Available online at www.sciencedirect.com
ScienceDirect
Journal of Discrete Algorithms 4 (2006) 499–510
JOURNAL OF DISCRETE ALGORITHMS
www.elsevier.com/locate/jda

A 3-approximation for the pathwidth of Halin graphs ☆

Fedor V. Fomin [a,*], Dimitrios M. Thilikos [b]

[a] *Department of Informatics, University of Bergen N-5020, Bergen, Norway*
[b] *Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, E-08034, Barcelona, Spain*

Available online 5 July 2005

Abstract

We prove that the pathwidth of Halin graphs can be 3-approximated in linear time. Our approximation algorithms is based on a combinatorial result about respectful edge orderings of trees. Using this result we prove that the linear width of Halin graph is always at most three times the linear width of its skeleton.

Keywords: Halin graph; Pathwidth; Linear width

1. Introduction

All graphs in this paper are finite without loops or multiple edges. For a graph G we denote by $V(G)$ the vertex set of G, by $E(G)$ the edge set of G, and by $A(G)$ the set of degree one vertices of G. Also, given two graphs G_1 and G_2 we define their union as $G_1 \cup G_2 = (V(G_1) \cup V(G_1), E(G_1) \cup E(G_2))$. In this paper we call by *cycle* a graph that

☆ Fedor Fomin is supported by Norges forskningsråd project 160778/V30. Dimitrios Thilikos is supported by the EU within the 6th Framework Programme under contract 001907 (DELIS) and by the Spanish CICYT project TIC-2002-04498-C05-03 (TRACER).

* Corresponding author.
E-mail addresses: fomin@ii.uib.no (F.V. Fomin), sedthilk@lsi.upc.es (D.M. Thilikos).

1570-8667/$ – see front matter
doi:10.1016/j.jda.2005.06.004

is connected and has all vertices of degree 2. We set $n = |V(G)|$ and $m = |E(G)|$. Finally, for any vertex $v \in V(G)$ we will denote as E_v the set of edges that have v as an endpoint.

A *plane graph* is a particular drawing of a planar graph in the plane without crossings. A plane graph G is a *Halin graph* if its edge set can be partitioned into a tree without vertices of degree 2 (this tree is called the *skeleton*) and a cycle passing through all leaves of the tree. A graph is a *Halin graph* if it is isomorphic to some plane Halin graph. In other words, a graph is Halin if it is planar and is the union of a cycle C and a tree T without vertices of degree 2 and where $A(T) = V(C)$. Halin graphs were introduced by Halin in [16] and have been extensively examined (see e.g. [1,8,11,18,26]).

It is easy to check that every Halin graph has treewidth $\leqslant 3$ [4] (for the definitions, see Section 2.1). Finding a polynomial time algorithm for computing (or approximating) the pathwidth of graphs of treewidth bounded by some fixed constant is an old problem mentioned first by Dean in [12]. A general answer to this problem was given by Bodlaender and Kloks in [7]. However, while the algorithm in [7] is polynomial, its exponent is heavily depending on k and this makes it unpractical even for small values of k. Therefore, it is interesting to find a low-degree polynomial algorithm for graphs of treewidth bounded by small values of k. For $k = 1$ (i.e., when the input graph is forest) there was known an exact algorithm with running time $O(n \log n)$ time [14,21] and only recently Skodinis obtained a linear time algorithm [25]. Even addition of one edge to a tree requires much more elaborate algorithms. Recently such an $O(n \log n)$ algorithm computing pathwidth of unicyclic graphs (i.e., non-tree graphs obtained from a tree by adding one edge) is given by Ellis and Markov [13]. In spite of many attacks, no explicit algorithm is known for graphs of treewidth 2. The best progress in this direction is the 2-approximation algorithm of [6] for biconnected outerplanar graphs, a class of graphs with treewidth $\leqslant 2$ (see also [15]). So far, no efficient exact or approximation algorithm is known for any class of graphs with treewidth 3 or more.

In this paper we give a linear time algorithm that approximates the pathwidth of Halin graphs within a factor of 3. Our algorithm is very simple: To find a good solution one only needs to compute the pathwidth of a skeleton of Halin graph. The proof of the correctness of our algorithm is more technical. We prove that the pathwidth of G cannot be more that three times the pathwidth of T plus one. It appears that for our proof the related parameter of linear-width defined by Thomas in [29] is much more convenient than pathwidth. Our proof is based on a combinatorial result about new type of edge orderings of trees, so called respectful orderings.

This paper is organized as follows. In Section 2 we give necessary definitions and known facts on width parameters. In Section 3 we discuss some basic properties of Halin graphs. In Section 4 we introduce the main tool of our proofs, respectful orderings and in Section 5 we conclude with the approximation algorithm.

2. Width parameters

2.1. Treewidth—pathwidth

The notions of treewidth and pathwidth were introduced by Robertson and Seymour in [24] and [23] (see [5] and [22] for surveys).

