Additive Tree $O(\rho \log n)$-Spanners from Tree Breadth ρ

Oliver Bendele[1], Dieter Rautenbach[1]

Institute of Optimization and Operations Research, Ulm University, Germany
{oliver.bendele,dieter.rautenbach}@uni-ulm.de

Abstract

The tree breadth $\text{tb}(G)$ of a connected graph G is the smallest non-negative integer ρ such that G has a tree decomposition whose bags all have radius at most ρ. We show that, given a connected graph G of order n and size m, one can construct in time $O(m \log n)$ an additive tree $O\big(\text{tb}(G) \log n\big)$-spanner of G, that is, a spanning subtree T of G in which $d_T(u, v) \leq d_G(u, v) + O\big(\text{tb}(G) \log n\big)$ for every two vertices u and v of G. This improves earlier results of Dragan and Köhler (Algorithmica 69 (2014) 884-905), who obtained a multiplicative error of the same order, and of Dragan and Abu-Ata (Theoretical Computer Science 547 (2014) 1-17), who achieved the same additive error with a collection of $O(\log n)$ trees.

Keywords : *Combinatorial optimization, graph theory, additive tree spanner, multiplicative tree spanner, tree breadth, tree length.*

1 Introduction

In the present paper we show how to construct in time $O(m \log n)$, for a given connected graph G of order n and size m, a tree spanner that approximates all distances up to some additive error of the form $O(\rho \log n)$, where ρ is the so-called tree breadth of G [8]. Our result improves a result of Dragan and Köhler [8] who show that one can construct in time $O(m \log n)$ a multiplicative tree $O(\rho \log n)$-spanner for a given graph G as above, that is, we improve their multiplicative error to an additive one of the same order. Our result also improves a result by Dragan and Abu-Ata [6] who show how to efficiently construct $O(\log n)$ collective additive tree $O(\rho \log n)$-spanners for a given graph G as above. Note that they obtain the same additive error bound but require several spanning trees that respect this bound only collectively, more precisely, for every pair of vertices, there is a tree in the collection that satisfies the distance condition for this specific pair. Not restricting the spanners to trees allows better guarantees; Dourisboure, Dragan, Gavoille, and Yan [5], for instance, showed that every graph G as above has an additive $O(\rho)$-spanner with $O(\rho n)$ edges. For more background on additive and multiplicative (collective) (tree) spanners please refer to [2, 5–9, 11] and the references therein.

Before we come to our results in Section 2, we collect some terminology and definitions. We consider finite, simple, and undirected graphs. Let G be a connected graph. The *vertex set*, *edge set*, *order*, and *size* of G are denoted by $V(G)$, $E(G)$, $n(G)$, and $m(G)$, respectively. The *distance* in G between two vertices u and v of G is denoted by $d_G(u, v)$. For a vertex u of G and a set U of vertices of G, the *distance* in G between u and U is

$$d_G(u, U) = \min \{d_G(u, v) : v \in U\},$$

and the *radius* $\text{rad}_G(U)$ of U in G is

$$\min \{ \max \{d_G(u, v) : v \in U\} : u \in V(G)\},$$

that is, it is the smallest radius of a ball around some vertex u of G that contains all of U. Note that the vertex u in the preceding minimum is not required to belong to U, and that all distances are considered within G.

Let H be a subgraph of G. For a non-negative integer k, the subgraph H is *k-additive* if

$$d_H(u,v) \leq d_G(u,v) + k \tag{1}$$

for every two vertices u and v of H. If, additionally, the subgraph H is *spanning*, that is, it has the same vertex set as G, then H is an *additive k-spanner* of G. Furthermore, if, again additionally, the subgraph H is a tree, then H is an *additive tree k-spanner* of G. Replacing the inequality (1) with

$$d_H(u,v) \leq k \cdot d_G(u,v)$$

yields the notions of a *k-multiplicative* subgraph, a *multiplicative k-spanner*, and a *multiplicative tree k-spanner* of G, respectively.

For a tree T, let $B(T)$ be the set of vertices of T of degree at least 3 in T, the so-called *branch* vertices, and let $L(T)$ be the set of leaves of T.

A *tree decomposition* of G is a pair $\left(T, (X_t)_{t\in V(T)}\right)$, where T is a tree and X_t is a set of vertices of G for every vertex t of T such that

- for every vertex u of G, the set $\{t \in V(T) : u \in X_t\}$ induces a non-empty subtree of T, and
- for every edge uv of G, there is some vertex t of T such that u and v both belong to X_t.

The set X_t is usually called the *bag* of t. The maximum radius

$$\max\{\text{rad}_G(X_t) : t \in V(T)\}$$

of a bag of the tree decomposition is the *breadth* of this decomposition, and the *tree breadth* $\text{tb}(G)$ of G [8] is the minimum breadth of a tree decomposition of G. While the tree breadth is an NP-hard parameter [10], one can construct in linear time, for a given connected graph G, a tree decomposition of breadth at most $3\text{tb}(G)$ [1], cf. also [3, 4, 8] involving the related notion of *tree length*.

