Balancing Degree, Diameter and Weight in Euclidean Spanners

Shay Solomon* and Michael Elkin**

Department of Computer Science, Ben-Gurion University of the Negev,
P.O.B. 653, Beer-Sheva 84105, Israel
{shayso,elkinm}@cs.bgu.ac.il

Abstract. In a seminal STOC'95 paper, Arya et al. [4] devised a construction that for any set S of n points in $\mathbb{R}^d$ and any $\epsilon > 0$, provides a $(1+\epsilon)$-spanner with diameter $O(\log n)$, weight $O(\log^2 n)w(MST(S))$, and constant maximum degree. Another construction of [4] provides a $(1+\epsilon)$-spanner with $O(n)$ edges and diameter $\alpha(n)$, where α stands for the inverse-Ackermann function. Das and Narasimhan [12] devised a construction with constant maximum degree and weight $O(w(MST(S)))$, but whose diameter may be arbitrarily large. In another construction by Arya et al. [4] there is diameter $O(\log n)$ and weight $O(\log n)w(MST(S))$, but it may have arbitrarily large maximum degree. These constructions fail to address situations in which we are prepared to compromise on one of the parameters, but cannot afford it to be arbitrarily large.

In this paper we devise a novel *unified* construction that trades between maximum degree, diameter and weight gracefully. For a positive integer k, our construction provides a $(1+\epsilon)$-spanner with maximum degree $O(k)$, diameter $O(\log_k n + \alpha(k))$, weight $O(k \log_k n \log n)w(MST(S))$, and $O(n)$ edges. For $k = O(1)$ this gives rise to maximum degree $O(1)$, diameter $O(\log n)$ and weight $O(\log^2 n)w(MST(S))$, which is one of the aforementioned results of [4]. For $k = n^{1/\alpha(n)}$ this gives rise to diameter $O(\alpha(n))$, weight $O(n^{1/\alpha(n)}(\log n)\alpha(n))w(MST(S))$ and maximum degree $O(n^{1/\alpha(n)})$. In the corresponding result from [4] the spanner has the same number of edges and diameter, but its weight and degree may be arbitrarily large. Our construction also provides a similar tradeoff in the complementary range of parameters, i.e., when the weight should be smaller than $\log^2 n$, but the diameter is allowed to grow beyond $\log n$.

1 Introduction

Euclidean Spanners. Consider the weighted complete graph $\mathcal{S} = (S, \binom{S}{2})$ induced by a set S of n points in $\mathbb{R}^d, d \geq 2$. The weight of an edge $(x, y) \in \binom{S}{2}$, for a pair of distinct points $x, y \in S$, is defined to be the Euclidean distance $\|x - y\|$ between x and y. Let $G = (S, E)$ be a spanning subgraph of $\mathcal{S}$, with $E \subseteq \binom{S}{2}$,

* This research has been supported by the Clore Fellowship grant No. 81265410.
** This research has been supported by the BSF grant No. 2008430. Both authors are partially supported by the Lynn and William Frankel Center for Computer Science.

M. de Berg and U. Meyer (Eds.): ESA 2010, Part I, LNCS 6346, pp. 48–59, 2010.

and assume that exactly as in $\mathcal{S}$, for any edge $e = (x, y) \in E$, its weight $w(e)$ in G is defined to be $\|x - y\|$. For a parameter $\epsilon > 0$, the spanning subgraph G is called a $(1 + \epsilon)$-*spanner* for the point set S if for every pair $x, y \in S$ of points, the distance $dist_G(x, y)$ between x and y in G is at most $(1 + \epsilon)\|x - y\|$. Euclidean spanners were introduced more than twenty years ago by Chew [11]. Since then they evolved into an important subarea of Computational Geometry [19,3,12,4,13,5,23,15,1,7,14]. (See also the recent book by Narasimhan and Smid on Euclidean spanners [21], and the references therein.) Also, Euclidean spanners have numerous applications in geometric approximation algorithms [23,16,17], geometric distance oracles [16,17], Network Design [18,20] and in other areas.

In many of these applications one is required to construct a $(1 + \epsilon)$-spanner $G = (S, E)$ that satisfies a number of useful properties. First, the spanner should contain $O(n)$ (or nearly $O(n)$) edges. Second, its *weight* $w(G) = \sum_{e\in E} w(e)$ should not be much greater than the weight $w(MST(S))$ of the minimum spanning tree $MST(S)$ of S. Third, its *diameter* $\Lambda = \Lambda(G)$ should be small, i.e., for every pair of points $x, y \in S$ there should exist a path P in G that contains at most Λ edges and has weight $w(P) = \sum_{e\in E(P)} w(e) \leq (1 + \epsilon)\|x - y\|$. Fourth, its *maximum degree* (henceforth, *degree*) $\Delta(G)$ should be small.

In a seminal STOC'95 paper, Arya et al. [4] devised a construction of $(1 + \epsilon)$-spanners with lightness[1] $O(\log^2 n)$, diameter $O(\log n)$ and constant degree. They also devised a construction of $(1 + \epsilon)$-spanners with $O(n)$ (respectively, $O(n \log^* n)$) edges and diameter $O(\alpha(n))$ (resp., at most $O(1)$). However, in the latter construction the resulting spanners may have arbitrarily large (i.e., at least $\Omega(n)$) lightness and degree. There are also a few other known constructions of $(1+\epsilon)$-spanners. Das and Narasimhan [12] devised a construction with constant degree and lightness, but the diameter may be arbitrarily large. (See also [15] for a faster implementation of a spanner construction with constant degree and lightness.) There is also another construction by Arya et al. [4] that guarantees that both the diameter and the lightness are $O(\log n)$, but the degree may be arbitrarily large. While these constructions address some important practical scenarios, they certainly do not address all of them. In particular, they fail to address situations in which we are prepared to compromise on one of the parameters, but cannot afford this parameter to be arbitrarily large.

