **15** (1965) 838-855.

[9] J.W.H. Liu, The role of elimination trees in sparse factorization, *SIAM J.Matrix Analysis and Appl.* **11** (1990) 134-172.