Sparse Additive Spanners for Bounded Tree-Length Graphs*

Yon Dourisboure and Cyril Gavoille

LaBRI, Université Bordeaux 1
{yon.dourisboure,gavoille}@labri.fr

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 $O(\delta n \log n)$ 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 a 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 [23,21]. Sparse spanners occur also in the efficiency, of compact routing schemes [24]. 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 [22], even if we restrict $r = 0$ (see also [6,5,7,11,18] for complexity issue). Best known results on (s,r)-spanners for general graphs are summarized in the following table.

(s,r)-spanner	edges	reference
$(1,2)$	$\Theta(n^{3/2})$	[12,16][1]
$(2k-1,0)$	$O(n^{1+1/k})$	[2,27,3][2]
$(k-1,2k-4)$	$O(kn^{1+1/k})$	[16], $k \geqslant 4$ even
$(k-2+\epsilon,2k-2-\epsilon)$	$O(\epsilon^{-1}kn^{1+1/k})$	[16], $k \geqslant 3$ odd, $\epsilon > 0$
$(k-1+\epsilon,2k-4-\epsilon)$	$O(\epsilon^{-1}kn^{1+1/k})$	[16], $k \geqslant 4$ even, $\epsilon > 0$
$(1+\epsilon,\beta(\epsilon,k))$	$O(\beta(\epsilon,k)n^{1+1/k})$	[16][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 [12,16] ($k = 2$) and implies also the observation of [2]. Another interesting question is to known whether the $O(n^{1+1/k})$ edge bound for multiplicative $(2k-1)$-spanner is tight or not. This bound

* Supported by the European Research Training Network COMBSTRU-HPRN-CT-2002-00278

[1] The bound of [12] was $O(n^{3/2}\log^{O(1)} n)$ edges but with a better running time, $O(n^{5/2})$ v.s. $O(n^2)$.

[2] This observation due to [2] is based on the classical result (see [1]) that every graph with at least $\frac{1}{2}n^{1+1/k}$ edges has a cycle of length at most $2k$, for every $k \geqslant 1$. [27] and [3] 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 $0 < \epsilon < 1$.

directly relies to an 1963 Erdös Conjecture [17] 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 $k = 5$.

Better bounds can be achieved if we restrict spanners to be trees, or if particular classes of graphs are considered: planar graphs [18], and more structured graphs (e.g., see [20] and [26] for a survey). Among them, the class of *chordal* graphs is of particular interests [6,22,9]. 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	$(2,0)$	$\Theta(n^{3/2})$	[22]
3	$(3,0)$	$O(n \log n)$	[22]
3	$(1,3)$	$O(n \log n)$	[9]
k	$(1,k+1)$	$\Theta(n)$	[9], $k \geqslant 3$

Tree-decomposition is a rich concept introduced by Robertson and Seymour [25] 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 [15], and extensively studied in [13]. 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. [10]). AT-free graphs, permutation graphs, and distance-hereditary graphs are of tree-length 2. More generally, [19] 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. So, bounded tree-length graphs is a larger class than bounded chordality graphs.

For several problems involving distance computation, like the design of approximate distance labeling schemes [19] or of near-optimal routing schemes [13], 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 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	$(1,2\delta)$	$O(\delta n \log n)$[5]
δ	$(1,4\delta)$	$O(\delta n)$
δ	$(\delta,0)$	$\Omega(n^{1+\epsilon})$, $\epsilon \geqslant 1/\lceil \delta/2 \rceil$
		for[6] $\delta = 1,2,3,4,5,6,9,10$

Thus our first result provides an additive 2-spanner with $O(n \log n)$ edges for chordal graphs (for $\delta = 1$), improving [9] and also implying [22].

In this paper, we also compare our algorithm to the Chepoi-Dragan-Yan's algorithm (CDY) used successfully for k-chordal graphs [9]. For small chordality k, our algorithm produces an additive $2k$-spanner (or $(2k-2)$-spanner of 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 3 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.

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

[5] In a forthcoming paper [14], we show that the number of edges can be reduced to $O(\delta n + n \log n)$.

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

We now recall the definition of *tree-decomposition* introduced by Robertson and Seymour in their work on graph minors [25]. A tree-decomposition of a graph G is a tree T whose vertices, 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 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, over all tree-decompositions T of G, of the length of T.

