Distributed Spanner Construction in Doubling Metric Spaces *

Mirela Damian Saurav Pandit Sriram Pemmaraju

Abstract

This paper presents a distributed algorithm that runs on an n-node unit ball graph (UBG) G residing in a metric space of constant doubling dimension, and constructs, for any $\varepsilon > 0$, a $(1+\varepsilon)$-spanner H of G with maximum degree bounded above by a constant. In addition, we show H is "lightweight", in the following sense. Let Δ denote the aspect ratio of G, that is, the ratio of the length of a longest edge in G to the length of a shortest edge in G. The total weight of H is bounded above by $O(\log \Delta) \cdot wt(MST)$, where MST denotes a minimum spanning tree of the metric space. Finally, we show that H satisfies the so called *leapfrog property*, an immediate implication being that, for the special case of Euclidean metric spaces with fixed dimension, the weight of H is bounded above by $O(wt(MST))$. Thus, the current result subsumes the results of the authors in PODC 2006 that apply to Euclidean metric spaces, and extends these results to metric spaces with constant doubling dimension.

1 Introduction

A *unit ball graph* (UBG) is a graph whose vertices reside in some metric space and whose edges connect pairs of vertices at distance at most one. The *doubling dimension* of a metric space is the smallest ρ such that any ball in this metric space can be covered by 2^ρ balls of half the radius. It is easy to verify that the d-dimensional Euclidean space, equipped with any of the L_p norms, has doubling dimension $\Theta(d)$. If ρ is a fixed constant (independent of the size of the UBG), then we call the UBG a *doubling UBG*. In this paper we present a distributed algorithm for constructing a constant-degree, low-weight spanner for doubling UBGs.

Precisely stated, our result is this: for any fixed $\varepsilon > 0$, our algorithm runs in $O(\log^* n)$ communication rounds on an n-node doubling UBG G, to construct a $(1+\varepsilon)$-spanner H of G with maximum degree bounded above by a constant. In addition, we show that H is "lightweight," in the following sense. Let Δ denote the aspect ratio of G, that is, the ratio of the length of a longest edge in G to the length of a shortest edge in G. We show that the total weight of H is bounded above by $O(\log \Delta) \cdot wt(MST)$, where MST denotes a minimum spanning tree of the metric space. Since $wt(MST)$ is a lower bound on the weight of any spanner of G, we have an $O(\log \Delta)$-approximation. Finally, we also show that H satisfies the so called *leapfrog property* [6], an immediate implication being that, for the special case of Euclidean metric spaces with fixed dimension, the weight of H is bounded above by $O(wt(MST))$. Thus, our current result subsumes the results in [5] that apply to Euclidean metric spaces, and extends these results to metric spaces with constant doubling dimension.

*The first author is at the Department of Computer Science, Villanova University, Villanova, PA 19085. E-mail: `mirela.damian@villanova.edu`. The other two authors are at the Department of Computer Science, The University of Iowa, Iowa City, IA 52242-1419. E-mail: [`spandit`, `sriram`]`@cs.uiowa.edu`.

1.1 Topology Control

Our result is motivated by the *topology control* problem in wireless ad-hoc networks. For an overview of topology control, see the survey by Rajaraman [14]. Since an ad-hoc network does not come with fixed infrastructure, there is no topology to start with and informally speaking, the topology control problem is one of selecting neighbors for each node so that the resulting topology has a number of useful properties such as sparseness, small weight, or maximum vertex degree bounded above by a constant. Most topology control protocols that provide worst case guarantees on the quality of the topology assume that the network is modeled by a unit disk graph (UDG) (see [12] for a recent example). The results in this paper apply to the more general model of doubling unit ball graphs (UBG). Doubling metric spaces have received a great deal of attention recently [3, 9, 10, 11, 15], partly because they are thought to capture real-world phenomena such as latencies in peer-to-peer networks and in the Internet. Also, doubling metrics are robust in the sense that the doubling dimension is roughly preserved under distortion (see Proposition 3 in [15]). Thus distorted versions of low dimensional Euclidean space also have small doubling dimension. Consequently, doubling UBGs can model wireless networks in which nodes have non-uniform transmission ranges or have erroneous perception of distances to other nodes. Finally, doubling metrics imply the following "bounded growth" phenomenon that seems to be characteristic of large scale wireless ad-hoc and sensor networks: the number of nodes that are far away from each other and yet are all in the vicinity of a particular node, is small. In other words, no node can have an arbitrarily large independent set in its neighborhood.

