Available online at www.sciencedirect.com
ScienceDirect
Theoretical Computer Science 383 (2007) 34–44
Theoretical Computer Science
www.elsevier.com/locate/tcs

Spanners for bounded tree-length graphs[☆]

Yon Dourisboure[a,*], Feodor F. Dragan[b], Cyril Gavoille[a], Chenyu Yan[b]

[a] LaBRI, Université Bordeaux 1, France
[b] Department of Computer Science, Kent State University, United States

Abstract

This paper concerns construction of additive stretched spanners with few edges for n-vertex graphs having a tree-decomposition into bags of diameter at most δ, i.e., the tree-length δ graphs. For such graphs we construct additive 2δ-spanners with $O(\delta n + n \log n)$ edges, and additive 4δ-spanners with $O(\delta n)$ edges. This provides new upper bounds for chordal graphs for which $\delta = 1$. We also show a lower bound, and prove that there are graphs of tree-length δ for which every multiplicative δ-spanner (and thus every additive $(\delta - 1)$-spanner) requires $\Omega(n^{1+1/\Theta(\delta)})$ edges.

Keywords: Additive spanner; Tree-decomposition; Tree-length; Chordality

1. Introduction

Let G be an unweighted connected graph with n vertices. A subgraph H of G is an (s, r)-*spanner* if $d_H(u, v) \leqslant s \cdot d_G(u, v) + r$ for all pair of vertices u, v of G. An $(s, 0)$-spanner is also termed *multiplicative* s-*spanner*, and an $(1, r)$-spanner is termed *additive* r-*spanner*. An (s, r)-spanner is also an $(s + r, 0)$-spanner (in particular, an additive r-spanner is a multiplicative $(r + 1)$-spanner), but the reverse is false in general.

The main objective is to construct for a graph an (s, r)-spanner with few edges. There are many applications of spanners, for example, the complexity of a lot of distributed algorithms depends on the number of messages, itself depending on the number of edges [3,4]. Sparse spanners occur also in the efficiency of compact routing schemes [5]. Unfortunately, given an arbitrary graph G and three integers s, r and m, determine whether G admits an (s, r)-spanner with m or fewer edges, is NP-complete [6], even if we restrict $r = 0$ (see also [7–11] for complexity issue). Best known results on (s, r)-spanners for general graphs are summarized in the following table.

☆ Preliminary results of this paper have been presented at the SIROCCO '04 and SWAT '04 conferences [Y. Dourisboure, C. Gavoille, Sparse additive spanners for bounded tree-length graphs, in: 11th International Colloquium on Structural Information & Communication Complexity (SIROCCO), in: Lecture Notes in Computer Science, vol. 3104, Springer, 2004, pp. 123–137; F.F. Dragan, C. Yan, I. Lomonosov, Collection tree spanners of graphs, in: T. Hagerup, J. Katajainen (Eds.), 9th Scandinavian Workshop on Algorithm Theory, SWAT, in: Lecture Notes in Computer Science, vol. 3111, Springer, 2004, pp. 64–76].

* Corresponding author.
E-mail addresses: yon.dourisboure@iit.cnr.it, dourisbo@labri.fr (Y. Dourisboure), dragan@cs.kent.edu (F.F. Dragan), gavoille@labri.fr (C. Gavoille), cyan@cs.kent.edu (C. Yan).

0304-3975/$ - see front matter
doi:10.1016/j.tcs.2007.03.058

(s, r)-spanner	Edges	Reference
$(1, 2)$	$\Theta(n^{3/2})$	[12–14][1]
$(1, 6)$	$O(n^{4/3})$	[15]
$(1, n^{(1-1/k)/2})$	$O(n^{1+1/k})$	[15], $k \geqslant 1$
$(2k-1, 0)$	$O(n^{1+1/k})$	[16–18],[2] $k \geqslant 1$
$(k, k-1)$	$O(n^{1+1/k})$	[15], $k \geqslant 1$
$(k-1, 2k-4)$	$O(kn^{1+1/k})$	[14], $k \geqslant 4$ even
$(k-2+\epsilon, 2k-2-\epsilon)$	$O(\epsilon^{-1}kn^{1+1/k})$	[14], $k \geqslant 3$ odd, $\epsilon > 0$
$(k-1+\epsilon, 2k-4-\epsilon)$	$O(\epsilon^{-1}kn^{1+1/k})$	[14], $k \geqslant 4$ even, $\epsilon > 0$
$(1+\epsilon, \beta(\epsilon, k))$	$O(\beta(\epsilon, k)n^{1+1/k})$	[14],[3] $k \geqslant 2$

An interesting question still left open is to know whether every graph has an additive $(2k-2)$-spanner with $O(n^{1+1/k})$ edges, for $k > 2$. In the affirmative, this would generalize the result of [13,14] ($k = 2$) and implies also the observation of [16]. Another interesting question is whether the $O(n^{1+1/k})$ edge bound for multiplicative $(2k-1)$-spanner is tight or not. This bound directly relies on an 1963 Erdös Conjecture [20] on the existence of graphs with $\Omega(n^{1+1/k})$ edges and girth at least $2k+2$. This has been proved only for $k = 1, 2, 3$ and $k = 5$ [21].

Better bounds can be achieved if we restrict spanners to be trees [22], or if particular classes of graphs are considered like: planar graphs [11] and more structured graphs (e.g., see [23] and [24] for a survey). Among them, the class of *chordal* graphs is of particular interests [7,6,25]. A graph is *k-chordal* if its induced cycles are of length at most k. Chordal graphs coincide with 3-chordal. Here below are summarized the best constructions for k-chordal graphs.

Chordal	(s, r)-spanner	Edges	Reference
3	$(2, 0)$	$\Theta(n^{3/2})$	[6]
3	$(3, 0)$	$O(n \log n)$	[6]
3	$(1, 3)$	$O(n \log n)$	[25]
k	$(1, k+1)$	$\Theta(n)$	[25], $k \geqslant 3$

Tree-decomposition is a rich concept introduced by Robertson and Seymour [26] and is widely used to solve various graph problems. In particular efficient algorithms exist for graphs having a tree-decomposition into subgraphs (or *bags*) of bounded size, i.e., for bounded *tree-width* graphs.