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.

The paper is organized as follows. In Section 2, we present the first algorithm (Line 1 of the previous table). Section 3 presents the second algorithms (Line 2) and the CDY's algorithm, both based on a *layering-tree* of the graph. We conclude in Section 4 with the lower bound.

2 Additive 2δ-Spanner with $O(\delta n \log n)$ Edges

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

The remaining of this section concerns the proof of Theorem 1. For this purpose we need several ingredients, and of two basic properties.

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 then 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 that the depth of H is at most[7] $\log |V(T)|$.

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. [4]).

From now, G is a (connected) graph with n vertices and of tree-length δ. T denotes a tree-decomposition of G of length δ supposed reduced, and H denotes a hierarchical tree of T. So, the depth of H is at most $\log n$. We denote by $\mathrm{NCA}_H(U, V)$ the *nearest common ancestor* of U and V in H.

Property 1. 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. □

We assume that T is rooted. For every vertex u of G, the *bag* of u, denoted by $\mathcal{B}(u)$, is a bag X of T of minimum depth such $u \in X$. Observe that, once T has been fixed, $\mathcal{B}(u)$ is unique for each u.

Property 2. For every edge $\{u, v\}$ of G, either

1. $\mathcal{B}(u)$ is an ancestor of $\mathcal{B}(v)$ in T and $u \in \mathcal{B}(v)$, or

[7] All the logs are in base two.

2. $\mathcal{B}(v)$ is an ancestor of $\mathcal{B}(u)$ in T and $v \in \mathcal{B}(u)$.

Proof. Let X be any bag of T with a vertex x. Since $x \in X \cup \mathcal{B}(x)$, by Rule 3 of tree-decomposition's definition and by minimality of the depth of $\mathcal{B}(x)$, we have that X is a descendant of $\mathcal{B}(x)$. Now, by Rule 2 there is a bag, say Y, containing u and v. It follows that Y is a descendant of $\mathcal{B}(u)$ and of $\mathcal{B}(v)$, i.e., either $\mathcal{B}(u)$ is an ancestor of $\mathcal{B}(v)$ or the reverse. If $\mathcal{B}(u)$ is an ancestor of $\mathcal{B}(v)$, then $\mathcal{B}(v)$ is on the path from $\mathcal{B}(u)$ to Y in T. By Rule 3, $u \in \mathcal{B}(v)$. Similarly if $\mathcal{B}(v)$ is an ancestor of $\mathcal{B}(u)$, then $v \in \mathcal{B}(u)$. □

We associate with every bag X of T a shortest path spanning tree of G rooted at an arbitrary vertex $r_X \in X$. A vertex u of G is *good* w.r.t. S_X if $\mathcal{B}(u)$ is a descendant of X in H. Otherwise u is *bad* w.r.t. S_X. Clearly a vertex u can be good for some trees and bad for others, but since H is of depth at most $\log n$, u is good for at most $\log n$ trees.

For every tree S_X, we define S'_X, the tree obtained from S_X by removing recursively each bad leaf w.r.t. S_X. The spanner of G claimed by Theorem 1 is simply the graph defined by $G' := \bigcup_{X \in V(T)} S'_X$.

Lemma 1. *G' is an additive 2δ-spanner of G.*

Proof. Let u, v be two vertices of G, and let $Z = \mathrm{NCA}_H(\mathcal{B}(u), \mathcal{B}(v))$. By definition, u, v are good in S_Z, thus u, v are both in S'_Z. By Property 1, Z belongs to the path from $\mathcal{B}(u)$ to $\mathcal{B}(v)$ in T, thus there is a vertex $z \in Z$ such that $d_G(u,v) = d_G(u,z) + d_G(z,v)$. Since S'_Z is a shortest path rooted at r_Z, we have $d_{G'}(u,v) \leqslant d_{S'_Z}(u,v) \leqslant d_G(u,r_Z) + d_G(r_Z,v)$. Note that $d_G(z,r_Z) \leqslant \delta$, and by the triangle inequality, we have:

$$\begin{aligned} d_{G'}(u,v) \;\leqslant\; d_{S'_Z}(u,v) &\leqslant d_G(u,z) + d_G(z,r_Z) + d_G(r_Z,z) + d_G(z,v) \\ &\leqslant d_G(u,z) + d_G(z,v) + 2d_G(z,r_Z) \\ &\leqslant d_G(u,v) + 2\delta \ . \end{aligned}$$

□

The difficult part is to upper bound the number of edges.