1.2 Net Trees

Let (V, d) be a metric space with $|V| = n$ and doubling dimension ρ. In a recent paper, Chan, Gupta, Maggs, and Zhou [1] show how to construct, via a sequential, polynomial-time algorithm, a $(1+\varepsilon)$-spanner of (V, d) with maximum degree bounded above by $\left(\frac{1}{\varepsilon}\right)^{O(\rho)}$. We will refer to this algorithm as the CGMZ algorithm. The problem of constructing a spanner for a metric space can be thought of as a special case our problem, in which the given UBG is a complete graph. Underlining the result in [1] is the notion of *net trees*, independently proposed by Har-Peled and Mendel [8]. Let $B(u, r)$ denote the ball of radius r centered at point u. A subset $U \subseteq V$ is an *r-net* of V if it satisfies two properties:

r-packing: For every u and v in U, $d(u, v) > r$.
r-covering: The union $\cup_{u \in U} B(u, r)$ covers V.

Such nets always exist for any $r > 0$, and can be easily determined using a greedy algorithm. Assume without loss of generality that the largest pairwise distance in V is exactly 1 (this can be achieved by appropriate scaling). Let α, with $\sqrt{1+\varepsilon} \le \alpha \le 1 + \varepsilon/2$, and $\gamma \ge \frac{\alpha}{\alpha-1}$ be constants. Let h be the smallest positive integer such every pairwise distance is greater than $\frac{1}{\alpha^h}$. Let $r_0 = \frac{1}{\alpha^h}$ and let $r_i = \alpha \cdot r_{i-1}$, for $i > 0$. A *net tree* is a sequence of subsets $\langle V_0, V_1, V_2, \ldots, V_h \rangle$, such that $V_0 = V$ and V_i is an r_i-net of V_{i-1}, for $i > 0$. Note that every V_i, including V_0, is a r_i-packing. Also note that V_h, which is a 1-net of V_{h-1}, is a singleton, since the maximum separation between any pair of points is one. To view the sequence $\langle V_0, V_1, V_2, \ldots, V_h \rangle$ as a tree, let $i(v) = \max\{i \mid v \in V_i\}$ for each $v \in V$. Then, for each $v \in V$, $i(v) + 1$ copies of v appear as nodes in the tree. These are denoted $(0, v), (1, v), \ldots, (i(v), v)$, where (i, v) represents the occurrence of v in V_i. For each $0 \le i < i(v)$, the parent of node (i, v) is $(i + 1, v)$. Node $(i(v), v)$ has no parent and is the root of the net tree, if $i(v) = h$; otherwise, vertex $v \notin V_{i(v)+1}$ and there is some vertex $u \in V_{i(v)+1}$ such that $B(u, r_{i(v)+1})$ contains v. Arbitrarily, pick one such u and let $(i(v) + 1, u)$ be the parent of $(i(v), v)$. Informally speaking, higher levels in the net tree (leaves are at level 0) represent the structure of V at lower resolution. Figure 1 shows an example of a net tree with 6 levels.

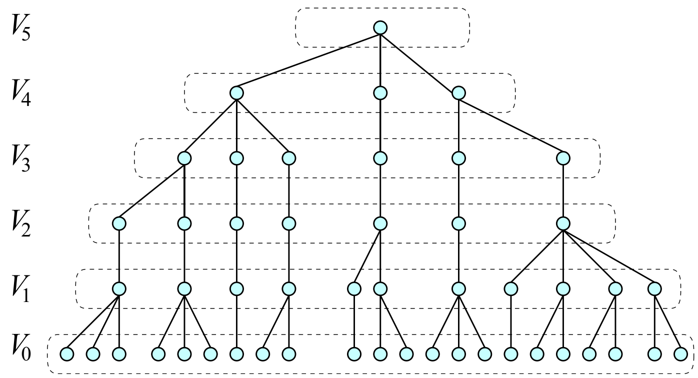

Figure 1: A net tree with six levels.