The *tree-length* of a graph G is the smallest integer δ for which G admits a tree-decomposition into bags of diameter at most δ. It has been formally introduced in [27], and extensively studied in [1,28,29]. Chordal graphs are exactly the graphs of tree-length 1, since a graph is chordal if and only if it has a tree-decomposition in cliques (cf. [30]). AT-free graphs, permutation graphs, and distance-hereditary graphs are of tree-length 2. More generally, [31] showed that k-chordal graphs have tree-length at most $k/2$. However, there are graphs with bounded tree-length and unbounded chordality,[4] like the wheel. In fact, there are infinitely many unbounded chordality graphs of bounded diameter and thus of bounded tree-length. For instance, any graph G can be transformed to a graph G' of tree-length at most 2 by adding a new vertex adjacent to all old vertices (universal vertex to G). So, if the chordality of G is k, then the chordality of G' is k, too. And, the tree-length of G' is at most 2. So, bounded tree-length graphs is a much larger class than bounded chordality graphs.

For several problems involving distance computation, like the design of approximate distance labeling schemes [31] or of near-optimal routing schemes [28], tree-length δ graphs are a natural generalization of chordal graphs, and their tree-decomposition induced can be successfully used. In this paper we highlight a new property of bounded tree-length

[1] The bound of [12] and [13] was $\widetilde{O}(n^{3/2})$ edges ($\widetilde{O}$ is similar to Big-O notation up to a poly-logarithmic factor), computable in time $\widetilde{O}(n^{5/2})$ and $\widetilde{O}(n^2)$ respectively. The bound of [14] is $O(n^{3/2})$, computable in time $O(n^{5/2})$.

[2] This observation due to [16] is based on the classical result (see [19]) that every graph with at least $\frac{1}{2}n^{1+1/k}$ edges has a cycle of length at most $2k$, for every $k \geqslant 1$. [17] and [18] gave respectively an $O(kmn^{1/k})$ and $O(km)$ time algorithm for the construction of such spanner.

[3] Actually $\beta(\epsilon, k) = k^{\max\{\log\log k - \log \epsilon, \log 6\}}$ for $2 \leqslant k \leqslant \log n$ and fixed $0 < \epsilon < 1$.

[4] The *chordality* is the smallest k such that the graph is k-chordal.

graphs: the design of sparse additive spanners. The following table summarizes the bounds we have obtained on the minimum number of edges of additive spanner.

Tree-length	(s, r)-spanner	Edges
δ	$(1, 2\delta)$	$O(\delta n + n \log n)$
δ	$(1, 4\delta)$	$O(\delta n)$
δ	$(\delta, 0)$	$\Omega(n^{1+\epsilon})$, $\epsilon \geqslant 1/\lceil \delta/2 \rceil$ for[5] $\delta = 1, 2, 3, 4, 5, 6, 9, 10$

Thus our first result provides an additive 2-spanner with $O(n \log n)$ edges for chordal graphs (for $\delta = 1$), improving [25] and also implying [6].

In this paper, we also compare our algorithm to the Chepoi–Dragan–Yan's algorithm (CDY) used successfully for k-chordal graphs [25]. For small chordality k, our algorithm produces an additive $2k$-spanner (or $(2k-2)$-spanner for odd k) with $O(n)$ edges (recall that $\delta \leqslant k/2$), whereas CDY's algorithm constructs an additive $(k+1)$-spanner with $O(n)$ edges, which is better for $k \geqslant 4$. However, we show in Section 4 that the CDY's algorithm cannot be used for bounded tree-length graphs of large chordality. More precisely, we construct a worst-case graph of tree-length 3 and chordality $\Omega(n^{1/3})$ for which the CDY's algorithm produces an $\Omega(n^{1/3})$-spanner with $O(n)$ edges. A generic algorithm would certainly combine both algorithms.

The lower bound shows that every additive $o(\delta)$-spanner requires $\Omega(n^{1+\epsilon})$ edges. However, combined with our two upper bounds, this naturally leads to the question of whether there exists, for every tree-length δ graph, an additive $O(\delta)$-spanner with $O(n \log n)$ or even with $O(n)$ edges.

The paper is organized as follows. In Section 2, we present all graph notions needed in this paper. In Section 3, we present the first algorithm (Line 1 of the previous table). Section 4 presents the second algorithms (Line 2) and the CDY's algorithm. We conclude in Section 5 with the lower bound.

2. Basic notions and notations

All graphs occurring in this paper are connected, finite, undirected and unweighted. Let $G = (V, E)$ be any graph, let X, Y be two subsets of V and let u be vertex of G. Then, the distance in G between u and X, denoted $d_G(u, X)$ is: $d_G(u, X) = \min_{v \in X} d_G(u, v)$. Moreover, the distance in G between X and Y is: $d_G(X, Y) = \min_{u \in X} d_G(u, Y)$.

A *shortest path spanning tree* T of a graph G is a rooted tree having the same vertex set as G and such that for every vertex u, $d_G(u, r) = d_T(u, r)$ where r is the root of T.

In the following, we will use the standard notions of *parent, children, ancestor, descendant* and *depth* in trees. The *nearest common ancestor* between two vertices u, v in a tree T is denoted by $\mathrm{NCA}_T(u, v)$.

2.1. Tree-decomposition and tree-length

In their work on graph minors [26], Robertson and Seymour introduce the notion of *tree-decomposition*. A tree-decomposition of a graph G is a tree T whose nodes, called *bags*, are subsets of $V(G)$ such that:

(1) $\bigcup_{X \in V(T)} X = V(G)$;
(2) for all $\{u, v\} \in E(G)$, there exists $X \in V(T)$ such that $u, v \in X$; and
(3) for all $X, Y, Z \in V(T)$, if Y is on the path from X to Z in T then $X \cap Z \subseteq Y$.

The *length* of a tree-decomposition T of a graph G is $\max_{X \in V(T)} \max_{u,v \in X} d_G(u, v)$, and the *tree-length of* G is the minimum of the length, over all tree-decompositions of G.

A well-known invariant related to tree-decompositions of a graph G is the *tree-width*, defined as minimum of $\max_{X \in V(T)} |X| - 1$ over all tree-decompositions T of G. We stress that the tree-width of a graph is not related to its tree-length. For instance cliques have unbounded tree-width and tree-length 1, whereas cycles have tree-width 2 and unbounded tree-length. Interestingly, the tree-length of a graph can be approximated in polynomial time within a constant factor [29] whereas such an approximation factor is unknown for the tree-width.

[5] The last line relies on the Erdös's Conjecture, but in any case $\epsilon \geqslant 1/\Theta(\delta)$ (cf. Table of Corollary 1).