In this paper we devise a novel *unified* construction that trades between degree, diameter and weight gracefully. For a positive integer k, our construction provides a $(1+\epsilon)$-spanner with degree $O(k)$, diameter $O(\log_k n+\alpha(k))$, lightness $O(k \log_k n \log n)$, and $O(n)$ edges. Also, we can improve the bound on the diameter from $O(\log_k n+\alpha(k))$ to $O(\log_k n)$, at the expense of increasing the number of edges from $O(n)$ to $O(n \log^* n)$. Note that for $k = O(1)$ our tradeoff gives rise to degree $O(1)$, diameter $O(\log n)$ and lightness $O(\log^2 n)$, which is one of the aforementioned results of [4]. Also, for $k = n^{1/\alpha(n)}$ it gives rise to a spanner with degree $O(n^{1/\alpha(n)})$, diameter $O(\alpha(n))$ and lightness $O(n^{1/\alpha(n)}(\log n)\alpha(n))$.

[1] For convenience, we will henceforth refer to the normalized notion of weight $\Psi(G) = \frac{w(G)}{w(MST(S))}$, which we call *lightness*.

In the corresponding result from [4] the spanner has the same number of edges and diameter, but its lightness and degree may be arbitrarily large.

In addition, we can achieve lightness $o(\log^2 n)$ at the expense of increasing the diameter. Specifically, for a parameter k the second variant of our construction provides a $(1+\epsilon)$-spanner with degree $O(1)$, diameter $O(k \log_k n)$, and lightness $O(\log_k n \log n)$. For example, for $k = \log^\delta n$, for an arbitrarily small constant $\delta > 0$, we get a $(1+\epsilon)$-spanner with degree $O(1)$, diameter $O(\log^{1+\delta} n)$ and lightness $O(\frac{\log^2 n}{\log\log n})$.

Our unified construction can be implemented in $O(n \log n)$ time in the algebraic computation-tree model[2]. This matches the state-of-the-art running time of the aforementioned constructions [4,15].

Note that in any construction of spanners with degree $O(k)$, the diameter is $\Omega(\log_k n)$. Also, Chan and Gupta [7] showed that any $(1+\epsilon)$-spanner with $O(n)$ edges must have diameter $\Omega(\alpha(n))$. Consequently, our upper bound of $O(\log_k n + \alpha(k))$ on the diameter is tight under the constraints that the degree is $O(k)$ and the number of edges is $O(n)$. If we allow $O(n \log^* n)$ edges in the spanner, than our bound on the diameter is reduced to $O(\log_k n)$, which is again tight under the constraint that the degree is $O(k)$.

In addition, Dinitz et al. [14] showed that for any construction of spanners, if the diameter is at most $O(\log_k n)$, then the lightness is at least $\Omega(k \log_k n)$ and vice versa, if the lightness is at most $O(\log_k n)$, the diameter is at least $\Omega(k \log_k n)$. This lower bound implies that the bound on lightness in both our tradeoffs cannot possibly be improved by more than a factor of $\log n$. The same slack of $\log n$ is present in the result of [4] that guarantees lightness $O(\log^2 n)$, diameter $O(\log n)$ and constant degree.

Euclidean Spanners for Random Point Sets. For random point sets in the d-dimensional unit cube (henceforth, unit cube), we "shave" a factor of $\log n$ from the lightness bound in both our tradeoffs, and show that the first (respectively, second) variant of our construction achieves maximum degree $O(k)$ (resp., $O(1)$), diameter $O(\log_k n + \alpha(k))$ (resp., $O(k \log_k n)$) and lightness that is with high probability (henceforth, w.h.p.) $O(k \log_k n)$ (resp., $O(\log_k n)$). Note that for $k = O(1)$ both these tradeoffs give rise to degree $O(1)$, diameter $O(\log n)$ and lightness (w.h.p.) $O(\log n)$. In addition to these tradeoffs, we can get a $(1+\epsilon)$-spanner with diameter $O(\log n)$ and lightness (w.h.p.) $O(1)$.

Spanners for Doubling Metrics. Spanners for doubling metrics[3] have received much attention in recent years (see, e.g., [8,7]). In particular, Chan et al. [8] showed that for any doubling metric (X, δ) there exists a $(1+\epsilon)$-spanner with constant maximum degree. In addition, Chan and Gupta [7] devised a construction of $(1+\epsilon)$-spanners for doubling metrics that achieves the optimal tradeoff

[2] See, e.g., Chapter 3 of [21] for the definition of the algebraic computation-tree model.
[3] The *doubling dimension* of a metric (X, δ) is the smallest value ζ such that every ball B in the metric can be covered by at most 2^ζ balls of half the radius of B. The metric (X, δ) is called *doubling* if its doubling dimension ζ is constant.

between the number of edges and the diameter. We present a single construction of $O(1)$-spanners for doubling metrics that achieves the optimal tradeoff between the degree, the diameter and the number of edges in the entire range of parameters. Specifically, for a parameter k, our construction provides an $O(1)$-spanner with maximum degree $O(k)$, diameter $O(\log_k n + \alpha(k))$ and $O(n)$ edges. Also, we can improve the bound on the diameter from $O(\log_k n + \alpha(k))$ to $O(\log_k n)$, at the expense of increasing the number of edges from $O(n)$ to $O(n \log^* n)$. More generally, we can achieve the same optimal tradeoff between the number of edges and the diameter as the spanners of [7] do, while also having the optimal maximum degree. The drawback is, however, that the stretch of our spanners is $O(1)$ rather than $1 + \epsilon$.