The CGMZ Algorithm.

1. Build a net tree $\langle V_0, V_1, \ldots, V_h \rangle$ of V.
2. Let $\lambda = \frac{\alpha}{\alpha-1}$, $\gamma = 2\lambda\left(1 + \frac{4\alpha}{\varepsilon}\right)$. Construct the edge sets
$$E_0 = \{\{u, v\} \in V_0 \times V_0 \mid \ d(u, v) \leq \gamma \cdot r_0\},$$
and
$$E_i = \{\{u, v\} \in V_i \times V_i \mid \ \gamma \cdot r_{i-1} < d(u, v) \leq \gamma \cdot r_i\},$$
for each $i = 1, \ldots, h$ and let $\widehat{E} = \cup_i E_i$.
3. Replace some edges in $\widehat{E}$ by other edges to obtain a new edge set $\widetilde{E}$.

Chan and coauthors [1] work with the version of the algorithm for $\alpha = 2$ and $\lambda = 2$. They show that the graph $H = (V, \widehat{E})$ obtained after Step (2) is a $(1+\varepsilon)$-spanner of the metric space and has linear number of edges, but may not satisfy the bounded degree requirement. Short paths in H can be obtained from the net tree in a natural manner. A uv-path in H whose length is at most $(1+\varepsilon) \cdot d(u, v)$ can be obtained by traveling up the net tree from the leaf u and from the leaf v until some level i is reached, such that the ancestors of u and v at level i are connected by an edge. In Step (3), a subset of the edges in $\widehat{E}$ is considered and each edge in this subset is replaced by at most one new edge. This step, which will be described in detail in Section 2.2, redistributes the edges so that all vertex-degrees are bounded above by a constant. The techniques used by Chan and coauthors for bounding vertex degrees play a critical role in this paper as well. In [5] we also describe an algorithm for constructing a bounded-degree $(1+\varepsilon)$-spanner for Euclidean UBGs, but our results rely on purely geometric arguments to bound the vertex degree of the constructed spanner. Chan and coauthors [1] obtain the following theorem.

Theorem 1 [Chan, Gupta, Maggs, Zhou] *Let (V, d) be a finite metric with doubling dimension bounded by ρ. For any $\varepsilon > 0$, the graph $(V, \widetilde{E})$ is a $(1+\varepsilon)$-spanner for (V, d), with maximum degree bounded above by $\left(\frac{1}{\varepsilon}\right)^{O(\rho)}$.*

Our algorithm is a modification of the CGMZ algorithm [1] that takes into account the fact that pairs of points separated by a distance greater than one are not connected by an edge and therefore such edges cannot be used in our spanner. A high level view of our algorithm is that it uses a slightly modified version of the CGMZ algorithm and constructs a graph H that may contain some *virtual edges*, that is, edges of length more than one. H has all the desired properties with respect to the input UBG G. Subsequently, we show how to replace each virtual edge in H by at most one *real edge*, that is, an edge of length at most one. The resulting graph is a constant-degree $(1+\varepsilon)$-spanner of G.

To obtain a distributed implementation of the above idea in $O(\log^* n)$ rounds, we use an algorithm due to Kuhn, Moscibroda, and Wattenhofer [11]. This algorithm computes a $(1, O(1))$-decomposition of a given doubling n-node UBG in $O(\log^* n)$ rounds, but can also be used to compute a net tree. After computing the net tree, we require a constant number of additional rounds to construct the spanner.