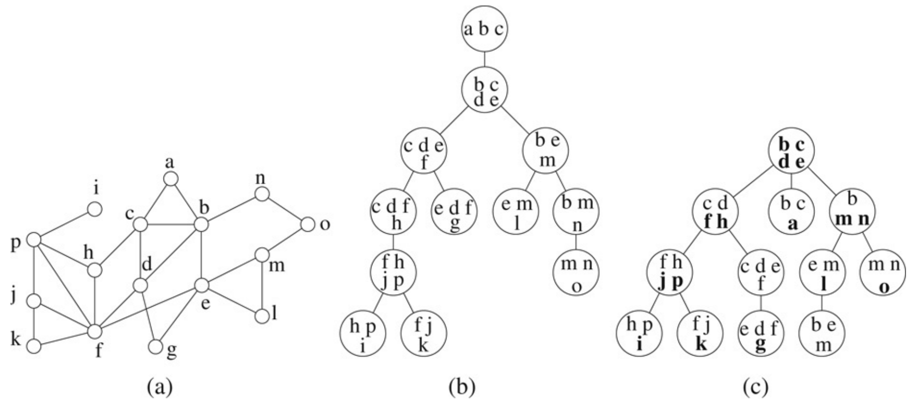

Fig. 1. A tree-length 2 graph G, a tree-decomposition T of G and a hierarchical tree of T.

A tree-decomposition is *reduced* if any bag is contained in no other bags. A leaf of such decomposition contains necessarily a vertex contained in none other bags. Thus, by induction, the number of bags of a reduced tree-decomposition does not exceed $\max\{n-1, 1\}$ for an n-vertex connected graph (cf. [32]).

2.2. Hierarchical tree

It is well known that every tree T has a vertex u, called *median*, such that each connected component of $T \setminus \{u\}$ has at most $\frac{1}{2}|V(T)|$ vertices. A *hierarchical tree* of T is a rooted tree H defined as follows: the root of H is the median of T, u, and its children are the roots of the hierarchical trees of the connected components of $T \setminus \{u\}$. Observe that T and H share the same vertex set, and the depth of H is at most[6] $\log|V(T)|$.

Property 1. *Let H be a hierarchical tree of a tree T. Then let U, V be two vertices of T and let Q be the path in T from U to V, and let $Z = \mathrm{NCA}_H(U, V)$. Then, $Z \in Q$, and Z is an ancestor in H of all the vertices of Q.*

Proof. By construction, the subtree induced by Z and its descendants in H is a connected component of T, say A. Thus, Z, U, V are in A, but U and V are in two different components of $T \setminus \{Z\}$. Thus in T, the path Q from U to V is wholly contained in A and intersects Z. So, $Z \in Q$ and Z is ancestor of all vertices of Q in H. □

2.3. k-chordal graphs

A graph G is *k-chordal* if the length of the longest induced cycle of G is at most k. This class of graphs is also discussed under the name k-bounded-hole graphs in [33]. Chordal graphs are 3-chordal graphs.

The *chordality of* G is the smallest integer k such that G is k-chordal. Trees are, by convention, of chordality 2.

2.4. Layering tree

Let G be a graph with a distinguished vertex s. Then we partition $V(G)$ into *layers*: for every integer $i \geqslant 0$, $L^i = \{u \in V(G) \mid d_G(s, u) = i\}$. Then, each layer L^i is partitioned into $L^i_1, \ldots, L^i_{p_i}$, such that two vertices stay in a same part if and only if they are connected by a path visiting only vertices at distance at least i from s.

A *layering tree* of G, denoted as LT, is the graph whose vertex set is the collection of all the parts L^i_j. In LT, two vertices L^i_j and $L^{i'}_{j'}$ are adjacent if and only if there exists $u \in L^i_j$ and $v \in L^{i'}_{j'}$ such that u and v are adjacent in G (see Fig. 2 for an example). The vertex s is called the *source* of LT.

Lemma 1 (*[34]*). *For any graph G,* LT *is a tree computable in linear time.*

6 All the logs are in base two.

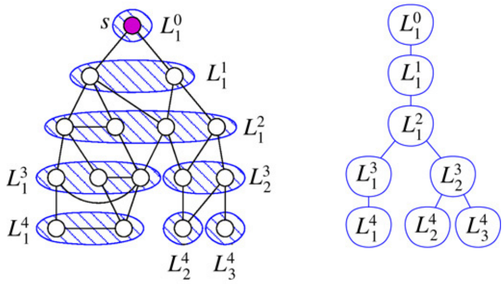

Fig. 2. A 5 chordal graph and a layering tree of it.

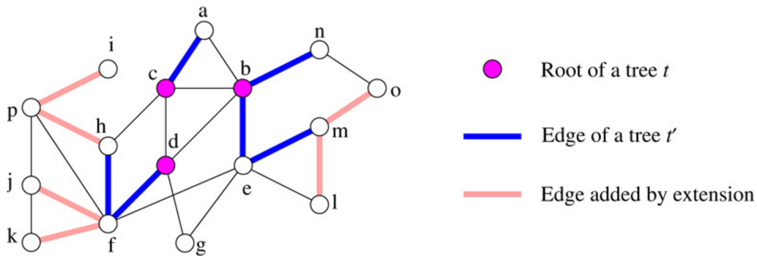

Fig. 3. The trees $t(cdfh)$, $t(bca)$, and $t(bmn)$.

3. Additive 2δ-spanner with $O(\delta n + n \log n)$ edges

Theorem 1. *Every n-vertex graph of tree-length δ has an additive 2δ-spanner with at most $(\delta + \log n)(n-1)$ edges.*

The remaining of this section concerns the proof of Theorem 1.

From now, G is a graph with n vertices and of tree-length δ. T denotes a reduced tree-decomposition of G of length δ, and H denotes a hierarchical tree of T. So, the depth of H is at most $\log n$.

For every vertex u of G, we define the *bag* of u, denoted by $\mathcal{B}(u)$, is a bag X of H of minimum depth such that $u \in X$. Observe that, even if the set of bags containing u in H does not induce a connected subtree, Property 1 implies that $\mathcal{B}(u)$ is defined and unique for every u.

For every bag X of H, we define the subbag X^- as follows: $X^- = \{u \in X \mid \mathcal{B}(u) = X\}$ (bold vertices in Fig. 1(c)) and the set $\downarrow X = \{u \in V(G) \mid \mathcal{B}(u)$ is a descendant of X in $H\}$ (here we assume that X is a descendant of itself). Note that $G[\downarrow X]$ is not necessary connected, for example in Fig. 1, $\downarrow(cdfh) = \{fghijkp\}$ which is not connected.