Spanners for Tree Metrics. We denote by ϑ_n the metric induced by n points $v_1, v_2, \ldots, v_n$ lying on the x-axis with coordinates $1, 2, \ldots, n$, respectively. In a classical STOC'82 paper [26], Yao showed that there exists a 1-spanner[4] $G = (V, E)$ for ϑ_n with $O(n)$ edges and diameter $O(\alpha(n))$, and that this is tight. Chazelle [9] extended the result of [26] to arbitrary tree metrics. (See also [2,6,25].) The problem is also closely related to the well-studied problem of computing partial-sums [24,26,10,22].

In all constructions [26,9,2,6,25] of 1-spanners for tree metrics, the degree and lightness of the resulting spanner may be arbitrarily large. Moreover, the constraint that the diameter is $O(\alpha(n))$ implies that the degree must be $n^{\Omega(1/\alpha(n))}$. A similar lower bound on lightness follows from the result of [14].

En route to our tradeoffs for Euclidean spanners, we have extended the results of [26,9,2,6,25] and devised a construction that achieves the *optimal* (up to constant factors) *tradeoff between all involved parameters*. Specifically, consider an n-vertex tree T of degree $\Delta(T)$, and let k be a positive integer. Our construction provides a 1-spanner for the metric M_T induced by T with $O(n)$ edges, degree $O(k + \Delta(T))$, diameter $O(\log_k n + \alpha(k))$, and lightness $O(k \log_k n)$. We can also get a spanner with $O(n \log^* n)$ edges, diameter $O(\log_k n)$, and the same degree and lightness as above. For the complementary range of diameter, another variant of our construction provides a 1-spanner with $O(n)$ edges, degree $O(\Delta(T))$, diameter $O(k \log_k n)$ and lightness $O(\log_k n)$. As was mentioned above, both these tradeoffs are optimal up to constant factors.

We show that this general tradeoff between various parameters of 1-spanners for tree metrics is useful for deriving new results (and improving existing results) in the context of Euclidean spanners and spanners for doubling metrics. We anticipate that this tradeoff will be found useful in the context of partial sums problems, and for other applications.

Structure of the Paper. In Sect. 2 we describe our construction of 1-spanners for tree metrics. Therein we start (Sect. 2.1) with outlining our basic scheme. We proceed (Sect. 2.2) with describing our 1-dimensional construction. In Sect.

[4] The graph G is said to be a *1-spanner* of ϑ_n if for every pair of distinct vertices $v_i, v_j \in V$, the distance between them in G is equal to their distance $\|i - j\|$ in ϑ_n. Yao stated this problem in terms of partial sums.

2.3 we extend this construction to general tree metrics. In Sect. 3 we derive our results for Euclidean spanners and spanners for doubling metrics. Due to space constraints, we leave all issues of running time as well as some proofs out of this extended abstract.

Preliminaries. Let G be a spanning subgraph of a metric space $M = (V, dist)$. The *stretch* between two vertices $u, v \in V$ is defined as $\frac{dist_G(u,v)}{dist(u,v)}$. We say that G is a *t-spanner* for M if the maximum stretch taken over all pairs of points in V is at most t. Let T be an arbitrary tree, and denote by $V(T)$ the vertex set of T. For any two vertices u, v in T, their (weighted) distance in T is denoted by $dist_T(u, v)$. The tree metric M_T induced by T is defined as $M_T = (V(T), dist_T)$. The *size* of T, denoted $|T|$, is the number of vertices in T. Finally, for a positive integer n, we denote the set $\{1, 2, \ldots, n\}$ by $[n]$.

2 1-Spanners for Tree Metrics

2.1 The Basic Scheme

Consider an arbitrary n-vertex (weighted) rooted tree (T, rt), and let M_T be the tree metric induced by T. Clearly, T is both a 1-spanner and an MST of M_T, but its diameter may be huge. We would like to reduce the diameter of this 1-spanner by adding to it some edges. On the other hand, the number of edges of the resulting spanner should still be linear in n. Moreover, the lightness and the maximum degree of the resulting spanner should also be reasonably small.

Let H be a spanning subgraph of M_T. The *monotone distance* between any two points u and v in H is defined as the minimum number of hops in a 1-spanner path in H connecting them. Two points in M_T are called *comparable* if one is an ancestor of the other in the underlying tree T. The *monotone diameter* (respectively, *comparable monotone diameter*) of H, denoted $\Lambda(H)$ (resp., $\bar{\Lambda}(H)$), is defined as the maximum monotone distance in H between any two points (resp., any two comparable points) in M_T. Observe that if any two *comparable* points are connected via a 1-spanner path that consists of at most h hops, then any two *arbitrary* points are connected via a 1-spanner path that consists of at most $2h$ hops. Consequently, $\bar{\Lambda}(H) \le \Lambda(H) \le 2\bar{\Lambda}(H)$. We henceforth restrict attention to comparable monotone diameter in the sequel.

Let k be a fixed parameter. The first ingredient of the algorithm is to select a set of $O(k)$ *cut-vertices* whose removal from T partitions it into a collection of subtrees of size $O(n/k)$ each. As will become clear in the sequel, we also require this set to satisfy several additional properties. Having selected the cut-vertices, the next step of the algorithm is to connect the cut-vertices via $O(k)$ edges, so that the monotone distance between any pair of comparable cut-vertices will be small. (This phase does not involve a recursive call of the algorithm.) Finally, the algorithm calls itself recursively for each of the subtrees.

We insert all edges of the original tree T into our final spanner H. These edges connect between cut-vertices and subtrees in the spanner. We remark that the spanner contains no other edges that connect between cut-vertices and subtrees, or between different subtrees.

2.2 1-Dimensional Spaces

In this section we devise an optimal construction of 1-spanners for ϑ_n. (See Sect. 1 for its definition.) Our argument extends easily to any 1-dimensional space.