2 Spanners for Doubling UBGs

Let (V, d) be a metric space with doubling dimension ρ. Let $G = (V, E)$ be the UBG induced by this metric space. Thus, for all $u, v \in V$, $u \neq v$, $\{u, v\} \in E$ iff $d(u, v) \leq 1$. For a fixed $\varepsilon > 0$, let the quantities h, r_i and γ be defined as in Section 1.2. Run Steps (1) and (2) of the CGMZ Algorithm to construct a set of edges $\widehat{E}$. Let $H = (V, \widehat{E})$. Note that V_h may not be a singleton since V may contain points whose pairwise distance is more than one. So the sequence $\langle V_0, V_1, \ldots, V_h \rangle$ should be viewed as a forest of net trees, rooted at points in V_h. Recall that $\widehat{E} = \cup_{i=0}^{h} E_i$ and further recall that for $i > 0$, E_i consists of edges connecting all pairs of points $u, v \in V$ such that $d(u, v) \in (\gamma \cdot r_{i-1}, \gamma \cdot r_i]$. Note that there are values of i for which the right endpoint of the interval $(\gamma \cdot r_{i-1}, \gamma \cdot r_i]$ may be greater than one and for such values of i, E_i may contain edges that are not in E. Thus H is not necessarily a subgraph of G. Let $\delta = \lceil \log_\alpha \gamma \rceil$. It is easy to verify that for $0 \leq i \leq h - \delta$, $E_i \subseteq E$; for $i = h - \delta + 1$, the edge-set E_i may contain some edges in E and some edges not in E; and for $i > h - \delta + 1$, all edges in E_i are outside E. We call edges in H that also belong to E, *real edges*. Any edge in H that is not real is a *virtual edge*. Clearly, a spanner for G may not contain virtual edges, however virtual edges in H do carry important proximity information that will provide clues on how to replace them with real edges.

2.1 Properties of H

We will now prove some important properties of H. Let d_H be the distance metric induced by shortest paths in H. Specifically, we will show that H satisfies the following three properties:

1. For every $\{u, v\} \in E$, $d_H(u, v) \leq (1 + \varepsilon) \cdot d(u, v)$ (Lemma 6).
2. Edges of H can be oriented in such a way that the out-degree of H is bounded by $\left(\frac{1}{\varepsilon}\right)^{O(\rho)}$ (Lemma 7).
3. The weight of H is $wt(H) = O(\log \Delta) \cdot \left(\frac{1}{\varepsilon}\right)^{O(\rho)} \cdot wt(MST)$ (Lemma 8).

Property (1) implies that H is connected, since G is assumed to be connected. Property (2) implies that H has a linear number of edges, though it does not imply that H has bounded maximum degree. In Section 2.2 we describe a method to alter H so as to bound the in-degree of H as well, while maintaining all the properties listed above. The proofs of these properties are based on some intermediate results, that we now establish. Proofs of Lemma 6 and Lemma 7 are similar to those in [2]. The next observation follows immediately from the definition of the doubling dimension of a metric space.

Proposition 2 *If (X, d) is a metric with doubling dimension ρ and $Y \subseteq X$ is a subset of points with aspect ratio Δ, then $|Y| \leq 2^{\rho \cdot \lceil \log_2 \Delta \rceil}$.*

For any point $u \in V_i$, let $N_i(u) = \{v \in V_i \mid \{u, v\} \in E_i\}$ denote the set of points connected to u by edges in E_i. We now show an upper bound on the size of $N_i(u)$.

Lemma 3 *For each $u \in V_i$, $|N_i(u)| \leq \left(\frac{1}{\varepsilon}\right)^{O(\rho)}$.*

Proof: That the aspect ratio of $N_i(u)$ is bounded by 2γ follows from two observations: (1) any two points in $N_i(u)$ are more than distance r_i apart, and (2) any point in $N_i(u)$ is at distance at most $\gamma \cdot r_i$ from u and therefore, by using the triangle inequality, any two points in $N_i(u)$ are at most $2\gamma \cdot r_i$ apart. Then Proposition 2 implies the lemma. □

Lemma 4 *Suppose $u, v \in V_i$ and $d(u,v) \leq \gamma \cdot r_i$. Then $\{u,v\} \in \widehat{E}$.*