Lemma 2. *G' has $O(\delta n \log n)$ edges.*

Proof. For every vertex u of G and every bag X of T, we define $\mathrm{cost}(u,X) = 1$ if $u \in S'_X$, and $\mathrm{cost}(u,X) = 0$ if $u \notin S'_X$. Clearly, the number of vertices of S'_X is $\sum_{u \in V(G)} \mathrm{cost}(u,X)$, and so the total number of edges of G' is

$$|E(G')| \;\leqslant\; \sum_{X \in T} \left(\left(\sum_{u \in V(G)} \mathrm{cost}(u,X) \right) - 1 \right) \;<\; \sum_{X \in T} \sum_{u \in V(G)} \mathrm{cost}(u,X)$$

The problem to estimate this upper bound is that neither $\sum_u \mathrm{cost}(u,X)$, nor $\sum_X \mathrm{cost}(u,X)$ are easy to calculate. To overcome this problem, we assume that each vertex u has a "charge" for each bag X, denoted by $\mathrm{charge}(u,X)$, that can be exchanged with its neighbors under the following condition. Initially, $\mathrm{charge}(u,X) = \mathrm{cost}(u,X)$, for all u and X. Then, while there is a vertex u and a bag X such that:

1. u is bad in S'_X, and
2. $\mathrm{charge}(u,X) > 0$,

$\mathrm{charge}(u,X)$ is decremented and $\mathrm{charge}(v,X)$ incremented, where v is one of the children of u in S'_X (observe that there is no leaf u in S'_X that is bad in S'_X). Such a procedure converges, and we denote by $\mathrm{charge}^*(u,X)$ the final charge of u w.r.t. X.

Clearly, $\sum_X \sum_u \mathrm{cost}(u,X) = \sum_X \sum_u \mathrm{charge}^*(u,X)$. On the other hand we have:

$$\sum_X \sum_u \mathrm{charge}^*(u,X) = \sum_u \sum_X \mathrm{charge}^*(u,X) \;\leqslant\; n \cdot \max_u \left\{ \sum_X \mathrm{charge}^*(u,X) \right\}$$

¿From the above procedure, if u is bad in S'_X then $\text{charge}^*(u,X) = 0$. Say in other words, if u is good in t trees, then the sum $\sum_X \text{charge}^*(u,X)$ has at most t non-null terms. We have seen that a vertex u is good in at most $\log n$ trees. Thus, $\sum_X \text{charge}^*(u,X) \leqslant \max_X \{\text{charge}^*(u,X)\} \cdot \log n$. To summarize,

$$|E(G')| \;<\; \sum_X \sum_u \text{charge}^*(u,X) \;\leqslant\; n \log n \cdot \max_{u,X} \{\text{charge}^*(u,X)\}$$

It remains to show that $\text{charge}^*(u,X) = O(\delta)$, for all u and X. We assume that u is good in S'_X (otherwise we have seen that $\text{charge}^*(u,X) = 0$). Observe that, in S'_X, u can receive the charge of v only if: 1) v is bad, 2) v is an ancestor of u, and 3) there is no good vertex on the path from u to v in S'_X. So, let v be the nearest ancestor of u that is good in S'_X (v is set to the root if such vertex does not exist). The charge received by u (w.r.t. X) is thus at most $d_{S'_X}(u,v)$, and thus its total charge is $\text{charge}^*(u,X) \leqslant 1 + d_{S'_X}(u,v)$.

Let $P = x_0, x_1, \ldots, x_p$ be the path in S'_X from $u = x_0$ to $v = x_p$. P is a shortest path in G and $p = d_G(u,v) \geqslant 1$. Our aim is to show that $p \leqslant 3\delta$. Let Q be the path in T from $\mathcal{B}(u)$ to $\mathcal{B}(v)$.

Claim. For every $0 < i < p$, $\mathcal{B}(x_i) \notin Q$.