With every bag X of H, we also associate a local tree $t(X)$ constructed as follows (see Fig. 3):

(1) Let $t'(X)$ be a tree obtained from a shortest path spanning tree of G, rooted at an arbitrary vertex $r_X \in X$, by recursively removing each leaf f which is not in X^-,
(2) $t(X)$ is the tree $t'(X)$ extended by a breadth-first search in $G[\downarrow X]$ started at the set $V(t'(X)) \cap \downarrow X$. Recall that $G[\downarrow X]$ is not necessary connected, thus $t(X)$ spans only the vertices of its connected components which contain vertices of $t'(X)$ (in Fig. 3, the vertex g is not spanned by $t(cdfh)$).

By construction of $t(X)$ one can prove the following lemma:

Lemma 2. *Let X be a bag of H, and u be a node of $t(X)$:*

(1) *either u belongs to $t'(X)$ and then $d_{t'(X)}(u, r_X) = d_{t(X)}(u, r_X) = d_G(u, r_X) \leqslant \delta$, moreover u has a descendant in (or belongs to) X^-,*
(2) *or $u \in \downarrow X$ and then u has an ancestor in $t'(X)$. Let v be the closest one from u, we have: $d_{t(X)}(u, v) = d_{G[\downarrow X]}(u, v) \leqslant d_{G[\downarrow X]}(u, X^-)$.*

The spanner of G claimed by Theorem 1 is simply the graph G' defined by $G' = \bigcup_{X \in V(H)} t(X)$.

Lemma 3. *G' is an additive 2δ-spanner of G.*

Proof. Let u, v be two vertices of G. Let $P = x_0, x_1, \ldots, x_l$ be a shortest path in G from $u = x_0$ to $v = x_l$. Let X be the bag of minimum depth in H among $\{\mathcal{B}(x_i) \mid i = 0, \ldots, l\}$, note that by Property 1, X exists.

Let $x \in P$ such that $\mathcal{B}(x) = X$ (i.e., $x \in X^-$). Property 1 implies that P is wholly contained in $\downarrow X$. Thus both u, v belong to $t(X)$ and $d_G(u, v) = d_{G[\downarrow X]}(u, v)$.

Moreover, $d_{G'}(u, v) \leqslant d_{t(X)}(u, v) \leqslant d_{t(X)}(u, r_X) + d_{t(X)}(r_X, v)$. We will prove that $d_{t(X)}(u, r_X) \leqslant d_G(u, x) + \delta$ and $d_{t(X)}(r_X, v) \leqslant d_G(v, x) + \delta$. In this way we will get that $d_{G'}(u, v) \leqslant d_G(u, v) + 2\delta$ as claimed.

If u belongs to $t'(X)$ then, by Lemma 2, $d_{t(X)}(u, r_X) = d_G(u, r_X) \leqslant \delta \leqslant d_G(u, x) + \delta$. Otherwise, let u' be the nearest ancestor of u in $t(X)$ which belongs to $t'(X)$. Lemma 2 implies that $d_{t(X)}(u, u') \leqslant d_{G[\downarrow X]}(u, X^-)$, but $d_{G[\downarrow X]}(u, X^-) \leqslant d_{G[\downarrow X]}(u, x) = d_G(u, x)$. So, we conclude that $d_{t(X)}(u, r_X) \leqslant d_G(u, x) + d_G(u', r_X) \leqslant d_G(u, x) + \delta$.

In the previous paragraph we can replace u with v and u' with v' to prove that $d_{t(X)}(v, r_X) \leqslant d_G(v, x) + \delta$. This completes the proof. $\square$

Lemma 4. *G' has at most $(\delta + \log n)(n - 1)$ edges.*

Proof. We will count the edges of G' by studying the two steps of the construction of trees $t(X)$.

Let X be a bag of H and u be a vertex of X^-. Recall that $r_X \in X$, thus $d_G(u, r_X) \leqslant \delta$. Thus the tree $t'(X)$ contains at most $\delta|X^-|$ edges, except when X is the root of H. In this latter case, since $X = X^-$, $r_X \in X^-$ and so $t'(X)$ has at most $\delta(|X^-| - 1)$ edges. Moreover, by definition, the X^- sets are pairwise disjoint, thus $\bigcup_{X \in V(H)} t'(X)$ contains at most $\delta(n - 1)$ edges.

Let $X_1^i, \ldots, X_{p_i}^i$ be the bags of depth i in H. By construction, $\downarrow X_1^i, \ldots, \downarrow X_{p_i}^i$ are pairwise disjoint. Thus the extension of the corresponding trees $t'(X_1^i), \ldots, t'(X_{p_i}^i)$ spans different vertices of G, so this extension adds at most $(n - 1)$ edges. Moreover, note that the extension of a tree $t'(X)$, where X is a leaf of H, does not add any edge, and recall that H is of depth at most $\log n$. It follows that the total number of edges added by all the extensions is at most $(n - 1)\log n$.

We can conclude that G' has at most $\delta(n - 1) + (n - 1)\log n$ edges, as claimed. $\square$

Theorem 1 directly follows from Lemmas 3 and 4. $\square$

Remark 1. Recall that in our algorithm, the root r_X of a tree $t(X)$ is chosen arbitrarily in X, in order to insure that for every vertex $u \in X^-$, $d_G(u, r_X) \leqslant \delta$. So if every bag X is of radius r, i.e., if there exists $c \in V(G)$ such that $\forall u \in X$, $d_G(c, u) \leqslant r$, then r_X can be set to the center of X. In this way, the spanner G' we obtain is an additive $2r$-spanner with at most $(rn + (n - 1)\log n)$ edges.

Remark 2. Note also that if for each bag X, the tree $t'(X)$ is contained in $\downarrow X$ then G' has no more than $(n - 1)\log n$ edges.

4. Additive spanner with a linear number of edges

In this section we present an algorithm to construct, for any tree-length δ graph, an additive 4δ-spanner with $O(\delta n)$ edges. Since for every graph the tree-length is at most half the chordality, for a k-chordal graph we obtain an additive $2k$-spanner (or $(2k - 2)$-spanner of odd k) with $O(kn)$ edges. This latter result is far from the optimal, because Chepoi et al. [25] have presented an algorithm which computes, for any graph of chordality k, an additive $(k + 1)$-spanner with $O(n)$ edges. Nevertheless we show that their algorithm is not designed efficient for tree-length δ graphs. Indeed there exist tree-length 3 graphs for which their algorithm returns an additive $\Omega(n^{1/3})$-spanner whereas our algorithm guarantees an additive 12-spanner[7] with $O(n)$ edges.