Denote by P_n the path $(v_1, v_2), (v_2, v_3), \ldots, (v_{n-1}, v_n)$ that induces the metric ϑ_n. We remark that the edges of P_n belong to all spanners that we construct.

Selecting the Cut-Vertices. The task of selecting the cut-vertices in the 1-dimensional case is trivial. (We assume for simplicity that n is an integer power of k.) In addition to the two endpoints v_1 and v_n of the path, we select the $k-1$ points $r_1, r_2, \ldots, r_{k-1}$ to be cut-vertices, where for each $i \in [k-1]$, $r_i = v_{i(n/k)}$. Indeed, by removing the $k+1$ cut-vertices $r_0 = v_1, r_1, \ldots, r_{k-1}, r_k = v_n$ from the path (along with their incident edges), we are left with k intervals $I_1, I_2, \ldots, I_k$ of length at most n/k each. The two endpoints v_1 and v_n of the path are called the *sentinels*, and they play a special role in the construction.

1-Spanners with Low Diameter. In this section we devise a construction $H_k(n)$ of 1-spanners for ϑ_n with comparable monotone diameter $\bar{\Lambda}(n) = \bar{\Lambda}(H_k(n))$ in the range $\Omega(\alpha(n)) = \bar{\Lambda}(n) = O(\log n)$.

First, the algorithm connects the $k+1$ cut-vertices $r_0 = v_1, r_1, \ldots, r_{k-1}, r_k = v_n$ via one of the aforementioned constructions of 1-spanners from [26,9,2,6,25] (henceforth, *list-spanner*). In other words, $O(k)$ edges are added between cut-vertices to guarantee that the monotone distance between any two cut-vertices will be at most $O(\alpha(k))$. Then the algorithm connects each of the two sentinels to all other k cut-vertices. Finally, the algorithm calls itself recursively for each of the intervals $I_1, I_2, \ldots, I_k$. At the bottom level of the recursion, i.e., when $n \le k$, the algorithm uses the list-spanner to connect all points, and also connects each of the two sentinels v_1 and v_n to all the other $n-1$ points.

Denote by $E(n)$ the number of edges in $H_k(n)$, excluding edges of P_n. Clearly, $E(n)$ satisfies the recurrence $E(n) \le O(k) + kE(n/k)$, with the base condition $E(n) = O(n)$, for $n \le k$, yielding $E(n) = O(n)$. Denote by $\Delta(n)$ the maximum degree of a vertex in $H_k(n)$, excluding edges of P_n. Clearly, $\Delta(n)$ satisfies the recurrence $\Delta(n) \le \max\{k, \Delta(n/k)\}$, with the base condition $\Delta(n) \le n-1$, for $n \le k$, yielding $\Delta(n) \le k$. Including edges of P_n, the number of edges increases by $n-1$ units, and the maximum degree increases by at most two units.

To bound the weight $w(n) = w(H_k(n))$ of $H_k(n)$, first note that at most $O(k)$ edges are added between cut-vertices. Each of these edges has weight at most $n-1$. The total weight of all edges within an interval I_i is at most $w(n/k)$. Observe also that $w(P_n) = n-1$. Hence $w(n)$ satisfies the recurrence $w(n) \le O(nk) + kw(n/k)$, with the base condition $w(n) = O(n^2)$, for $n \le k$. It follows that $w(n) = O(nk \log_k n) = O(k \log_k n) w(MST(\vartheta_n))$.

Next, we show that the comparable monotone diameter $\bar{\Lambda}(n)$ of $H_k(n)$ is $O(\log_k n + \alpha(k))$. The *monotone radius* $R(n)$ of $H_k(n)$ is defined as the maximum monotone distance in $H_k(n)$ between one of the sentinels (either v_1 or v_n) and some other point in ϑ_n. Let v_j be a point in ϑ_n, and let i be the index such that $i(n/k) \le j < (i+1)(n/k)$. Then the 1-spanner path Π in $H_k(n)$ connecting the sentinel v_1 and the point v_j will start with the two edges $(v_1, v_{i(n/k)})$,

$(v_{i(n/k)}, v_{i(n/k)+1})$. The point $v_{i(n/k)+1}$ is a sentinel of the ith interval I_i, and so the path Π will continue recursively from $v_{i(n/k)+1}$ to v_j. Hence, the monotone radius $R(n)$ satisfies the recurrence $R(n) \leq 2 + R(n/k)$, with the base condition $R(n) = 1$, for $n \leq k$, yielding $R(n) = O(\log_k n)$. Finally, it is easy to verify that $\bar{\Lambda}(n)$ satisfies the recurrence $\bar{\Lambda}(n) \leq \max\{\bar{\Lambda}(n/k), O(\alpha(k)) + 2R(n/k)\}$, with the base condition $\bar{\Lambda}(n) = O(\alpha(n))$, for $n \leq k$. Hence $\bar{\Lambda}(n) = O(\log_k n + \alpha(k))$.

Theorem 1. *For any n-point 1-dimensional space and a parameter k, there exists a 1-spanner with $O(n)$ edges, maximum degree at most $k+2$, diameter $O(\log_k n + \alpha(k))$ and lightness $O(k \log_k n)$.*

1-Spanners with High Diameter. In this section we devise a construction $H'_k(n)$ of 1-spanners for ϑ_n with comparable monotone diameter $\bar{\Lambda}'(n) = \bar{\Lambda}(H'_k(n))$ in the range $\bar{\Lambda}'(n) = \Omega(\log n)$.

The algorithm connects the $k+1$ cut-vertices $r_0 = v_1, r_1, \ldots, r_{k-1}, r_k = v_n$ via a path of length k, i.e., it adds the edges $(r_0, r_1), (r_1, r_2), \ldots, (r_{k-1}, r_k)$ into the spanner. In addition, it calls itself recursively for each of the intervals $I_1, I_2, \ldots, I_k$. At the bottom level of the recursion, i.e., when $n \leq k$, the algorithm adds no additional edges to the spanner.