Let $Z = \text{NCA}_H(\mathcal{B}(u), \mathcal{B}(v))$. By construction of P, every $x_i \in P \setminus \{x_0, x_p\}$ is bad, i.e., $\mathcal{B}(x_i)$ is not a descendant of Z. By Property 1, all the bags of Q are descendant of Z, so $\mathcal{B}(x_i) \notin Q$, for every $0 < i < p$, as claimed.

Let j be the largest index such that $x_j \in \mathcal{B}(u)$. Obviously, $j \leqslant \delta$ because x_0, x_j belongs to the same bag, $\mathcal{B}(x_0)$. W.l.o.g. $p > j$, since otherwise $p \leqslant \delta$ and we are done. Let $Y = \text{NCA}_T(\mathcal{B}(u), \mathcal{B}(v))$.

Claim. $x_j \in Y$.

If $\mathcal{B}(u) = Y$, then we are done. Assume, $\mathcal{B}(u) \neq Y$. If $j = 0$, then we get a contradiction applying Property 2 on the edge $\{u, x_1\}$. Indeed, either $\mathcal{B}(x_1)$ is an ancestor in T of $\mathcal{B}(u)$ and $x_1 \in \mathcal{B}(u)$, contradicting $j = 0$, or $\mathcal{B}(u)$ is an ancestor of $\mathcal{B}(x_1)$. In this latter case, since the path P intersects Y and $\mathcal{B}(u) \neq Y$, the sub-path $x_1, \ldots, x_p$ must intersect $\mathcal{B}(u)$ in some x_i with $i \geqslant 1$, contradicting again $j = 0$. So we have $0 < j < p$, and thus $\mathcal{B}(x_j) \notin Q$. Now, $\mathcal{B}(x_j)$ must be an ancestor in T of $\mathcal{B}(u)$, because $x_j \in \mathcal{B}(u)$. Since $\mathcal{B}(x_j) \notin Q$, it follows that $\mathcal{B}(x_j)$ is an ancestor in T of all the ancestors of $\mathcal{B}(u)$ in Q. In particular x_j belongs to all the ancestors of $\mathcal{B}(u)$ in Q, so $x_j \in Y$ as claimed.

Let k be the largest index such that $x_k \in Y$. Since $d_G(x_j, x_k) \leqslant \delta$, we have $k \leqslant 2\delta$. W.l.o.g. $k < p$, since otherwise $p \leqslant 2\delta$ and we are done. Since $p > k$, the vertex x_{k+1} exists. Let $Y' = \text{NCA}_T(\mathcal{B}(x_{k+1}), \mathcal{B}(x_p))$.

Claim. $Y' \in Q$, and $x_k \in Y'$.

First observe that for every $t > k$, $\mathcal{B}(x_t)$ is a descendant of Y. Otherwise, since the sub-path $x_t, \ldots, x_p$ must intersect Y, Y would contain some x_i with $i > k$, contradicting the definition of k. In particular, $\mathcal{B}(x_{k+1})$ is a descendant of Y. Since $\mathcal{B}(x_{k+1})$ is descendant of Y', it follows that Y' is a descendant of Y, and thus Y' is on the path in T from Y to $\mathcal{B}(x_p)$. I.e., $Y' \in Q$ as claimed. Let us apply Property 2 on the edge $\{x_k, x_{k+1}\}$. $\mathcal{B}(x_k)$ is ancestor of Y ($x_k \in Y$), and thus $\mathcal{B}(x_k)$ is ancestor of $\mathcal{B}(x_{k+1})$, and thus $x_k \in \mathcal{B}(x_{k+1})$. Thus x_k belongs to all the bags of the path in T from Y to $\mathcal{B}(x_{k+1})$. In particular, $x_k \in Y'$ completing the proof of the claim.

The sub-path $x_{k+1}, \ldots, x_p$ must intersect Y', say in some vertex x_ℓ. Since $\ell > k$, $\mathcal{B}(x_\ell)$ is a descendant of Y. It is also an ancestor of Y'. Thus, $\mathcal{B}(x_\ell)$ belongs to the path in T from Y to Y', i.e., $\mathcal{B}(x_\ell) \in Q$. Since $\mathcal{B}(x_i) \notin Q$ for $i < p$, it turns out that $\ell = p$. We conclude with the fact that $d_G(x_k, x_\ell) \leqslant \delta$, thus $p = \ell \leqslant k + \delta \leqslant 3\delta$.