A *tree decomposition* of a graph G is a pair (X, U) where U is a tree whose vertices we will call *nodes* and $X = (\{X_i \mid i \in V(U)\})$ is a collection of subsets of $V(G)$ such that

(1) $\bigcup_{i \in V(U)} X_i = V(G)$,
(2) for each edge $\{v, w\} \in E(G)$, there is an $i \in V(U)$ such that $v, w \in X_i$, and
(3) for each $v \in V(G)$ the set of nodes $\{i \mid v \in X_i\}$ forms a subtree of U.

The *width* of a tree decomposition $(\{X_i \mid i \in V(U)\}, U)$ equals $\max_{i \in V(U)}\{|X_i| - 1\}$. The *treewidth* of a graph G is the minimum width over all tree decompositions of G.

If in the definitions of a tree decomposition and treewidth we restrict U to be a tree with all vertices of degree at most 2 (i.e., a path) then we have the definitions of path decomposition and pathwidth. It is convenient to denote a path decomposition as a sequence $(X_1, X_2, \ldots, X_r)$ of subsets of $V(G)$.

2.2. Vertex orderings

For our purposes it is more convenient to work with vertex and edge orderings than with width parameters.

For $S \subseteq V(G)$ we define

$$\partial S = \bigl\{u \in S \text{ and there exists } w \in V(G) \setminus S \text{ such that } \{u, w\} \in E(G)\bigr\}.$$

Let $\sigma = (v_1, v_2, \ldots, v_n)$ be an ordering of $V(G)$. For $j \in \{1, \ldots, n\}$ we put

$$V_j = \bigcup_{i=1}^{j} \{v_i\}.$$

Setting

$$\mathbf{vs}(G, \sigma) = \max_{i \in \{1, \ldots, n\}} |\partial V_i|,$$

we define the *vertex separation* of G as

$$\mathbf{vs}(G) = \min\bigl\{\mathbf{vs}(G, \sigma)\colon\ \sigma \text{ is an ordering of } V(G)\bigr\}.$$

The linear width was introduced by Thomas [29] and is closely related to crusades of Bienstock and Seymour [3] (see also Bienstock's survey [2]). For $X \subseteq E(G)$ let $\delta(X)$ be the set of all vertices incident to edges in X and $E(G) \setminus X$. Let $\sigma = (e_1, e_2, \ldots, e_m)$ be an ordering of $E(G)$. For $i \in \{1, \ldots, m\}$ we put $E_i = \bigcup_{j=1}^{i} \{e_j\}$. We define

$$\mathbf{lw}(G, \sigma) = \max_{i \in \{1, \ldots, m\}} \bigl|\delta(E_i)\bigr|,$$

and the *linear width* of G as

$$\mathbf{lw}(G) = \min\{\mathbf{lw}(G, \sigma)\text{: } \sigma \text{ is an ordering of } E(G)\}.$$

The results of Bienstock and Seymour [3] imply that for graphs without vertices of degree 1, the linear width has a game theoretic interpretation in terms of mixed search number. (See also [27] on further discussions of mixed search number.)

The following proposition is well known. (See the survey of Möhring [22] for an overview of the related results.) It follows directly form the results of Kirousis and Papadimitriou [20] on interval width of a graph, see also [19].

Proposition 1. [19,20] *For any graph G,* $\mathbf{vs}(G) = \mathbf{pw}(G)$.

Lemma 2. *For any graph G,* $\mathbf{pw}(G) \leqslant \mathbf{lw}(G) \leqslant \mathbf{pw}(G) + 1$.

Proof. Let $\sigma = (v_1, v_2, \ldots, v_n)$ be an ordering of the vertices of G of vertex separation $\leqslant k$. For each $i \in \{1, \ldots, n\}$ let E_i be the subset of edges of G induced by the vertex set V_i. Then $\delta(E_i) = \partial(V_i)$. We define an ordering $\pi = (e_1, \ldots, e_m)$ of the edges of G such that for every $1 \leqslant k \leqslant l \leqslant n$, the conditions $e_i \in E_k$ and $e_j \in E_l$ imply $i \leqslant j$. It is easy to verify that $\mathbf{lw}(G, \pi) \leqslant k+1$. Thus $\mathbf{lw}(G) \leqslant \mathbf{vs}(G)+1$ and by Proposition 1, $\mathbf{lw}(G) \leqslant \mathbf{pw}(G)+1$.