From now, G is a graph with n vertices and of tree-length δ. LT denotes a layering tree of G of source s (see Section 2).

[7] Actually, on the counter-example, it produces an additive 3-spanner.

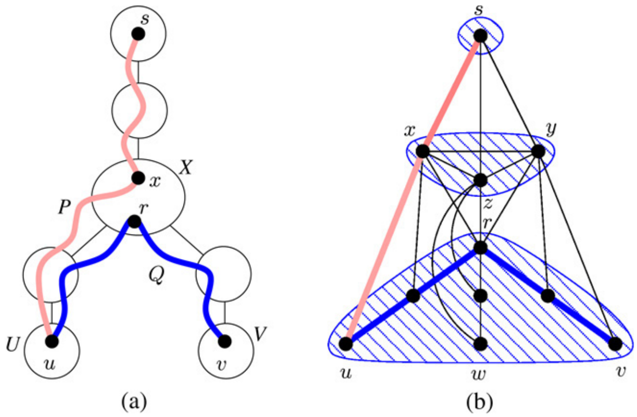

Fig. 4. A part of LT is of diameter at most 3δ and of radius at most 2δ.

4.1. Additive $O(\delta)$-spanner with $O(\delta n)$ edges

Theorem 2. *Every n-vertex graph of tree-length δ has an additive 4δ-spanner with at most $(2\delta + 1)(n - 1)$ edges.*

Lemma 5. *For every part W of* LT*, there is a vertex r of G, called* center *of W, such that for all $u, v \in W$, $d_G(u, v) \leqslant 3\delta$ and $d_G(u, r) \leqslant 2\delta$. Moreover, for every δ, these bounds are best possible.*

Proof. Let T be a tree-decomposition of G of length δ. W.l.o.g., T is supposed to be rooted at a bag containing s, the source of LT. Let W be a part of LT at distance i from s. Let X be the bag of T that is the nearest common ancestor of all the bags containing vertices of W, and let $d_X = \max_{u,v \in X} d_G(u, v)$ be its diameter. Let us prove that for every $u \in W$, $d_G(u, X) \leqslant \delta$. In this way, we will prove that:

- $\forall u, v \in W$, $d_G(u, v) \leqslant d_G(u, X) + d_X + d_G(v, X) \leqslant 3\delta$;
- $\forall u \in W$ and $\forall r \in X$, $d_G(u, r) \leqslant d_G(u, X) + d_X \leqslant 2\delta$.

Let u be an arbitrary vertex of W. Consider a vertex $v \in W$ such that there are two bags, U and V, such that: $u \in U$, $v \in V$, and $X = \mathrm{NCA}_T(U, V)$ (we check that v, U, V exist). Let P be a shortest path from s to u, P intersects X. Let x be the closest from s vertex in $P \cap X$. Since u, v are both in W, there exists a path Q from u to v using only intermediate vertices w such that $d_G(s, w) \geqslant i$. Let Q intersect X at a vertex r (see Fig. 4(a)).

Note that $d_G(s, u) = i = d_G(s, x) + d_G(x, u)$ and $d_G(s, r) \leqslant d_G(s, x) + \delta$. So, $d_G(s, r) \leqslant i + \delta - d_G(x, u)$. If $d_G(x, u) \geqslant \delta + 1$ then $d_G(s, r) \leqslant i - 1$: a contradiction since $r \in Q$. So, $d_G(u, X) \leqslant d_G(u, x) \leqslant \delta$ as claimed.

These bounds are best possible for each $\delta \geqslant 1$. For $\delta = 1$, the graph depicted on Fig. 4(b) is chordal, u, v, w belong to the same part and $d_G(u, v) = d_G(u, w) = d_G(v, w) = 3$. By replacing each edge by a path of length δ, the tree-length of this subdivision increases to δ, u, v, w still belong to the same part and are at distance 3δ. We check that a center c for W can be chosen arbitrarily among $\{x, y, z, r, s\}$ and attains a radius 2δ. Moreover, if $c \notin \{x, y, z, r, s\}$, one can prove that either $d_G(u, c) > 2\delta$ or $d_G(v, c) > 2\delta$ or $d_G(w, c) > 2\delta$. Thus the radius of the part containing u, v, w is exactly 2δ. □

The spanner satisfying Theorem 2 is simply the graph defined by $G' := S \cup \bigcup_{W \in V(\mathrm{LT})} S_W$, where S is a shortest path tree spanning G and rooted at s, and S_W is a shortest path tree spanning of W rooted at a center of W ($r_X \notin X$ possible).

Lemma 6. *G' is an additive 4δ-spanner of G.*

Proof. Let u, v be two vertices of G, and let A, B be the two parts of LT containing respectively u and v. Let us show that every path from u to v must intersect the part $W = \mathrm{NCA}_{\mathrm{LT}}(A, B)$.

This clearly holds if $W = A$ or $W = B$. If $W \neq A$ and $W \neq B$, (so in particular $A \neq B$), then, by definition of LT, every intermediate vertex of a path from u to v must intersect an ancestor of A and of B. So, by induction, it must intersect the nearest common ancestor of A and of B, W.

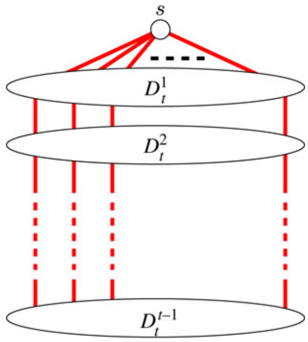

Fig. 5. Counter-example G_0.

Let $u', v' \in W$ be the ancestors in tree S of u and v. We observe that $d_G(u, u') = d_{\text{LT}}(A, W)$. Indeed, since G' contains a shortest path spanning tree of G rooted at s, it follows that $d_{\text{LT}}(A, W) = d_{G'}(u, u')$, and finally, $d_G(u, u') = d_{G'}(u, u')$. Similarly, $d_G(v, v') = d_{G'}(v, v')$, and thus $d_{G'}(u, u') + d_{G'}(v, v') \leqslant d_G(u, v)$.