We denote by $\Delta'(n)$ the maximum degree of a vertex in $H'_k(n)$, excluding edges of P_n. Clearly, $\Delta'(n)$ satisfies the recurrence $\Delta'(n) \leq \max\{2, \Delta'(n/k)\}$, with the base condition $\Delta'(n) = 0$, for $n \leq k$, yielding $\Delta'(n) \leq 2$. Including edges of P_n, the maximum degree increases by at most two units, and so $\Delta(H'_k(n)) \leq 4$. Consequently, the number of edges in $H'_k(n)$ is no greater than $2n$. To bound the weight $w'(n) = w(H'_k(n))$ of $H'_k(n)$, first note that the path connecting all $k+1$ cut-vertices has weight $n-1$. Observe also that $w(P_n) = n-1$. Thus $w'(n)$ satisfies the recurrence $w'(n) \leq 2(n-1) + kw'(n/k)$, with the base condition $w'(n) \leq n-1$, for $n \leq k$, yielding $w'(n) = O(n \log_k n) = O(\log_k n) w(MST(\vartheta_n))$.

Note that the monotone radius $R'(n)$ of $H'_k(n)$ satisfies the recurrence $R'(n) \leq k + R'(n/k)$, with the base condition $R'(n) \leq n-1$, for $n \leq k$. Hence, $R'(n) = O(k \log_k n)$. It is easy to verify that the comparable monotone diameter $\bar{\Lambda}'(n) = \bar{\Lambda}(H'_k(n))$ of $H'_k(n)$ satisfies the recurrence $\bar{\Lambda}'(n) \leq \max\{\bar{\Lambda}'(n/k), k + 2R'(n/k)\}$, with the base condition $\bar{\Lambda}'(n) \leq n-1$, for $n \leq k$, and so $\bar{\Lambda}'(n) = O(k \log_k n)$.

Finally, we remark that the spanner $H'_k(n)$ is a planar graph.

Theorem 2. *For any n-point 1-dimensional space and a parameter k, there exists a 1-spanner with maximum degree 4, diameter $O(k \log_k n)$ and lightness $O(\log_k n)$. Moreover, this 1-spanner is a planar graph.*

2.3 General Tree Metrics

In this section we extend the constructions of Sect. 2.2 to general tree metrics.

Selecting the Cut-Vertices. In this section we present a procedure for selecting, given a tree T, a subset of $O(k)$ vertices whose removal from the tree partitions it into subtrees of size $O(|T|/k)$ each. This subset will also satisfy several additional special properties.

Let (T, rt) be a rooted tree. For an inner vertex v in T with $ch(v)$ children, we denote its children by $c_1(v), c_2(v), \ldots, c_{ch(v)}(v)$. Suppose without loss of generality that the size of the subtree $T_{c_1(v)}$ of v is no smaller than the size of any other subtree of v, i.e., $|T_{c_1(v)}| \geq |T_{c_2(v)}|, |T_{c_3(v)}|, \ldots, |T_{c_{ch(v)}(v)}|$. We say that the vertex $c_1(v)$ is the *left-most* child of v. Also, an edge in T is called *left-most* if it connects a vertex v in T and its left-most child $c_1(v)$. We denote by $P(v) = (v, c_1(v), \ldots, l(v))$ the path of left-most edges leading down from v to the left-most vertex $l(v)$ in the subtree T_v of T rooted at v. A vertex v in T is called *d-balanced*, for $d \geq 1$, or simply *balanced* if d is clear from the context, if $|T_{c_1(v)}| \leq |T| - d$. The first balanced vertex along $P(v)$ is denoted by $B(v)$.

Next, we present the Procedure CV that accepts as input a rooted tree (T, rt) and a parameter d, and returns as output a subset of $V(T)$. If $|T| < 2d$, the procedure returns the empty set $\emptyset$. Otherwise, for each child $c_i(b)$ of the first balanced vertex $b = B(rt)$ along $P(rt)$, $i \in [ch(b)]$, the procedure recursively constructs the subset $C_i = CV((T_{c_i(b)}, c_i(b)), d)$, and then returns $\bigcup_{i=1}^{ch(b)} C_i \cup \{b\}$.

Let (T, rt) be an n-vertex rooted tree, and let d be a fixed parameter. Next, we analyze the properties of the set $C = CV((T, rt), d)$ of *cut-vertices*.

For a tree τ, the root $rt(\tau)$ of τ and its left-most vertex $l(\tau)$ are called the *sentinels* of τ. Similarly to the 1-dimensional case, we add the two sentinels $rt(T)$ and $l(T)$ of the original tree T to the set C of cut-vertices. From now on we refer to the appended set $\tilde{C} = C \cup \{rt(T), l(T)\}$ as the set of *cut-vertices*. Observe that the subset $\tilde{C}$ induces a tree $\tilde{\mathcal{Q}} = \mathcal{Q}(T, \tilde{C})$ over $\tilde{C}$ in the natural way: a vertex $v \in \tilde{C}$ is defined to be a child of its closest ancestor in T that belongs to $\tilde{C}$. We denote by $T \setminus \tilde{C}$ the forest obtained from T by removing all vertices in $\tilde{C}$, along with the edges that are incident to them.

Proposition 1. *1) For $n \geq 2d$, $|\tilde{C}| \leq (n/d) + 1$. 2) The size of any subtree in $T \setminus \tilde{C}$ is smaller than $2d$. 3) $\tilde{\mathcal{Q}} = \mathcal{Q}(T, \tilde{C})$ is a spanning tree of $\tilde{C}$ rooted at $rt(T)$, with $\Delta(\tilde{\mathcal{Q}}) \leq \Delta(T)$. 4) For any subtree T' in $T \setminus \tilde{C}$, only the two sentinels of T' are incident to a vertex in $\tilde{C}$. Also, $rt(T')$ is incident only to its parent in T and $l(T')$ is incident only to its left-most child in T, unless it is a leaf in T.*

Intuitively, part (4) of this proposition shows that the Procedure CV "slices" the tree in a "path-like" fashion analogous to the partition of ϑ_n into intervals.