Let now $\pi = (e_1, \ldots, e_m)$ be an edge ordering of G. W.l.o.g. we assume that G is connected and we define the vertex set sequence $\mathcal{X} = (X_1, \ldots, X_r)$ so that $X_i = \delta\{e_1, \ldots, e_{i-1}\} \cup e_i$, $i \in \{1, \ldots, r\}$. It is easy to see that $\mathcal{X}$ is a path decomposition of G where the maximum size of a X_i is at most one more than the linear-width of L. Therefore $\mathbf{pw}(G) \leqslant \mathbf{lw}(G)$. □

Let T be a tree and let P be a path in T. We define $\mathcal{T}(T, P)$ as the set of trees defined by the connected components of the graph taken after subdividing in T all edges not in P but with endpoints in P and then removing all the vertices in P. For reasons of simplicity, we will use the same notation as in T for all the vertices of the trees in $\mathcal{T}(T, P)$ (the only essential difference is that the vertices of P may be present in several trees in $\mathcal{T}(T, P)$—it is convenient to imagine each of the trees in $\mathcal{T}(T, P)$ as the closure of each of the connected

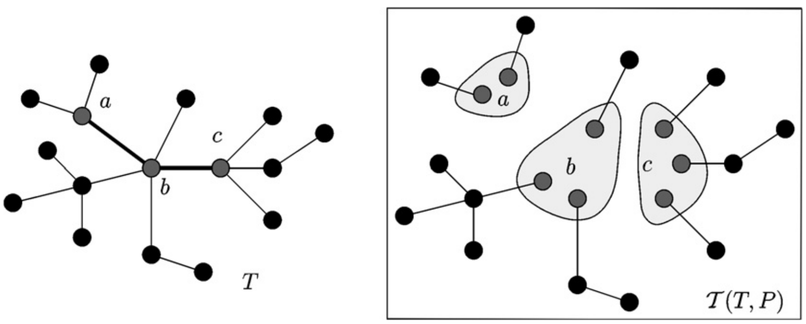

Fig. 1. An example of the definition of the graph collection $\mathcal{T}(T, P)$ for some tree T and some path $P = (a, b, c)$ in it.

components of the set $(E(T) - E(P)) \cup (V(T) - V(P)))$. For an example of the definition of $\mathcal{T}(T, P)$ see Fig. 1.

The following theorem is the analogous for linear-width of the corresponding results of [14] on pathwidth. It can be directly derived by the results of [27] on the mixed search number and its connection with linear-width as explained in Theorem 24 of [28], so we omit its proof.

Theorem 3. *The following two statements hold*:

(1) *For any tree T and integer $k \geqslant 1$, $\mathbf{lw}(G) \leqslant k$ if and only if for any vertex $v \in V(T)$ the set $\mathcal{T}(T, \{v\})$ contains at most two trees of linear width $\leqslant k$ while all the other trees of this set are of linear width $\leqslant k-1$.*
(2) *Any tree T with $\mathbf{lw}(G) \leqslant k$ contains a path P where for any $v \in V(P)$, all the trees in $\mathcal{T}(T, \{v\})$ have linear width at most $k-1$.*

For a tree T of linear-width at most k, we call a path P given by the second statement of Theorem 3 the *spine* of T.

3. Halin graphs

It is a part of folklore that Halin graphs can be recognized in linear time (see [10, p. 118]). In this section we give a linear time Halin graph recognition algorithm which also outputs one of the skeletons, if the graph is Halin. This algorithm also seems to be a folklore and we sketch its main steps for completeness.

Given a face F of a planar embedding of a graph, we define $V(F)$ as the set of vertices and $E(F)$ as the set of edges incident to F. A face F of a plane graph G is *wrapping face* if $2 \cdot |V(F)| \geqslant |V(G)| + 2$ and every vertex of F is of degree 3 in G. Finally, if T is a tree we define $I(T) = V(T) - A(T)$, i.e., $I(T)$ and $A(T)$ are the sets of internal and leaf vertices of T respectively.

Lemma 4. *Every planar drawing of a Halin graph G has at least one and at most four wrapping faces. Moreover, there is at least one wrapping face such that $E(G) - E(F)$ is a skeleton of G.*

Proof. Let G be a plane Halin graph with a skeleton T. The edges of the cycle $E(G) - E(T)$ are incident to a face F and $|V(F)|$ is the number of leaves of T. There is no vertices of degree two in T, thus the number $|A(T)|$ of leaves in T is at least $|I(T)| + 2$ and $|V(F)| \geqslant |I(T)| + 2$. Notice also that $|V(G)| = |I(T)| + |A(T)| \leqslant |V(F)| - 2 + |V(F)|$. Thus F is a wrapping face and $E(G) - E(F)$ is a skeleton of G.

Let us prove that there are at most four wrapping faces in any Halin graph. We claim first that every two wrapping faces F_1, F_2 of G have at least one edge in common. In fact, $|V(G)| = |V(F_1)| + |V(F_2)| - |V(F_1) \cap V(F_2)| \geqslant |V(G)| + 2 - |V(F_1) \cap V(F_2)|$, therefore $|V(F_1) \cap V(F_2)| \geqslant 2$. Since all vertices of $V(F_1)$ and $V(F_2)$ have degree 3 in G, we have that each vertex in $V(F_1) \cap V(F_2)$ is incident to an edge that is incident to both

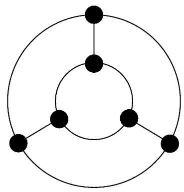

Fig. 2. A Halin graph with 3 wrapping faces and an isomorphic Halin plane graph.