Using the tree S_W contained in G' and rooted at the center of W, and by Lemma 5, we have $d_{G'}(u', v') \leqslant 4\delta$. Therefore, we obtain:

$$d_{G'}(u, v) \leqslant d_{G'}(u, u') + d_{G'}(u', v') + d_{G'}(v', v) \leqslant d_G(u, v) + 4\delta \ . \quad \square$$

Lemma 7. *G' has at most $(2\delta + 1)(n - 1)$ edges.*

Proof. S has $n - 1$ edges. Every tree S_W has at most $|W|$ leaves and so at most $2\delta|W|$ edges, except when W is the root of LT. In this latter case, S_W contains no edges. The parts of LT are disjoint, so the number of edges of G' is at most $n - 1 + 2\delta(n - 1) = (2\delta + 1)(n - 1)$. $\square$

4.2. CDY's algorithm on graphs of bounded tree-length

Theorem 3. *There is a graph of tree-length 3 and with $n + o(n)$ vertices for which every execution of the CDY's algorithm proposed in [25], constructs an additive $\Omega(n^{1/3})$-spanner with $O(n)$ edges.*

First of all, let us present the CDY's algorithm of [25]:

(1) For every part W of LT, let $r_W \in W$ be a vertex chosen in advance.[8]
(2) $E(G') \leftarrow E(S)$ where S is shortest path spanning tree of G rooted at s.
(3) For all $W \in V(LT)$ and all $u \in W$ do
 $E(G') \leftarrow E(G') \cup \{x, y\}$, where x is the nearest ancestor in S of u having a neighbor y ancestor in S of r_W.

The remaining of this section is devoted to the proof of Theorem 3.

The *Cartesian product* of two graphs A and B is the graph denoted by $A \times B$ such that $V(A \times B) = \{(x, y) \mid x \in V(A), y \in V(B)\}$, and $E(A \times B) = \{((x, x'), (y, y')) \mid (x = x' \text{ and } (y, y') \in E(B)) \text{ or } (y = y' \text{ and } (x, x') \in E(A))\}$. E.g., the mesh is the Cartesian product of two paths. Let K_t and P_t denote, respectively, the complete graph and the path with t vertices.

We set $D_p = K_p \times K_p$. The counter-example, denoted by G_0, is the graph $D_t \times P_{t-1}$, so composed of $t - 1$ copies $D_t^1, \ldots, D_t^{t-1}$ of D_t, with an extra vertex s connected to all the vertices of D_t^1 (see Fig. 5). To every vertex u of G_0, $u \neq s$, we denote by $P(u)$ the copy of the path P_{t-1} containing u. Hereafter, we set $t := \lceil n^{1/3} \rceil$, so that G_0 has $t^2(t - 1) + 1 = n + O(n^{2/3})$ vertices.

A subgraph H of a graph G is *isometric* if $d_H(x, y) = d_G(x, y)$, for all $x, y \in V(H)$. It is a natural generalization of induced subgraph (any isometric subgraph is clearly an induced subgraph). We have:

Lemma 8 ([27]). *The tree-length of any isometric subgraph of G is no more than the tree-length of G.*

[8] In the original algorithm r_W is given by a fixed policy we do not detail here. Here, we consider that any vertex of W can be chosen, it is simpler and equally powerful.

Let u, v, w be three vertices of some D_t^i inducing a path of length two (since D_t is of diameter two, such vertices exist). We check that the graph induced by the vertices of the paths $P(u)$, $P(v)$, $P(w)$ is an isometric mesh of G_0. This mesh has $t-1$ rows and 3 columns.

Lemma 9. *G_0 has chordality at least $2t = \Omega(n^{1/3})$, and tree-length 3 for $t \geqslant 5$.*

Proof. In a $(t-1) \times 3$ mesh, the perimeter is an induced cycle of the mesh of length $2t$. Since this mesh is an isometric subgraph of G_0, it follows that G_0 is of chordality at least $2t$.

It is proved in [27] that the tree-length of the mesh with p rows and q columns is $\min\{p, q\}$ if $p \neq q$ or p is even, and is $p-1$ otherwise. In particular, the $(t-1) \times 3$ mesh has tree-length 3 if $t \geqslant 5$. By Lemma 8, G_0 has tree-length at least 3 for $t \geqslant 5$.

We obtain a tree-decomposition of G_0 of length 3 by considering a path $X_0, X_1, \ldots X_{t-2}$ where $X_0 = \{s\} \cup V(D_t^1)$ and $X_i = V(D_t^i) \cup V(D_t^{i+1})$ for $i \geqslant 1$. □

A *dominating set* of a graph G is a set of vertices R such that for every vertex u of G either $u \in R$ or u is adjacent to a vertex of R.

Lemma 10. *If R is a dominating set of D_t, then $|R| \geqslant t$.*

Proof. The graph D_t is the union of two disjoint sets of cliques $\mathcal{K}_1$ and $\mathcal{K}_2$, each one composed of t disjoint copies of K_t, so that every edge belongs either to a clique of $\mathcal{K}_1$ or of $\mathcal{K}_2$. Every clique of $\mathcal{K}_1$ intersects each clique of $\mathcal{K}_2$ and vice versa. Assume $|R| < t$. By the Pigeon Hole Principle, there is a clique $A \in \mathcal{K}_1$ with no vertices of R. Similarly, there is a clique $B \in \mathcal{K}_2$ with no vertices of R. The cliques A and B share exactly one vertex, say u (otherwise there would exist an edge that belongs to a clique of $\mathcal{K}_1$ and to a clique of $\mathcal{K}_2$). All the incident edges of u belong either to A or to B. It follows that u is not adjacent to any vertex of R: a contradiction. □

Proof of Theorem 3. Let G' be the spanner obtained by CDY's algorithm applied on the source s of G_0. The parts of H are the sets $L^0 = \{s\}$ and $L^i = V(D_t^i)$ for $i \geqslant 1$. The spanning tree S rooted at s used in G' contains exactly the edges incident to s and the edges of the paths P_{t-1}. No edge of any D_t^i is contained in S.

We assume that a special vertex r_i has been arbitrarily selected for each part L^i of H, and let $R = \{r_1, \ldots, r_{t-1}\}$.

Let $u \in L^i$ be a vertex of G_0, $i \neq 0$. Observe that if u and r_i are not adjacent in G_0, then there is no edge in G_0 (and thus in G') between $P(u)$ and $P(r_i)$. Let R' be the projection of R on D_t^{t-1}: $R' = \{u \in D_t^{t-1} \mid V(P(u)) \cap R \neq \varnothing\}$. $|R'| \leqslant |R| = t-1$, so R' is not a dominating set of D_t^{t-1} (Lemma 10).