1-Spanners with Low Diameter. Consider an n-vertex (weighted) tree T, and let M_T be the tree metric induced by T. In this section we devise a construction $\mathcal{H}_k(n)$ of 1-spanners for M_T with comparable monotone diameter $\bar{\Lambda}(n) = \bar{\Lambda}(\mathcal{H}_k(n))$ in the range $\Omega(\alpha(n)) = \bar{\Lambda}(n) = O(\log n)$. Both in this construction and in the one with high diameter presented in the sequel, all edges of the original tree T are added to the spanner.

Let k be a fixed parameter such that $4 \leq k \leq n/2 - 1$, and set $d = n/k$. (We have $n \geq 2k + 2$ and $d > 2$.) To select the set $\tilde{C}$ of cut-vertices, we invoke the procedure CV on the input (T, rt) and d. Set $C = CV((T, rt), d)$ and $\tilde{C} = C \cup \{rt(T), l(T)\}$. Denote the subtrees in the forest $T \setminus \tilde{C}$ by $T_1, T_2, \ldots, T_p$. By Proposition 1, $|\tilde{C}| \leq k + 1$, and each subtree T_i in $T \setminus \tilde{C}$ has size less than $2n/k$.

To connect the set $\tilde{C}$ of cut-vertices, the algorithm first constructs the tree $\tilde{\mathcal{Q}} = \mathcal{Q}(T, \tilde{C})$. Note that $\tilde{\mathcal{Q}}$ inherits the tree structure of T, i.e., for any two points u and v in $\tilde{C}$, u is an ancestor of v in $\tilde{\mathcal{Q}}$ iff it is its ancestor in T. Consequently, any 1-spanner path in $\tilde{\mathcal{Q}}$ between two arbitrary comparable[5] points is also a 1-spanner path for them in the original tree T. The algorithm proceeds by building a 1-spanner for $\tilde{\mathcal{Q}}$ via one of the aforementioned generalized constructions from [9,2,25] (henceforth, *tree-spanner*). In other words, $O(k)$ edges between cut-vertices are added to the spanner $\mathcal{H}_k(n)$ to guarantee that the monotone distance in the spanner between any two comparable cut-vertices is $O(\alpha(k))$. Then the algorithm adds to the spanner $\mathcal{H}_k(n)$ edges that connect each of the two sentinels to all other cut-vertices. (In fact, the leaf $l(T)$ needs not be connected to all cut-vertices, but rather only to those which are its ancestors in T.) Finally, the algorithm calls itself recursively for each of the subtrees $T_1, T_2, \ldots, T_p$. At the bottom level of the recursion, i.e., when $n < 2k + 2$, the algorithm uses the tree-spanner to connect all points, and, in addition, it adds to the spanner edges that connect each of the two sentinels $rt(T)$ and $l(T)$ to all the other $n - 1$ points.

The properties of the spanner $\mathcal{H}_k(n)$ are summarized in the next theorem.

Theorem 3. *For any tree metric M_T and a parameter k, there exists a 1-spanner $\mathcal{H}_k(n)$ with $O(n)$ edges, maximum degree at most $\Delta(T) + 2k$, diameter $O(\log_k n + \alpha(k))$ and lightness $O(k \log_k n)$.*

We remark that the maximum degree $\Delta(\mathcal{H})$ of the spanner $\mathcal{H} = \mathcal{H}_k(n)$ cannot be in general smaller than the maximum degree $\Delta(T)$ of the original tree. Indeed, consider a unit weight star T with edge set $\{(rt, v_1), (rt, v_2), \ldots, (rt, v_{n-1})\}$. Obviously, any spanner $\mathcal{H}$ for M_T with $\Delta(\mathcal{H}) < n - 1$ distorts the distance between the root rt and some other vertex.

1-Spanners with High Diameter. The next theorem gives our construction of 1-spanners for M_T with comparable monotone diameter in the range $\Omega(\log n)$.

Theorem 4. *For any tree metric M_T and a parameter k, there exists a 1-spanner with $O(n)$ edges, maximum degree at most $2\Delta(T)$, diameter $O(k \log_k n)$ and lightness $O(\log_k n)$. Moreover, this 1-spanner is a planar graph.*

3 Euclidean Spanners

In this section we demonstrate that our 1-spanners for tree metrics can be used for constructing Euclidean spanners and spanners for doubling metrics.

We start with employing the Dumbbell Theorem of [4] in conjunction with our 1-spanners for tree metrics to construct Euclidean spanners.

Theorem 5. *("Dumbbell Theorem", Theorem 2 in [4]) Given a set S of n points in $\mathbb{R}^d$ and a parameter $\epsilon > 0$, a forest $\mathcal{D}$ consisting of $O(1)$ rooted binary trees of size $O(n)$ can be built in $O(n \log n)$ time, having the following properties: 1) For*

[5] This may not hold true for two points that are not comparable, as their least common ancestor may not belong to $\tilde{\mathcal{Q}}$.

each tree in $\mathcal{D}$, there is a 1-1 correspondence between the leaves of this tree and the points of S. 2) Each internal vertex in the tree has a unique representative point, which can be selected arbitrarily from the points in any of its descendant leaves. 3) Given any two points $u, v \in S$, there is a tree in $\mathcal{D}$, so that the path formed by walking from representative to representative along the unique path in that tree between these vertices, is a $(1+\epsilon)$-spanner path for u and v.