Therefore, $|E(G')| < (3\delta + 1)n \log n$, completing the proof of Lemma 2. □

Theorem 1 directly follows from lemmas 1 and 2.

3 Additive Spanner with a Linear Number of Edges

We present in this section 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, it follows that this spanner is as well an additive $2k$-spanner (or $(2k-2)$-spanner of odd k) with $O(kn)$ edges if the graph is k-chordal. This latter construction is far from the optimal. Indeed, Chepoi et al [9] have presented an algorithm which computes, for any graph G of chordality k, an additive $(k+1)$-spanner with $O(n)$ edges.

The CDY's algorithm, and our algorithm as well, is based on a *Layering-tree*, technique we present in Section 3.1 with the CDY's algorithm. Our variant is detailed in Section 3.2. In Section 3.3, we compare both algorithms and show that, there are tree-length 3 graph for which the CDY's algorithm returns an additive $\Omega(n^{1/3})$-spanner whereas our algorithm guarantees an additive 12-spanner[8] with $O(n)$ edges.

3.1 Layering-Tree and the CDY's Algorithm

Let G be a graph with a distinguished vertex s. For every integer $i \geqslant 0$, we define $L^i := \{u \in V(G) \mid d_G(s,u) = i\}$. A *layering partition* of G is a partition of each set L^i into $L^i_1, \ldots, L^i_{p_i}$ such that $u, v \in L^i_j$ if and only if there exists a path from u to v using only intermediate vertices w such that $d_G(s,w) \geqslant i$.

Let H be the graph whose vertex set is the collection of all the parts L^i_j. In H, 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. 1 for an example). The vertex s is called the *source* of H.

Lemma 3. [8] *The graph H, called* layering-tree *of G, is a tree and is computable in linear time.*

In each part W of H, a special vertex r_W is distinguished[9] (grayed vertices on Fig. 1(a)).

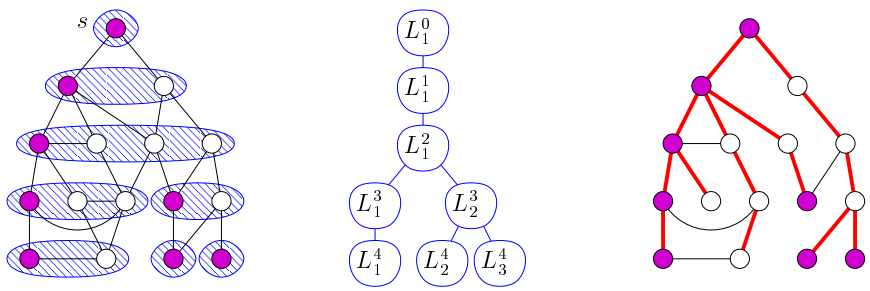

Fig. 1. (a) A 5-chordal graph, (b) a layering-tree of it, and (c) the CDY's spanner.

The CDY's spanner of G is composed of the following edges (see Fig. 1(c)):

1. all the edges of a shortest path spanning tree S of G rooted at s, and
2. for every vertex u of a part W of H, the edge $\{x,y\}$ (if it exists) such that x is the nearest ancestor of u in S with a neighbor y ancestor of r_W.

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

[9] In the original CDY's 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 but more powerful.

3.2 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.*

The construction is also based on the layering-tree H of G, presented in Section 3.1, and on the following result:

Lemma 4. *Let H be a layering-tree of G. For every part W of H, 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 rooted at a bag containing s, the source of H. Let W be a part of H at distance i of s. Let X be the bag of T that is the nearest common ancestor of all the of bags containing a vertex 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 paths 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 $d_G(s,w) \geqslant i$. Q intersects X at a vertex r (see Fig. 2(a)).

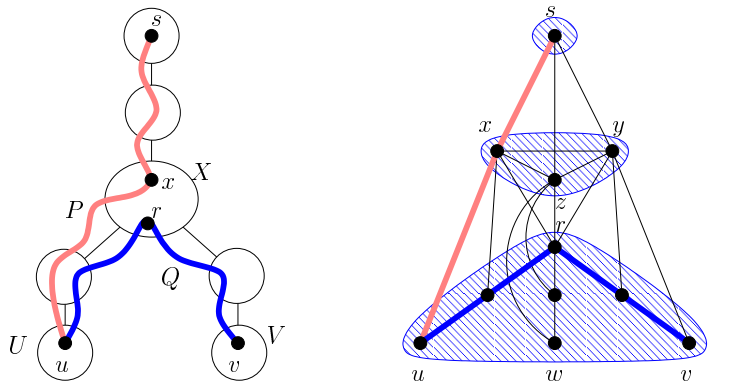

Fig. 2. A part of H is of diameter at most 3δ and of radius at most 2δ.