F_1 and F_2 and the claim holds. The claim implies that G has at most four wrapping edges because otherwise the dual of G contains complete graph on five vertices K_5 as a subgraph and thus is not planar. □

In fact the largest possible number of wrapping faces a drawing of a Halin graph may have is 3 (see the graph in Fig. 2). We do not prove the tight version of the above lemma because the current bound is sufficient to support the linearity of the following algorithm.

Lemma 5. *For a given graph G on n vertices there exists an algorithm that in* $\mathrm{O}(n)$ *steps checks if G is a Halin graph and, in cases of a positive answer, returns its skeleton.*

Proof. The algorithm returning the correct answer to the question "is G a Halin graph?" is the following:

> Consider any planar drawing of G (such a drawing can be constructed in linear time [9,17]). For each face, check whether it is a wrapping face or not. If G does not have any wrapping face then stop and answer "no". Otherwise, for any wrapping face, check whether the removal of its edges creates a tree T. If such a wrapping face exists, then answer "yes and the skeleton of G is T", otherwise answer "no".

To see that the algorithm is correct observe that if G is a Halin graph, then from Lemma 4 it contains a wrapping face whose removal creates the skeleton of G. As the algorithm checks this for all possible wrapping faces it finally gives a correct (positive) answer. On the other side if the algorithm answers positively then this means that there exists a face in G such that the removal of its edges created a tree which means that the graph is a Halin graph. Notice now that to check whether a face F is a wrapping face needs $\mathrm{O}(|E(F)|)$ steps. Therefore, to detect the wrapping faces requires in total $\sum_{F \text{ is a face of } G} \mathrm{O}(|E(F)|) = \mathrm{O}(|V(G)|)$ steps. According to Lemma 4, the second check of the algorithm is applied only a constant number of times. Clearly, computing $T = E(G) - E(F)$ can be done in $\mathrm{O}(n)$ steps. It also needs the same time to check whether T is a tree. Therefore the algorithm can be implemented in $\mathrm{O}(n)$ steps. □

4. Respectful orderings

Let G be a plane graph. Let M_G be a plane graph with vertex set $E(G)$. We say that the graph M_G is a *medial graph* of G if $M_G = \bigcup_{v \in V(G)} C_v$ where,

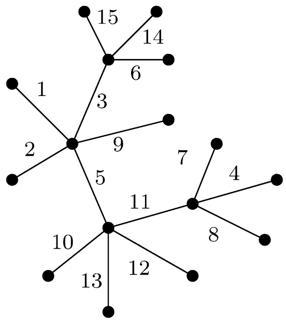

Fig. 3. A tree T, its medial graph and a respectful ordering of $E(T)$.

- C_v $(v \in V(G))$ are mutually edge-disjoint cycles.
- For each $v \in V(G)$, if $\{v, x_1\}, \{v, x_2\}, \ldots, \{v, x_t\}$ are the edges of E_v enumerated according to the cyclic order in the drawing of G, then C_v has vertex set $\{v, x_1\}, \{v, x_2\}, \ldots, \{v, x_t\}$ and, in C_v, vertex $\{v, x_{i-1}\}$ is adjacent to $\{v, x_i\}$ $(1 \leqslant i \leqslant t)$, where $x_0 = x_t$ (note that for every vertex v of G, $V(C_v) = E_v$).

For an example of a tree and its medial graph, see Fig. 3.

Let E be some subset of $E(G)$ and let $\sigma = (e_1, \ldots, e_m)$ be an ordering of E. We set $E_i = \bigcup_{j=1}^{i} e_j$, $1 \leqslant i \leqslant m$. We say that the ordering σ is *v-respectful* if for every $i \in \{1, \ldots, m\}$, the set of edges $E_i \cap V(C_v)$ form a connected subgraph (i.e., path or cycle) of M_G. An edge ordering σ of E is *respectful* if it is v-respectful for every vertex v of G. See Fig. 3 for an example of respectful ordering of $E(T)$.

In the rest of this paper, we will use the symbol $\oplus$ to denote the concatenation of two edge sequences, i.e., if $\sigma_i = (e_1^i, \ldots, e_{r_i}^i)$, $i = 1, 2$, then $\sigma_1 \oplus \sigma_2 = (e_1^1, \ldots, e_{r_1}^2, e_1^2, \ldots, e_{r_2}^2)$.

Lemma 6. *For any plane tree T with linear-width at most k, there exists a respectful edge ordering L of $E(T)$ that has linear-width at most k.*

Proof. We apply induction on k. The case $k = 0$, i.e., when the tree has at most one edge, is trivial. Let $k \geqslant 1$. We assume that the result holds for any $i < k$ and we prove that it also holds for k. Let $P = (v_1, \ldots, v_r)$ be the spine path of T. All the trees from $\mathcal{T}(T, P)$ are of linear-width $\leqslant k - 1$ and have respectful orderings. The proof idea is to concatenate these orderings in such a way, that for any vertex v of P the new ordering will be v-respectful (for the vertices out of P this is a direct consequence of the induction hypothesis).