For each dumbbell tree in $\mathcal{D}$, we use the following representative assignment from [4]. Leaf labels are propagated up the tree. An internal vertex chooses to itself one of the propagated labels and propagates the other one up the tree. Each label is used at most twice, once at a leaf and once at an internal vertex. Any label assignment induces a weight function over the edges of the dumbbell tree in the obvious way. (The weight of an edge is set to be the Euclidean distance between the representatives corresponding to the two endpoints of that edge.) Arya et al. [4] proved that the lightness of dumbbell trees is always $O(\log n)$, regardless of which representative assignment is chosen for the internal vertices.

Next, we describe our construction of Euclidean spanners with diameter in the range $\Omega(\alpha(n)) = \Lambda = O(\log n)$. For each dumbbell tree $DT_i \in \mathcal{D}$, denote by M_i the $O(n)$-point tree metric induced by DT_i. To obtain our construction of $(1+\epsilon)$-spanners with low diameter, we set $k = n^{1/\Lambda}$, and build the 1-spanner construction $\mathcal{H}^i = \mathcal{H}^i_k(O(n))$ of Theorem 3 for each of the tree metrics M_i. Then we translate each $\mathcal{H}^i$ to be a spanning subgraph $\hat{\mathcal{H}}^i$ of S in the obvious way. Let $\mathcal{E}_k(n)$ be the Euclidean spanner obtained from the union of all the graphs $\hat{\mathcal{H}}^i$.

It is easy to see that the number of edges in $\mathcal{E}_k(n)$ is $O(n)$.

Next, we show that $\Lambda(\mathcal{E}_k(n)) = O(\log_k n + \alpha(k))$. By the Dumbbell Theorem, for any pair of points $u, v \in S$, there exists a dumbbell tree DT_i, so that the unique path $P_{u,v}$ between u and v in DT_i is a $(1+\epsilon)$-spanner path. Theorem 3 implies that there is a 1-spanner path P in $\mathcal{H}^i$ between u and v that consists of $O(\log_k n + \alpha(k))$ hops. By the triangle inequality, the weight of the corresponding translated path $\hat{P}$ in $\hat{\mathcal{H}}^i$ is no greater than the weight of $P_{u,v}$. Hence, $\hat{P}$ is a $(1+\epsilon)$-spanner path for u and v that consists of $O(\log_k n + \alpha(k))$ hops.

We proceed by showing that $\Delta(\mathcal{E}_k(n)) = O(k)$. Since each dumbbell tree DT_i is binary, theorem 3 implies that $\Delta(\mathcal{H}^i) = O(k)$. Recall that each label is used at most twice in DT_i, and so $\Delta(\hat{\mathcal{H}}^i) \leq 2\Delta(\mathcal{H}^i) = O(k)$. The union of $O(1)$ such graphs will also have maximum degree $O(k)$.

Finally, we argue that the lightness $\Psi(\mathcal{E}_k(n))$ of $\mathcal{E}_k(n)$ is $O(k \log_k n \log n)$. Consider a dumbbell tree DT_i. Recall that the lightness of all dumbbell trees is $O(\log n)$, and so $w(DT_i) = O(\log n) w(MST(S))$. By Theorem 3, the weight $w(\mathcal{H}^i)$ of $\mathcal{H}^i$ is at most $O(k \log_k n) w(DT_i)$. By the triangle inequality, the weight of each edge in $\hat{\mathcal{H}}^i$ is no greater than the corresponding weight in $\mathcal{H}^i$, implying that $w(\hat{\mathcal{H}}^i) \leq w(\mathcal{H}^i) = O(k \log_k n \log n) w(MST(S))$. The union of $O(1)$ such graphs will also have weight $O(k \log_k n \log n) w(MST(S))$.

To obtain our construction of Euclidean spanners in the range $\Lambda = \Omega(\log n)$, we use our 1-spanners for tree metrics from Theorem 4 instead of Theorem 3.

Corollary 1. *For any set S of n points in $\mathbb{R}^d$, any $\epsilon > 0$ and a parameter k, there exists a $(1+\epsilon)$-spanner with $O(n)$ edges, maximum degree $O(k)$, diameter*

$O(\log_k n + \alpha(k))$ and lightness $O(k \log_k n \log n)$. There also exists a $(1+\epsilon)$-spanner with degree $O(1)$, diameter $O(k \log_k n)$ and lightness $O(\log_k n \log n)$.

We show that the lightness of well-separated pair constructions for random point sets in the unit cube is (w.h.p.) $O(1)$. Also, the lightness of well-separated pair constructions provides an asymptotic upper bound on the lightness of dumbbell trees. We derive the following result as a corollary.

Corollary 2. *For any set S of n points that are chosen independently and uniformly at random from the unit cube, any $\epsilon > 0$ and a parameter k, there exists a $(1+\epsilon)$-spanner with $O(n)$ edges, maximum degree $O(k)$, diameter $O(\log_k n + \alpha(k))$ and lightness (w.h.p.) $O(k \log_k n)$. There also exists a $(1+\epsilon)$-spanner with maximum degree $O(1)$, diameter $O(k \log_k n)$ and lightness (w.h.p.) $O(\log_k n)$.*

Chan et al. [8] showed that for any doubling metric (X, δ) there exists a $(1+\epsilon)$-spanner with constant maximum degree. On the way to this result they proved the following lemma, which we employ in conjunction with our 1-spanners for tree metrics to construct our spanners for doubling metrics.