Let v be a vertex of D_t^{t-1} with no neighbors in R', and let $r' \in R'$. From the above observation, in G', there is no edge between $P(v)$ and $P(r')$. During the second phase of the CDY's algorithm, only edges incident to r_i are added, for all $i \geqslant 1$. It follows that every vertex of $P(v)$ has no incident edges in G', except those of $P(v)$. So, $d_{G'}(v, r') \geqslant 2(t-1) = \Omega(n^{1/3})$ whereas $d_{G_0}(v, r') = 2$. G' is an additive $\Omega(n^{1/3})$-spanner, as claimed. □

5. Lower bound

Let $m(n, g)$ be the maximum number of edges contained in a graph with n vertices and of girth at least g. It is clear that there exists an n-vertex graph for which every additive $(g-3)$-spanner (or multiplicative $(g-2)$-spanner) needs $m(n, g)$ edges. Indeed, any graph G of girth g and of $m(n, g)$ edges has no proper additive $(g-3)$-spanner: removing any edge $\{u, v\}$ of G implies $d_H(u, v) \geqslant g - 1 = d_G(u, v) + g - 2$ and thus H is not an additive $(g-3)$-spanner of G.

Theorem 4. *For each $\delta \geqslant 1$, there exists a graph of $n + 3\delta - 2$ vertices and of tree-length δ for which every multiplicative δ-spanner (and thus every additive $(\delta - 1)$-spanner) needs $m(n, \delta + 2) + 3\delta - 1$ edges.*

Proof. Consider a graph G with n vertices, girth at least $\delta + 2$, and with $m(n, \delta + 2)$ edges. The diameter of G is at most δ. Indeed, otherwise G has two vertices, say u and v, at distance $\delta + 1$. So, augmenting G by the edge u, v would provide a graph with n vertices, girth at least $\delta + 2$, and with $m(n, \delta + 2) + 1$ edges: a contradiction with the definition of $m(n, \delta + 2)$. So, G is of tree-length at most its diameter, that is $\leqslant \delta$.

Now we construct a graph G^* obtained from G by selecting an edge of G, say $\{u, v\}$, and by adding a path of length $3\delta - 1$, so that G^* contains a cycle C of length 3δ. The graph G^* has $n + 3\delta - 2$ vertices, girth at least $\delta + 2$, and $m(n, \delta + 2) + 3\delta - 1$ edges. Again, G^* does not contain any proper multiplicative δ-spanner.

The tree-length of G^* is exactly δ observing that the tree-length of a graph composed of two subgraphs, say G and C, sharing a vertex or an edge is the maximum between the tree-length of G and the tree-length of C (because G and C are isometric subgraphs, and the common vertex or edge can be used to combine both optimal tree-decompositions). As shown in [27], the tree-length of a cycle of length $k = 3\delta$ is $\lceil k/3 \rceil = \delta$. □

An Erdös Conjecture [20] claims existence of n-vertex graphs with $\Omega(n^{1+1/k})$ edges and of girth at least $2k + 2$. This has been proved only for $k = 1, 2, 3$ and $k = 5$ [21]. It is known however that there are graphs of girth at least $2k + 2$ with $\Omega(n^{1+1/(2k)})$ edges. From Theorem 4, we have:

Corollary 1. *For every constant $\delta \geqslant 1$, there are graphs with $O(n)$ vertices and tree-length δ for which every multiplicative δ-spanner requires $\Omega(n^{1+\epsilon})$ edges, where $\epsilon \geqslant 1/\lceil \delta/2 \rceil$ for $\delta \leqslant 6$. Moreover, for every δ, $\epsilon \geqslant 1/\Theta(\delta)$, where the best current lower bound on ϵ is given by the table below.*

Proof. For each fixed integer $k \geqslant 1$, let $f(k)$ be the largest real such that there exists an n-vertex graph of girth at least $2k + 2$ and with $\Omega(n^{1+f(k)})$ edges. We have $m(n, 2k + 2) = \Omega(n^{1+f(k)})$.

Consider the worst-case graph G_δ given by Theorem 4. It has at most $5n/2$ vertices (recall that $\delta < n/2$ because the chordality of a graph is at most $n - 1$), and at least $m(n, \delta + 2)$. Note that $m(n, \delta + 2) \leqslant m(n, \delta + 1)$. So, G_δ has at least $m(n, 2\lceil \delta/2 \rceil + 2) = \Omega(n^{1+f(\lceil \delta/2 \rceil)})$ edges.

It is known that $f(k) = 1/k$ for all $k \geqslant 1$, if the Erdös's Conjecture holds. The following table summarizes the best known lower bounds on $f(k)$. Complete references can be found in [17].

$k = \lceil \delta/2 \rceil$	$f(k)$
$1, 2, 3, 5$	$= 1/k$
4	$\geqslant 1/(k + 1)$
$6, 7$	$\geqslant 1/(k + 2)$
$k = 2r,\ r \geqslant 4$	$\geqslant 1/(3k/2 - 1)$
$k = 2r - 1,\ r \geqslant 5$	$\geqslant 1/(3k/2 - 3/2)$

□

6. Conclusion

In this paper we showed that any n-vertex graph of tree-length δ admits an additive 2δ-spanner with $O(\delta n + n \log n)$ edges and an additive 4δ-spanner with $O(\delta n)$ edges. These results provided new upper bounds for chordal graphs ($\delta = 1$). We have also shown a lower bound which says that there are graphs of tree-length δ for which every multiplicative δ-spanner (and thus every additive $(\delta - 1)$-spanner) must have $\Omega(n^{1+1/\Theta(\delta)})$ edges.

We conclude this paper with few open problems.

(1) Do the tree-length δ graphs admit additive 2δ-spanners with $O(n \log n)$ edges (δ independent number of edges)? This was the case for the k-chordal graphs [2].
(2) What about additive 4δ-spanners with $O(n)$ edges? Again, this was the case for the k-chordal graphs [25].
(3) Do the tree-length δ graphs admit sparse additive δ-spanners?
(4) Given an arbitrary graph G, what is the complexity of constructing a tree-decomposition of G into bags of smallest diameter δ?

Acknowledgement

The first author's work was partially supported by the Research and Training Network COMBSTRU (HPRN-CT-2002-00278).