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. 2(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 $G' := S \cup \bigcup_{W \in V(H)} S_W$, where S is a shortest path tree spanning G, H a layering-tree of G, and S_W a shortest path tree spanning W and rooted at a center of W.

Lemma 5. *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 H containing respectively u and v. Let us show that every path from u to v must intersect the part $W = \mathrm{NCA}_H(A, B)$.

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 H, every path 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.

Let $u', v' \in W$ be the ancestors in S of u and v. We observe that $d_G(u, u') = d_H(A, W)$. Since G' contains a shortest path spanning tree of G rooted at s, it follows that $d_H(A, W) = d_{G'}(u, u')$, 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 4, we have $d_{G'}(u', v') \leqslant 4\delta$. Therefore, we obtain:

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

Lemma 6. *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 H. In this latter case, S_W contains no edges. The parts of H are disjoint, so the number of edges of G' is at most $n - 1 + 2\delta(n - 1) = (2\delta + 1)(n - 1)$. □

3.3 Counter-Example

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

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. 3). 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 7. [15] *The tree-length of any isometric subgraph of G is no more than the tree-length of G.*

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 exists). We check that the graph induced by the vertices of the three paths $P(u)$, $P(v)$, $P(w)$ is a isometric mesh of G_0. This mesh has $t - 1$ rows and 3 columns.

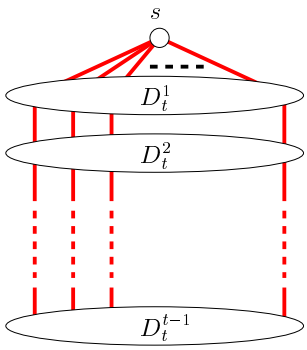

Fig. 3. Counter-example G_0.

Lemma 8. *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 [15] 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 7, 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 9. *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 et 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 exists 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 belongs 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 set $L^0 = \{s\}$, 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}: $R' = \left\{u \in D_t^{t-1} \mid V(P(u)) \cap R \neq \varnothing\right\}$. $|R'| = |R| = t-1$, so R' is not a dominating set of D_t^{t-1} (Lemma 9).

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', excepted those of $P(v)$. So, $d_{G'}(v, r') \geqslant 2(t-1) + 2 = \Omega(n^{1/3})$ whereas $d_{G_0}(v, r') = 2$. G' is an additive $\Omega(n^{1/3})$-spanner, as claimed. □

4 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 exist at least 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 that $d_H(u,v) \geqslant g-1 = d_G(u,v)+g-2$ and thus that 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, a 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, a 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, a 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 combined both optimal tree-decompositions). As shown in [15], the tree-length of a cycle of length $k=3\delta$ is $\lceil k/3 \rceil = \delta$. □

An Erdös Conjecture [17] 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 $k=5$. 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 results on $f(k)$. Complete references can be found in [27].

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

□

Acknowledgements. We would like to thank F. Dragan for fruitful discussions about Lemma 4.