Proof: If $\gamma \cdot r_{i-1} < d(u,v) \leq \gamma \cdot r_i$, then $\{u,v\} \in E_i$. Otherwise, (a) $d(u,v) \leq \gamma \cdot r_0$ or (b) for some $j < i$, $\gamma \cdot r_{j-1} < d(u,v) \leq \gamma \cdot r_j$. Since $V_i \subseteq V_j$ for all $0 \leq j \leq i$, in case (a), $\{u,v\} \in E_0$ and in case (b), $\{u,v\} \in E_j$. □

Lemma 5 *For each $u \in V$ and for each i, there exists $v \in V_i$ such that $d_H(u,v) \leq \lambda \cdot r_i$.*

Proof: The proof is by induction on i. For $i = 0$, $u \in V_0 = V$ and $d_H(u,u) = 0 < \lambda \cdot r_0$, proving this case true. For $i > 0$, apply the inductive hypothesis to infer that there exists $w \in V_{i-1}$ such that $d_H(u,w) \leq \lambda \cdot r_{i-1}$. Furthermore, since V_i is an r_i-net of V_{i-1}, there exists $v \in V_i \subseteq V_{i-1}$ such that $d(w,v) \leq r_i \leq \gamma \cdot r_{i-1}$. Therefore, by Lemma 4, $\{w,v\} \in \widehat{E}$ and hence $d_H(w,v) = d(w,v) \leq r_i$. By the triangle inequality we have that $d_H(u,v) \leq d_H(u,w) + d_H(w,v) \leq \lambda \cdot r_{i-1} + r_i = \lambda \cdot r_i$. □

In addition to proving the existence of a vertex v at each level i, Lemma 5 implies a certain path from vertex u to $v \in V_i$. Start from node $(0,u)$ in the tree (that is, the copy of u corresponding to a leaf) and follow the path through a sequence of parents, until node (i,v) is reached. Lemma 5 shows that the distance in H along this path is at most $\lambda \cdot r_i$.

Lemma 6 [Property 1] *For any edge $\{u,v\} \in E$, $d_H(u,v) \leq (1+\varepsilon) \cdot d(u,v)$.*

Proof: Let q be the smallest integer such that $\frac{4\lambda}{\alpha^q} \leq \varepsilon < \frac{8\lambda}{\alpha^q}$. Thus $q = \lceil \log_\alpha \frac{4\lambda}{\varepsilon} \rceil$. Let i be such that $r_i \leq d(u,v) \leq r_{i+1}$, and assume first that $i \leq q-1$. Then $d(u,v) \leq \alpha^q \cdot r_0 \leq \frac{8\lambda}{\varepsilon} \cdot r_0 \leq \gamma r_0$. Since both u and v belong to V_0, by Lemma 4, we have that $\{u,v\} \in \widehat{E}$. This implies that $d_H(u,v) = d(u,v)$, proving the lemma true for this case. Assume now that $i \geq q$ and let $s = i - q \geq 0$. Note that $r_i = \alpha^q \cdot r_s$. By Lemma 5, there exist $x,\ y \in V_s$ such that $d_H(u,x) \leq \lambda \cdot r_s$ and $d_H(v,y) \leq \lambda \cdot r_s$. By the triangle inequality,

$$\begin{aligned}
d(x,y) &\leq d(x,u) + d(u,v) + d(v,y) \\
&\leq \lambda \cdot r_s + \alpha \cdot r_i + \lambda \cdot r_s \\
&= r_s(2\lambda + \alpha \cdot \alpha^q) \\
&\leq r_s(2\lambda + \alpha \tfrac{8\lambda}{\varepsilon}) \\
&= \gamma \cdot r_s
\end{aligned}$$

Hence, by Lemma 4, $d_H(x,y) = d(x,y)$. Using the triangle inequality again, we get

$$\begin{aligned}
d_H(u,v) &\leq d_H(u,x) + d_H(x,y) + d_H(y,v) \\
&\leq 2\lambda \cdot r_s + d(x,y) \\
&\leq 4\lambda \cdot r_s + d(u,v) \\
&\leq (1 + \tfrac{4\lambda}{\alpha^q}) \cdot d(u,v) \\
&\leq (1+\varepsilon) \cdot d(u,v)
\end{aligned}$$

This completes the proof. □

Lemma 6 also identifies a short uv-path in H. Simply follow the sequence of parents, starting at the node $(0,u)$ in the tree and similarly, starting at the node $(0,v)$. At a certain level (denoted s in the proof), the ancestor of u and the ancestor of v at that level are connected by an edge.