2 Results

For a tree T, let $\text{pbt}(T)$ be the maximum depth of a perfect binary tree that is a topological minor of T. In some sense $\text{pbt}(T)$ quantifies how much T differs from a path.

Our main result is the following.

Theorem 1 *Given a connected graph G of size m and a tree decomposition $\left(T, (X_t)_{t\in V(T)}\right)$ of G of breadth ρ, one can construct in time $O(m\cdot\text{pbt}(T))$ an additive tree $8\rho(2\text{pbt}(T)+1)$-spanner of G.*

Some immediate consequences of Theorem 1 are the following.

Corollary 1 *Given a connected graph G of order n and size m, one can construct in time $O(m \log n)$ an additive tree $O(\text{tb}(G) \log n)$-spanner of G.*

Proof : As observed towards the end of the introduction, given G, one can construct in linear time a tree decomposition $\left(T, (X_t)_{t\in V(T)}\right)$ of G of breadth at most $3\text{tb}(G)$. Possibly by contracting edges st of T with $X_s \subseteq X_t$, we may assume that $n(T) \leq n$. Since a perfect binary tree of depth b has $2^{b+1} - 1$ vertices, it follows that $2^{\text{pbt}(T)+1} - 1 \leq n(T) \leq n$, and, hence,

$$\text{pbt}(T) \leq \log_2(n+1) - 1.$$

Applying Theorem 1 allows to construct in time $O(m \cdot \text{pbt}(T)) = O(m \log n)$ an additive tree $24\text{tb}(G)(2\log_2(n+1) - 1)$-spanner of G. □

Corollary 2 *Given a connected graph G of order n and size m, and a multiplicative tree k-spanner T of G, one can construct in time $O(mn)$ an additive tree $O(k \log n)$-spanner of G.*

Proof : For every vertex u of G, let X_u be the set containing all vertices v of G with $d_T(u,v) \leq \left\lceil \frac{k}{2} \right\rceil$. Since T is a multiplicative tree k-spanner, it follows easily that $\left(T, (X_t)_{t \in V(T)}\right)$ is a tree decomposition of G of breadth at most $\left\lceil \frac{k}{2} \right\rceil$, cf. also [8]. Note that $(X_t)_{t \in V(T)}$ can be determined by n breadth first searches, each of which requires $O(m)$ time. Applying Theorem 1 allows to construct in time $O(m \cdot \text{pbt}(T)) = O(m \log n)$ an additive tree $O(k \log n)$-spanner of G. $\square$

Note that if the tree T in Theorem 1 is a path, then we obtain an additive tree $O(\rho)$-spanner. Kratsch et al. [11] constructed a sequence of outerplanar chordal graphs $G_1, G_2, \ldots$, which limit the extend to which Theorem 1 can be improved. The graph G_1 is a triangle, and, for every positive integer k, the graph G_{k+1} arises from G_k by adding, for every edge uv of G_k that contains a vertex of degree 2 in G_k, a new vertex w that is adjacent to u and v. It is easy to see $n(G_k) = 3 \cdot 2^{k-1}$ and that $\text{tb}(G_k) = 1$ for every positive integer k, in particular, we have $k - 1 = \log_2\left(\frac{n(G_k)}{3}\right)$. Now, Kratsch et al. showed that G_k admits no additive tree $(k-1)$-spanner, that is, the graph G_k admits no additive tree $\text{tb}(G_k) \log_2\left(\frac{n(G_k)}{3}\right)$-spanner.

Our proof of Theorem 1 relies on four lemmas. The first is a simple consequence of elementary properties of breadth first search

Lemma 2 *Given a connected graph G of size m, a subtree S of G, and a set U of vertices of G, one can construct in time $O(m)$ a subtree S' of G containing S as well as all vertices from U such that*

(i) $d_{S'}(u, V(S)) = d_G(u, V(S))$ for every vertex u in U, and

(ii) $L(S') \subseteq L(S) \cup U$.

The following lemma was inspired by Lemma 2.2 in [11]. It will be useful to complete the construction of our additive tree spanner starting from a suitable subtree.