References

1. N. Alon, S. Hoory, and N. Linial, *The Moore bound for irregular graphs*, Graph and Combinatorics, 18 (2002), pp. 53–57.
2. I. Althöfer, G. Das, D. Dobkin, D. Joseph, and J. Soares, *On sparse spanners of weighted graphs*, Discrete & Computational Geometry, 9 (1993), pp. 81–100.
3. S. Baswana and S. Sen, *A simple linear time algorithm for computing a* $(2k-1)$*-spanner of* $O(n^{1+1/k})$ *size in weighted graphs*, in 30^{th} International Colloquium on Automata, Languages and Programming (ICALP), vol. 2719 of LNCS, Springer, July 2003, pp. 384–396.
4. H. L. Bodlaender, *A linear time algorithm for finding tree-decompositions of small treewidth*, SIAM Journal on Computing, 25 (1996), pp. 1305–1317.
5. U. Brandes and D. Handke, *NP-completeness results for minimum planar spanners*, Discrete Mathematics & Theoretical Computer Science, 3 (1998), pp. 1–10.
6. A. Brandstädt, F. F. Dragan, H.-O. Le, and V. B. Le, *Tree spanners on chordal graphs: Complexity and algorithms*, Theoretical Computer Science, 310 (2004), pp. 329–354.
7. L. Cai and D. G. Corneil, *Tree spanners*, SIAM Journal on Discrete Mathematics, 8 (1995), pp. 359–387.
8. V. D. Chepoi and F. F. Dragan, *Distance approximating trees in graphs*, in 6^{th} International Conference on Graph Theory (ICGT), Elsevier, ed., Electronical Notes in Discrete Mathematics, Aug. 2000.
9. V. D. Chepoi, F. F. Dragan, and C. Yan, *Additive spanners for k-chordal graphs*, in 5^{th} Italian Conference on Algorithms and Complexity (CIAC), vol. 2653 of LNCS, Springer, May 2003, pp. 96–107.
10. R. Diestel, *Graph Theory (second edition)*, vol. 173 of Graduate Texts in Mathematics, Springer, Feb. 2000.
11. Y. Dodis and S. Khanna, *Designing networks with bounded pairwise distance*, in 30^{th} Annual ACM Symposium on Theory of Computing (STOC), 1999, pp. 750–759.
12. D. Dor, S. Halperin, and U. Zwick, *All-pairs almost shortest paths*, SIAM Journal on Computing, 29 (2000), pp. 1740–1759.
13. Y. Dourisboure, *Routage compact et longueur arborescente*, PhD thesis, Université Bordeaux 1, Talence, France, Dec. 2003.
14. Y. Dourisboure, F. F. Dragan, C. Gavoille, and C. Yan, *Improved spanners for bounded tree-length graphs*, 2004. In preparation.
15. Y. Dourisboure and C. Gavoille, *Tree-decomposition of graphs with small diameter bags*, in 2^{nd} European Conference on Combinatorics, Graph Theory and Applications (EUROCOMB), Sept. 2003, pp. 100–104.
16. M. Elkin and D. Peleg, $(1+\epsilon,\beta)$*-spanner constructions for general graphs*, in 33^{rd} Annual ACM Symposium on Theory of Computing (STOC), July 2001, pp. 173–182.
17. P. Erdös, *Extremal problems in graph theory*, in Publ. House Cszechoslovak Acad. Sci., Prague, 1964, pp. 29–36.
18. S. P. Fekete and J. Kremer, *Tree spanners in planar graphs*, Discrete Applied Mathematics, 108 (2001), pp. 85–103.
19. C. Gavoille, M. Katz, N. A. Katz, C. Paul, and D. Peleg, *Approximate distance labeling schemes*, in 9^{th} Annual European Symposium on Algorithms (ESA), vol. 2161 of LNCS, Springer, Aug. 2001, pp. 476–488.
20. M. Madanlal, G. Venkatesan, and C. Pandu Rangan, *Tree 3-spanners on interval, permutation and regular bipartite graphs*, Information Processing Letters, 59 (1996), pp. 97–102.
21. D. Peleg, *Distributed Computing: A Locality-Sensitive Approach*, SIAM Monographs on Discrete Mathematics and Applications, 2000.
22. D. Peleg and A. A. Schäffer, *Graph spanners*, Journal of Graph Theory, 13 (1989), pp. 99–116.
23. D. Peleg and J. D. Ullman, *An optimal synchornizer for the hypercube*, SIAM Journal on Computing, 18 (1989), pp. 740–747.
24. D. Peleg and E. Upfal, *A trade-off between space and efficiency for routing tables*, Journal of the ACM, 36 (1989), pp. 510–530.
25. N. Robertson and P. D. Seymour, *Graph minors. II. Algorithmic aspects of tree-width*, Journal of Algorithms, 7 (1986), pp. 309–322.
26. J. Soares, *Graphs spanners: A survey*, Congressus Numerantium, 89 (1992), pp. 225–238.
27. M. Thorup and U. Zwick, *Approximate distance oracles*, in 33^{rd} Annual ACM Symposium on Theory of Computing (STOC), July 2001, pp. 183–192.