For every $i \in \{1, \ldots, r\}$, we denote by $\sigma_i = (e_1^i, \ldots, e_{\rho_i}^i)$ an ordering of $E_{v_i} = V(C_{v_i})$ which is v_i-respectful and such that $e_1^i = \{v_{i-1}, v_i\}$ for $i \in \{2, \ldots, r\}$, and $e_{\rho_i}^i = \{v_i, v_{i+1}\}$ for $i \in \{1, \ldots, r-1\}$. For $i \in \{1, \ldots, r\}$ and $j \in \{1, \ldots, \rho_i\}$, we denote by T_j^i the connected component of $\mathcal{T}(T, v_i)$ that contains e_j^i as an edge.

Let $I = \{(i, j) \mid 1 \leqslant i \leqslant r, 1 \leqslant j \leqslant \rho_i$ and e_j^i is not an edge of $P\}$ and notice that for each $(i, j) \in I$, e_j^i belongs to a different tree of $T_j^i \in \mathcal{T}(T, P)$. By Theorem 3, any such a T_j^i has linear-width $\leqslant k - 1$ and by the induction hypothesis it has a respectful edge ordering σ^i_j of linear-width $\leqslant k-1$. We now define

$$\sigma = \sigma^1_1 \oplus \cdots \oplus \sigma^1_{\rho_1 - 1} \oplus \big(\{v_1, v_2\}\big) \oplus \sigma^2_2 \oplus \cdots \oplus \sigma^2_{\rho_2 - 1} \oplus \big(\{v_2, v_3\}\big) \oplus \cdots \\ \oplus \big(\{v_{r-1}, v_r\}\big) \oplus \sigma^r_2 \oplus \cdots \oplus \sigma^r_{\rho_r}$$

and we observe that σ is a respectful edge ordering of G of linear-width $\leqslant k$ (notice that the σ^i_j's that are omitted in this concatenation are exactly those that correspond to pairs (i, j) missing from I). $\square$

We call a tree T *ternary* if all its internal vertices have degree 3. A graph H obtained by a sequence of edge-contractions and edge-removals is said to be a *minor* of G. Clearly, for any minor H of a graph G, $\mathbf{lw}(H) \leqslant \mathbf{lw}(H)$.

Lemma 7. *Any plane tree T without vertices of degree* 2 *is the minor of a plane ternary tree T' where* $\mathbf{lw}(T) = \mathbf{lw}(T')$ *and* $A(T) = A(T')$.

Proof. Let σ be an edge ordering of T of minimum linear-width. By Lemma 6, there exists a respectful ordering σ of $E(T)$ with the same linear-width. We apply the following algorithm on T.

1. If T contains a vertex v of degree $\geqslant 4$ then goto the next step, otherwise output T and stop.

2. Let e_i and e_j be the two first edges in C_v that appear in σ (in terms of the medial graph of T, e_i and e_j are consecutive vertices in the cyclic ordering of C_v). We construct a new tree T' as follows: first construct the tree U_1 by taking by the union of the two trees of $\mathcal{T}(T, v)$ containing the edges e_i and e_j, then construct the tree U_2 by taking the union of the rest of the trees of $\mathcal{T}(T, v)$, then, for $i = 1, 2$, rename to v_i the vertex v in U_i and finally define T' as the disjoint union of U_1 and U_2 with $e_{\text{new}} = \{v^1, v^2\}$ as an additional edge, i.e., $T' = U_1 \cup U_2 \cup (\{v_1, v_2\}, \{v_1, v_2\}\})$ (see Fig. 4).

Notice that $\sigma' = (e_1, \ldots, e_i, \ldots, e_j, e_{\text{new}}, e_{j+1}, \ldots, e_q)$ is a respectful edge ordering for T' with linear-width $\leqslant k$. Therefore $\mathbf{lw}(T') \leqslant \mathbf{lw}(T)$. As T is a minor of T', we have that $\mathbf{lw}(T) \leqslant \mathbf{lw}(T')$ and thus T' has the same linear-width as T. Moreover, as the vertex splitting operation of this step is not applied to a leave, we get hat $A(T) = A(T')$.

3. Set $T := T', \sigma := \sigma'$ and goto step 1.

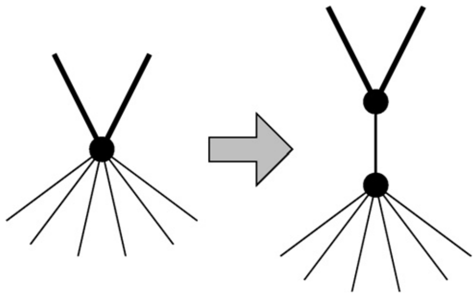

Fig. 4. Step 2 of the algorithm in Lemma 7.

The above algorithm stops only when T is a ternary tree as required. To see this, notice that after each step the quantity $\sum_{v\in I(T)}(\mathbf{deg}_T(v)-3)$ becomes smaller and the algorithms stops when its value is equal to 0 (we use the notation $\mathbf{deg}_T(v)$ to denote the degree of vertex v in T). $\square$

5. Approximation algorithm

Lemma 8. *Let H be a Halin graph with skeleton T. Then* $\mathbf{lw}(T) \leqslant \mathbf{lw}(H) \leqslant 3 \cdot \mathbf{lw}(T)$.