Lemma 3 *Given a connected graph G of size m and a ρ-additive subtree S of G such that $d_G(u, V(S)) \leq \rho'$ for every vertex u of G, one can construct in time $O(m)$ an additive tree $(\rho + 4\rho')$-spanner of G.*

Proof : Let S' be the spanning tree of G obtained by applying Lemma 2 to G, S, and $V(G) \setminus V(S)$ as the set U. We claim that S' has the desired properties. Therefore, let u and v be any two vertices of G. Let u' be the vertex of S closest to u within S', and define v' analogously. Clearly, we have that $d_{S'}(u,u') = d_G(u,u') \leq \rho'$, $d_{S'}(v,v') = d_G(v,v') \leq \rho'$, and $d_{S'}(u',v') = d_S(u',v') \leq d_G(u',v') + \rho$. By several applications of the triangle inequality, we obtain

$$\begin{aligned} d_{S'}(u,v) &= d_{S'}(u,u') + d_S(u',v') + d_{S'}(v',v) \\ &\leq \rho' + d_G(u',v') + \rho + \rho' \\ &\leq d_G(u',u) + d_G(u,v) + d_G(v,v') + \rho + 2\rho' \\ &\leq d_G(u,v) + \rho + 4\rho', \end{aligned}$$

which completes the proof. $\square$

Our next lemma states that $\text{pbt}(T)$ can easily be determined for a given tree T, by constructing a suitable finite sequence

$$T_0 \supset T_1 \supset T_2 \supset \ldots \supset T_{\text{d}(T)} \quad (2)$$

of nested trees. The construction of this sequence is also important for the proof of our main technical lemma, cf. Lemma 5 below. The sequence starts with T_0 equal to T. Now, suppose that T_i has been defined for some non-negative integer i. If $B(T_i)$ is not empty, then let T_{i+1} be the minimal subtree of T_i that contains all vertices from $B(T_i)$, and continue the construction. Note that in this case

$$B(T_i) = B(T_{i+1}) \cup L(T_{i+1}).$$

Otherwise, if $B(T_i)$ is empty, then T_i is a path of some length ℓ. If $\ell \geq 3$, then let T_{i+1} be the tree containing exactly one internal vertex of T_i as its only vertex, and let $\mathrm{d}(T) = i+1$. Finally, if $\ell \leq 2$, then let $\mathrm{d}(T) = i$. Once $\mathrm{d}(T)$ has been defined, the construction of the sequence (2) terminates.

Lemma 4 $\mathrm{pbt}(T) = \mathrm{d}(T)$ *for every tree* T.

The following is our core technical lemma.

Lemma 5 *Given a connected graph* G *of size* m *and a tree decomposition* $\left(T, (X_t)_{t \in V(T)}\right)$ *of* G *of breadth* ρ, *one can construct in time* $O(m \cdot \mathrm{d}(T))$ *a* $16\rho \cdot \mathrm{d}(T)$-*additive subtree* S *of* G *intersecting each bag of the given tree-decomposition.*

Theorem 1 now follows immediately by combining Lemma 5 with Lemma 3, choosing ρ' equal to 2ρ for the latter. Note that, since the tree S produced by Lemma 5 intersects every bag of the tree decomposition, we have $d_G(u, V(S)) \leq 2\rho$ for every vertex u of G.

References

[1] M. Abu-Ata and F.F. Dragan, Metric tree-like structures in real-world networks: an empirical study, Networks 67 (2016) 49-68.

[2] L. Cai and D.G. Corneil, Tree Spanners, SIAM Journal on Discrete Mathematics 8 (1995) 359-387.

[3] V.D. Chepoi and F.F. Dragan, A note on distance approximating trees in graphs, European Journal of Combinatorics 21 (2000) 761-768.

[4] Y. Dourisboure and C. Gavoille, Tree-decompositions with bags of small diameter, Discrete Mathematics 307 (2007) 2008-2029.

[5] Y. Dourisboure, F.F. Dragan, C. Gavoille, and C. Yan, Spanners for bounded tree-length graphs, Theoretical Computer Science 383 (2007) 34-44.

[6] F.F. Dragan and M. Abu-Ata, Collective additive tree spanners of bounded tree-breadth graphs with generalizations and consequences, Theoretical Computer Science 547 (2014) 1-17.

[7] F.F. Dragan, D.G. Corneil, E. Köhler, and Y. Xiang, Collective additive tree spanners for circle graphs and polygonal graphs, Discrete Applied Mathematics 160 (2012) 1717-1729.

[8] F.F. Dragan and E. Köhler, An approximation algorithm for the tree t-spanner problem on unweighted graphs via generalized chordal graphs, Algorithmica 69 (2014) 884-905.

[9] F.F. Dragan, C. Yan, and D.G. Corneil, Collective tree spanners and routing in AT-free related graphs, Journal of Graph Algorithms and Applications 10 (2006) 97-122.

[10] G. Ducoffe, S. Legay, and N. Nisse, On the complexity of computing treebreadth, Algorithmica (2019) https://doi.org/10.1007/s00453-019-00657-7.

[11] D. Kratsch, H.-O. Le, H. Müller, E. Prisner, and D. Wagner, Additive tree spanners, SIAM Journal on Discrete Mathematics 17 (2002) 332-340.