Lemma 1 (Lemma 3.1 in [8]). *For any doubling metric (X, δ), there exists a collection $\mathcal{T}$ of $m = O(1)$ spanning trees for (X, δ), $\mathcal{T} = \{\tau_1, \tau_2, \ldots, \tau_m\}$, that satisfies the following two properties: 1) For each index $i \in [m]$, the maximum degree $\Delta(\tau_i)$ of the tree τ_i is constant, i.e., $\Delta(\tau_i) = O(1)$. 2) For each pair of points $x, y \in X$ there exists an index $i \in [m]$ such that $dist_{\tau_i}(x, y) = O(1)\delta(x, y)$.*

To obtain our spanners for doubling metrics we start with constructing the collection $\mathcal{T} = \{\tau_1, \tau_2, \ldots, \tau_m\}$ of spanning trees with properties listed in Lemma 1. Next, we apply Theorem 3 with some parameter k to construct a 1-spanner $\mathcal{Z}^i = \mathcal{Z}^i_k(n)$ for the tree metric induced by the ith tree τ_i in $\mathcal{T}$, for each $i \in [m]$. Our spanner $\mathcal{Z}$ is set to be the union of all the 1-spanners $\mathcal{Z}_i$, i.e., $\mathcal{Z} = \bigcup_{i=1}^{m} \mathcal{Z}_i$. We summarize the properties of the resulting spanner $\mathcal{Z}$ in the next statement.

Corollary 3. *For any n-point doubling metric (X, δ) and a parameter k, there is an $O(1)$-spanner $\mathcal{Z}$ with $O(n)$ edges, degree $O(k)$ and diameter $O(\log_k n + \alpha(k))$.*

Acknowledgments. We are grateful to Sunil Arya, David Mount and Michiel Smid for helpful discussions.

References

1. Agarwal, P.K., Wang, Y., Yin, P.: Lower bound for sparse Euclidean spanners. In: Proc. of 16th SODA, pp. 670–671 (2005)
2. Alon, N., Schieber, B.: Optimal preprocessing for answering on-line product queries. Manuscript (1987)
3. Althöfer, I., Das, G., Dobkin, D.P., Joseph, D., Soares, J.: On sparse spanners of weighted graphs. Discrete & Computational Geometry 9, 81–100 (1993)
4. Arya, S., Das, G., Mount, D.M., Salowe, J.S., Smid, M.H.M.: Euclidean spanners: short, thin, and lanky. In: Proc. of 27th STOC, pp. 489–498 (1995)

5. Arya, S., Smid, M.H.M.: Efficient construction of a bounded degree spanner with low weight. Algorithmica 17(1), 33–54 (1997)
6. Bodlaender, H.L., Tel, G., Santoro, N.: Trade-offs in non-reversing diameter. Nord. J. Comput. 1(1), 111–134 (1994)
7. Chan, H.T.-H., Gupta, A.: Small hop-diameter sparse spanners for doubling metrics. In: Proc. of 17th SODA, pp. 70–78 (2006)
8. Chan, H.T.-H., Gupta, A., Maggs, B.M., Zhou, S.: On hierarchical routing in doubling metrics. In: Proc. of 16th SODA, pp. 762–771 (2005)
9. Chazelle, B.: Computing on a free tree via complexity-preserving mappings. Algorithmica 2, 337–361 (1987)
10. Chazelle, B., Rosenberg, B.: The complexity of computing partial sums off-line. Int. J. Comput. Geom. Appl. 1, 33–45 (1991)
11. Chew, L.P.: There is a planar graph almost as good as the complete graph. In: Proc. of 2nd SOCG, pp. 169–177 (1986)
12. Das, G., Narasimhan, G.: A fast algorithm for constructing sparse Euclidean spanners. In: Proc. of 10th SOCG, pp. 132–139 (1994)
13. Das, G., Narasimhan, G., Salowe, J.S.: A new way to weigh malnourished Euclidean graphs. In: Proc. of 6th SODA, pp. 215–222 (1995)
14. Dinitz, Y., Elkin, M., Solomon, S.: Shallow-low-light trees, and tight lower bounds for Euclidean spanners. In: Proc. of 49th FOCS, pp. 519–528 (2008)
15. Gudmundsson, J., Levcopoulos, C., Narasimhan, G.: Fast greedy algorithms for constructing sparse geometric spanners. SIAM J. Comput. 31(5), 1479–1500 (2002)
16. Gudmundsson, J., Levcopoulos, C., Narasimhan, G., Smid, M.H.M.: Approximate distance oracles for geometric graphs. In: Proc. of 13th SODA, pp. 828–837 (2002)
17. Gudmundsson, J., Levcopoulos, C., Narasimhan, G., Smid, M.H.M.: Approximate distance oracles for geometric spanners. ACM Transactions on Algorithms 4(1) (2008)
18. Hassin, Y., Peleg, D.: Sparse communication networks and efficient routing in the plane. In: Proc. of 19th PODC, pp. 41–50 (2000)
19. Keil, J.M., Gutwin, C.A.: Classes of graphs which approximate the complete Euclidean graph. Discrete & Computational Geometry 7, 13–28 (1992)
20. Mansour, Y., Peleg, D.: An approximation algorithm for min-cost network design. DIMACS Series in Discr. Math and TCS 53, 97–106 (2000)
21. Narasimhan, G., Smid, M.: Geometric Spanner Networks. Cambridge University Press, Cambridge (2007)
22. Pătraşcu, M., Demaine, E.D.: Tight bounds for the partial-sums problem. In: Proc. of 15th SODA, pp. 20–29 (2004)
23. Rao, S., Smith, W.D.: Approximating geometrical graphs via "spanners" and "banyans". In: Proc. of 30th STOC, pp. 540–550 (1998)
24. Tarjan, R.E.: Applications of path compression on balanced trees. J. ACM 26(4), 690–715 (1979)
25. Thorup, M.: Parallel shortcutting of rooted trees. J. Algorithms 23(1), 139–159 (1997)
26. Yao, A.C.: Space-time tradeoff for answering range queries. In: Proc. of 14th STOC, pp. 128–136 (1982)