Proof. The first inequality is obvious as T is a subgraph of H. In what remains we will prove that $\mathbf{lw}(H) \leqslant 3 \cdot \mathbf{lw}(T)$. By Lemma 7, there is a ternary tree T' such that $\mathbf{lw}(T') = \mathbf{lw}(T)$ and $A(T) = A(T')$. Let H' be a plane Halin graph having T' as a skeleton and with exterior wrapping face. It easy to check that $\mathbf{lw}(H) \leqslant \mathbf{lw}(H')$ (this follows from the fact that H is a minor of H'). So to prove the lemma it is sufficient to show that $\mathbf{lw}(H') \leqslant 3 \cdot \mathbf{lw}(T')$.

We construct a new graph J by modifying the graph H'. For every internal face F of H' we do the following. Let $(v_1, \dots, v_r, v_1)$ be the cycle bordering F, where $r \geqslant 3$ and $\{v_1, v_r\}$ is the edge of $E(H') - E(T')$. We replace the edge $\{v_1, v_r\}$ by a path $(v_1 = a_1, a_2, \dots, a_r = v_r)$ of length $r-1$. (See Fig. 5.)

Notice that there is a natural one-to-one correspondence between edges of paths $(v_1, \dots, v_r)$ and $(a_1, a_2, \dots, a_r)$. For every $i \in \{1, \dots, r-1\}$, we call the edge $\{a_i, a_{i+1}\}$ the *shadow* of the edge $\{v_i, v_{i+1}\}$ and for every $i \in \{2, \dots, r-1\}$ we cal the vertex a_i *shadow* of v_i. As every edge $e \in E(T')$ is adjacent to two faces, it has two shadows in J. Moreover, as every vertex of $I(T')$ is of degree 3, it has 3 shadows. For $e \in E(T')$ and $v \in I(T')$, let $S(e)$ and $S(v)$ be the sets of shadows of e and v.

Notice that J is a Halin graph with skeleton T'. Clearly, $\mathbf{lw}(H') \leqslant \mathbf{lw}(J)$ and to finish the proof it is enough to prove that $\mathbf{lw}(J) \leqslant 3 \cdot \mathbf{lw}(T')$.

Let $E \subseteq E(T')$ and let $E^* = \bigcup_{e\in E}(\{e\} \cup S(e))$. Let also $v \in \delta_{T'}(E)$. Notice that $v \in I(T')$. As the internal vertices of T' have all degree 3, this means that either two or one of the three edges with v as endpoint are members of E. In each case, $v \in \delta_J(E^*)$ and moreover exactly 2 of the shadows of v will be members of $\delta_J(E^*)$ (see Fig. 6). That way we can correspond to each vertex of $\delta_{T'}$ exactly three vertices of $\delta_J(E^*)$. Certainly, to each

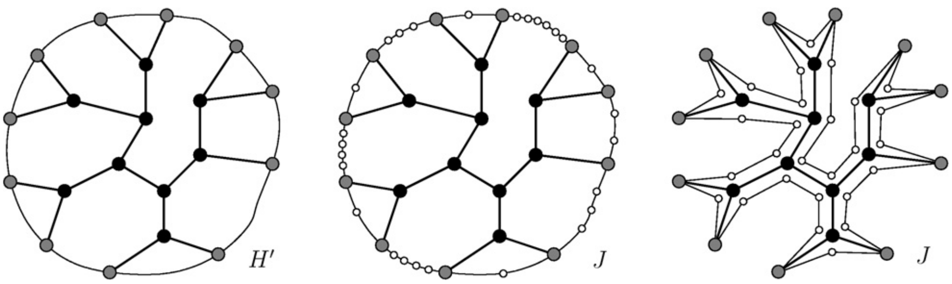

Fig. 5. The transformation of the proof of Lemma 8.

such triple in $\delta_J(E^*)$ corresponds exactly one vertex in $\delta_{T'}(E)$. Therefore,

$$\text{for any } E \subseteq E(T'), \quad \left|\delta_J\left(\bigcup_{e\in E}\big(\{e\}\cup S(e)\big)\right)\right| = 3\left|\delta_{T'}(E)\right|. \tag{1}$$

Let now $\sigma = (e_1, \ldots, e_q)$ be an edge ordering of T' of linear-width $\leqslant k$.
By (1), for any $\ell \in \{1, \ldots, q\}$,

$$\left|\delta_J\left(\bigcup_{i=1,\ldots,\ell}\big(\{e\}\cup S(e)\big)\right)\right| = 3\left|\delta_{T'}\big(\{e_1, \ldots, e_i\}\big)\right|.$$

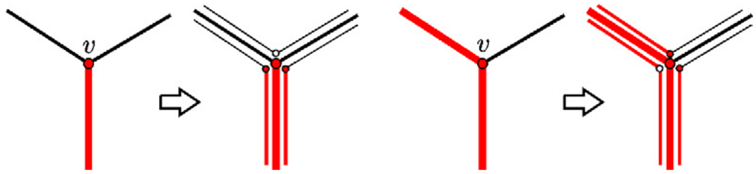

Fig. 6. Two ways to correspond a vertex v of T' to 3 vertices of J (filled vertices).