References

[1] Y. Dourisboure, C. Gavoille, Sparse additive spanners for bounded tree-length graphs, in: 11th International Colloquium on Structural Information & Communication Complexity (SIROCCO), in: Lecture Notes in Computer Science, vol. 3104, Springer, 2004, pp. 123–137.
[2] F.F. Dragan, C. Yan, I. Lomonosov, Collective tree spanners of graphs, in: T. Hagerup, J. Katajainen (Eds.), 9th Scandinavian Workshop on Algorithm Theory, SWAT, in: Lecture Notes in Computer Science, vol. 3111, Springer, 2004, pp. 64–76.
[3] D. Peleg, J.D. Ullman, An optimal synchronizer for the hypercube, SIAM Journal on Computing 18 (1989) 740–747.

[4] D. Peleg, Distributed computing: A locality-sensitive approach, in: SIAM Monographs on Discrete Mathematics and Applications, 2000.
[5] D. Peleg, E. Upfal, A trade-off between space and efficiency for routing tables, Journal of the ACM 36 (3) (1989) 510–530.
[6] D. Peleg, A.A. Schäffer, Graph spanners, Journal of Graph Theory 13 (1) (1989) 99–116.
[7] A. Brandstädt, F.F. Dragan, H.-O. Le, V.B. Le, Tree spanners on chordal graphs: Complexity and algorithms, Theoretical Computer Science 310 (2004) 329–354.
[8] U. Brandes, D. Handke, NP-completeness results for minimum planar spanners, Discrete Mathematics & Theoretical Computer Science 3 (1) (1998) 1–10.
[9] L. Cai, D.G. Corneil, Tree spanners, SIAM Journal on Discrete Mathematics 8 (3) (1995) 359–387.
[10] Y. Dodis, S. Khanna, Designing networks with bounded pairwise distance, in: 30th Annual ACM Symposium on Theory of Computing, STOC, 1999, pp. 750–759.
[11] S.P. Fekete, J. Kremer, Tree spanners in planar graphs, Discrete Applied Mathematics 108 (2001) 85–103.
[12] D. Aingworth, C. Chekuri, P. Indyk, R. Motwani, Fast estimation of diameter and shortest paths (without matrix multiplication), SIAM Journal on Computing 28 (4) (1999) 1167–1181.
[13] D. Dor, S. Halperin, U. Zwick, All-pairs almost shortest paths, SIAM Journal on Computing 29 (2000) 1740–1759.
[14] M. Elkin, D. Peleg, $(1+\epsilon, \beta)$-spanner constructions for general graphs, in: 33rd Annual ACM Symposium on Theory of Computing, STOC, Hersonissos, Crete, Greece, 2001, pp. 173–182.
[15] S. Baswana, T. Kavitha, K. Mehlhorn, S. Pettie, New constructions of (α, β)-spanners and purely additive spanners, in: 16th Symposium on Discrete Algorithms, SODA, ACM–SIAM, 2005, pp. 672–681.
[16] I. Althöfer, G. Das, D. Dobkin, D. Joseph, J. Soares, On sparse spanners of weighted graphs, Discrete & Computational Geometry 9 (1) (1993) 81–100.
[17] M. Thorup, U. Zwick, Approximate distance oracles, in: 33rd Annual ACM Symposium on Theory of Computing, STOC, Hersonissos, Crete, Greece, 2001, pp. 183–192.
[18] S. Baswana, S. Sen, A simple linear time algorithm for computing a $(2k-1)$-spanner of $O(n^{1+1/k})$ size in weighted graphs, in: 30th International Colloquium on Automata, Languages and Programming, ICALP, in: Lecture Notes in Computer Science, vol. 2719, Springer, 2003, pp. 384–396.
[19] N. Alon, S. Hoory, N. Linial, The Moore bound for irregular graphs, Graph and Combinatorics 18 (1) (2002) 53–57.
[20] P. Erdös, Extremal problems in graph theory, in: Publ. House Czechoslovak Acad. Sci., Prague, 1964, pp. 29–36.
[21] R. Wenger, Extremal graphs with no C^4's, C^6's, or C^{10}'s, Journal of Combinatorial Theory, Series B 52 (1) (1991) 113–116.
[22] D. Kratsch, H.-O. Le, H. Müller, E. Prisner, D. Wagner, Additive tree spanners, SIAM Journal on Discrete Mathematics 17 (2) (2004) 332–340.
[23] M. Madanlal, G. Venkatesan, C. Pandu Rangan, Tree 3-spanners on interval, permutation and regular bipartite graphs, Information Processing Letters 59 (2) (1996) 97–102.
[24] J. Soares, Graphs spanners: A survey, Congressus Numerantium 89 (1992) 225–238.
[25] V.D. Chepoi, F.F. Dragan, C. Yan, Additive sparse spanners for graphs with bounded length of largest induced cycle, Theoretical Computer Science 347 (2005) 54–75.
[26] N. Robertson, P.D. Seymour, Graph minors. II. Algorithmic aspects of tree-width, Journal of Algorithms 7 (1986) 309–322.
[27] Y. Dourisboure, C. Gavoille, Tree-decomposition of graphs with small diameter bags, in: J. Fila (Ed.), 2nd European Conference on Combinatorics, Graph Theory and Applications (EUROCOMB), 2003, pp. 100–104.
[28] Y. Dourisboure, Routage compact et longueur arborescente, Ph.D. Thesis, Université Bordeaux 1, Talence, France, Dec. 2003.
[29] Y. Dourisboure, C. Gavoille, Small diameter bag tree-decompositions, Research Report RR-1326-04, LaBRI, University of Bordeaux 1, Discrete Mathematics (in press).
[30] R. Diestel, Graph Theory, second edition, in: Graduate Texts in Mathematics, vol. 173, Springer, 2000.
[31] C. Gavoille, M. Katz, N.A. Katz, C. Paul, D. Peleg, Approximate distance labeling schemes, in: F.M. auf der Heide (Ed.), 9th Annual European Symposium on Algorithms, ESA, in: Lecture Notes in Computer Science, vol. 2161, Springer, 2001, pp. 476–488.
[32] H.L. Bodlaender, A linear time algorithm for finding tree-decompositions of small treewidth, SIAM Journal on Computing 25 (1996) 1305–1317.
[33] F. Gavril, Algorithms for maximum weight induced paths, Information Processing Letters 81 (4) (2002) 203–208.
[34] V.D. Chepoi, F.F. Dragan, A note on distance approximating trees in graphs, European Journal of Combinatorics 21 (2000) 761–768.