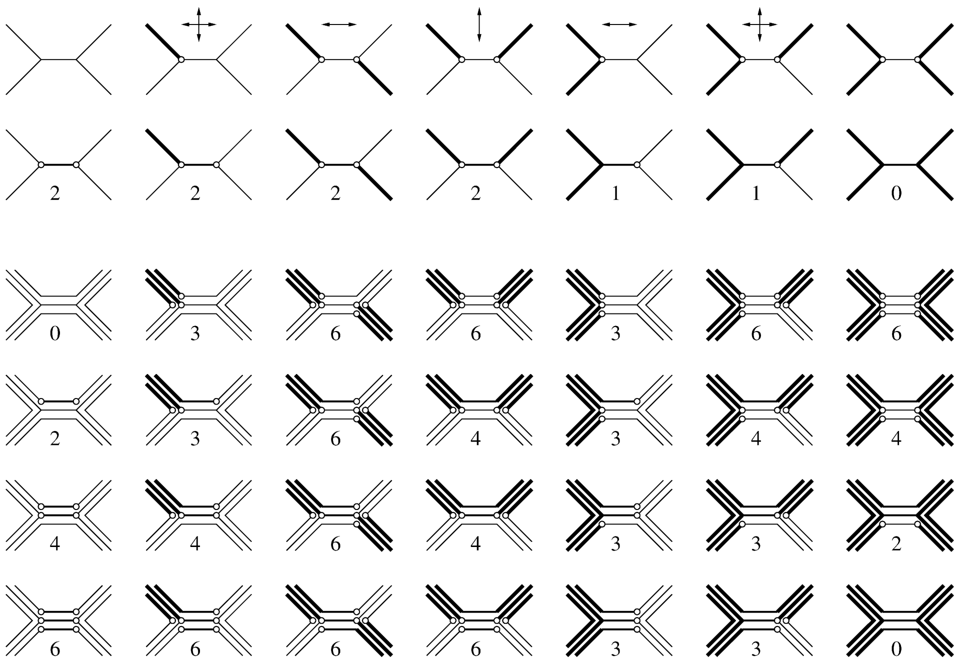

Fig. 7. There are 16 possible ways shadow edges can be distributed around an edge e with both endpoints of degree 3. We present 7 representative configurations (the rest come up as symmetric ones—arrows depict the symmetries) and the way the sub-ordering e^1, e, e^2 is contributing to the linear-width of L^*. Notice that, in each case, the contribution of the endpoints of e^1, e, e^2 in L^* is never more than 3 times the contribution of the endpoints of e in L.

To finish the proof we need to show how to transform σ to an ordering σ^* of J by suitable inserting into σ all the shadow edges. Each edge e of T' has two shadows in $E(J) - E(T')$. We call these two shadows e^1 and e^2 always choosing indices 1 or 2 arbitrarily except the following case: For $i = 1, 2$, each endpoint of $e = \{v, v'\}$ belongs to exactly one edge e_i appearing before e in σ and each such edge has a shadow sharing an endpoint with the same shadow e^* of e. In this case we set $e^1 = e^*$ and let e^2 be the other shadow of e (this special case, corresponds to the 4th row of the configurations in Fig. 7).

A detailed case analysis is depicted in Fig. 7 shows that $\sigma^* = (e_1^1, e_1, e_1^2, \ldots, e_q^1, e_q, e_q^2)$ is an edge layout of J with linear-width $\leqslant 3k$.

Thus

$$\mathbf{lw}(T) \leqslant \mathbf{lw}(H) \leqslant \mathbf{lw}(H') \leqslant \mathbf{lw}(J) \leqslant 3 \cdot \mathbf{lw}(T') = 3 \cdot \mathbf{lw}(T). \qquad \square$$

Taking in mind that the pathwidth of a tree can be computed in linear time [25] we may resume the results of this section to the following.

Theorem 9. *There exists an linear time algorithm that for any Halin graph H returns an integer k such that $k - 1 \leqslant \mathbf{pw}(H) \leqslant \mathbf{lw}(H) \leqslant 3 \cdot k$.*

Proof. Use the algorithm of Lemma 5 and find a skeleton T of H in $\mathrm{O}(n)$ time. Then, use the algorithm of [25] to compute the pathwidth of T in $\mathrm{O}(n)$ time and output $k = \mathbf{pw}(T) + 1$. From Lemma 8, $\mathbf{lw}(H) \leqslant 3 \cdot \mathbf{lw}(T)$. From Lemma 2, $\mathbf{lw}(T) \leqslant k$ therefore $\mathbf{lw}(H) \leqslant 3 \cdot k$. From Lemma 2, $\mathbf{pw}(H) \leqslant \mathbf{lw}(H)$ and as T is a subgraph of H we also have $k - 1 = \mathbf{pw}(T) \leqslant \mathbf{pw}(H)$. $\square$

References

[1] C.A. Barefoot, 4-connected Halin graphs are 2-Hamiltonian connected, Ars Combinatoria 35 (1993) 109–128.

[2] D. Bienstock, Graph searching, path-width, tree-width and related problems (a survey), in: Reliability of Computer and Communication Networks, in: DIMACS Ser. Discrete Math. Theoret. Comput. Sci., vol. 5, Amer. Math. Soc., Providence, RI, 1991, pp. 33–49.

[3] D. Bienstock, P. Seymour, Monotonicity in graph searching, J. Algorithms 12 (1991) 239–245.

[4] H.L. Bodlaender, Planar graphs with bounded treewidth, Tech. Rep. RUU-CS-88-14, Dept. of Computer Science, Utrecht University, Utrecht, the Netherlands, 1988.

[5] H.L. Bodlaender, A partial k-arboretum of graphs with bounded treewidth, Theoret. Comput. Sci. 209 (1998) 1–45.

[6] H.L. Bodlaender, F.V. Fomin, Approximation of pathwidth of outerplanar graphs, J. Algorithms 43 (2002) 190–200.

[7] H.L. Bodlaender, T. Kloks, Efficient and constructive algorithms for the pathwidth and treewidth of graphs, J. Algorithms 21 (1996) 358–402.

[8] J.A. Bondy, L. Lovász, Lengths of cycles in Halin graphs, J. Graph Theory 9 (1985) 397–410.

[9] K.S. Booth, G.S. Lueker, Testing the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms, J. Comput. System Sci. 13 (1976) 335–379.

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

[11] G. Cornuejols, D. Naddef, W.R. Pulleyblank, Halin graphs and the travelling salesman problem, Math. Programming 26 (1983) 287–294.

[12] N. Dean, Open problems, in: N. Robertson, P.D. Seymour (Eds.), Graph Structure Theory, Contemporary Mathematics, vol. 147, Amer. Math. Soc., Providence, RI, 1993, pp. 677–688.
[13] J.A. Elis, M. Markov, Computing the vertex separation of unicyclic graphs, Inform. and Comput. (2004), submitted for publication.
[14] J.A. Ellis, I.H. Sudborough, J.S. Turner, The vertex separation and search number of a graph, Inform. and Comput. 113 (1994) 50–79.
[15] R. Govindan, M.A. Langston, X. Yan, Approximating the pathwidth of outerplanar graphs, Inform. Process. Lett. 68 (1998) 17–23.
[16] R. Halin, Über simpliziale Zerfällungen beliebiger (endlicher oder unendlicher) Graphen, Math. Ann. 156 (1964) 216–225.
[17] J. Hopcroft, R. Tarjan, Efficient planarity testing, J. ACM 21 (1974) 549–568.
[18] S.B. Horton, R.G. Parker, On some results pertaining to Halin graphs, in: Proceedings of the Twenty-third Southeastern International Conference on Combinatorics, Graph Theory, and Computing, vol. 89, 1992, pp. 65–87.
[19] N.G. Kinnersley, The vertex separation number of a graph equals its path-width, Inform. Process. Lett. 42 (1992) 345–350.
[20] L.M. Kirousis, C.H. Papadimitriou, Interval graphs and searching, Discrete Math. 55 (1985) 181–184.
[21] N. Megiddo, S.L. Hakimi, M.R. Garey, D.S. Johnson, C.H. Papadimitriou, The complexity of searching a graph, J. ACM 35 (1988) 18–44.
[22] R.H. Möhring, Graph problems related to gate matrix layout and PLA folding, in: Computational Graph Theory, Comput. Suppl., vol. 7, Springer, Vienna, 1990, pp. 17–51.
[23] N. Robertson, P.D. Seymour, Graph minors. I. Excluding a forest, J. Combin. Theory Ser. B 35 (1983) 39–61.
[24] N. Robertson, P.D. Seymour, Graph minors II. Algorithmic aspects of treewidth, J. Algorithms 7 (1986) 309–322.
[25] K. Skodinis, Construction of linear tree-layouts which are optimal with respect to vertex separation in linear time, J. Algorithms 47 (2003) 40–59.
[26] P.F. Stadler, Minimum cycle bases of Halin graphs, J. Graph Theory 43 (2003) 150–155.
[27] A. Takahashi, S. Ueno, Y. Kajitani, Minimal forbidden minors for the family of graphs with proper-path-width at most two, IEICE Trans. Fundamentals E78-A (1995) 1828–1839.
[28] D.M. Thilikos, Algorithms and obstructions for linear-width and related search parameters, Discrete Appl. Math. 105 (2000) 239–271.
[29] R. Thomas, Tree-decompositions of graphs, Lecture Notes, School of Mathematics, Georgia Institute of Technology, Atlanta, Georgia 30332, USA, 1996.