We now prove Property (2) of H. For each point u, define $i(u) = \max\{i \mid u \in V_i\}$, and for each edge $\{u,v\} \in \widehat{E}$, direct $\{u,v\}$ from u to v, if $i(u) < i(v)$. If $i(u) = i(v)$, pick an arbitrary orientation. This edge orientation is identical to the one used in [1]. Call the resulting digraph $\overrightarrow{H}$.

Lemma 7 **[Property 2]** *The out-degree of $\overrightarrow{H}$ is bounded above by $(\frac{1}{\epsilon})^{O(\rho)}$.*

Proof: Let $\{u,v\} \in \widehat{E}$ be an arbitrary edge directed from u to v, and let i be such that $\{u,v\} \in E_i$. Then $d(u,v) \leq \gamma \cdot r_i$. Now note that $r_{i+\delta} = \alpha^{\delta} \cdot r_i \geq \gamma \cdot r_i$ (recall that $\delta = \lceil \log_\alpha \gamma \rceil$). This, along with the fact that $V_{i+\delta}$ is an $r_{i+\delta}$-net, implies that it is not possible for both u and v to exist in $V_{i+\delta}$. Since $i(u) \leq i(v)$ (by our assumption), it follows that $i(u) \leq i+\delta$. On the other hand, $u \in V_i$ and so $i(u) \geq i$.

Summarizing, we have that $i(u) - \delta \leq i \leq i(u)$. This tells us that there are at most $\delta + 1 = O(\log_\alpha \gamma)$ values of i for which E_i may contain an edge outgoing from u. For each such i, by Lemma 3 there are at most $|N_i(u)| \leq \left(\frac{1}{\varepsilon}\right)^{O(\rho)}$ edges in E_i outgoing from u. It follows that the total number of edges in $\widehat{E}$ outgoing from u is $\left(\frac{1}{\varepsilon}\right)^{O(\rho)} \cdot O(\log(1/\varepsilon)) = \left(\frac{1}{\epsilon}\right)^{O(\rho)}$. $\square$

We now prove Property (3) of H, showing that H has bounded weight.

Lemma 8 **[Property 3]** *The total weight of H is $wt(H) = O(\log \Delta) \cdot \left(\frac{1}{\epsilon}\right)^{O(\rho)} \cdot wt(MST)$, where MST is a minimum spanning tree of V, and Δ is the aspect ratio of G.*

Proof: We show that, for each i, $wt(E_i) = \left(\frac{1}{\epsilon}\right)^{O(\rho)} \cdot wt(MST)$. This along with the fact that there are $h + 1 = \log_\alpha \frac{1}{r_0} + 1 = O(\log_\alpha \Delta)$ levels i, proves the claim of the lemma.

Let $U_i \subseteq V_i$ be the points in V_i incident to edges in E_i, and let $t = |U_i|$. Recall that any edge $\{u,v\} \in E_i$ satisfies $r_i < d(u,v) \leq \gamma \cdot r_i$. Thus, any spanning tree of a set of points containing U_i has weight at least $(t-1) \cdot r_i$, implying that $wt(MST) \geq (t-1) \cdot r_i$. Also note that the weight of E_i is bounded by $\Sigma_{u \in U_i} |N_i(u)| \cdot \gamma \cdot r_i \leq \left(\frac{1}{\varepsilon}\right)^{O(\rho)} \cdot t \cdot \gamma \cdot r_i$, using the upper bound on $|N_i(u)|$ given by Lemma 3. This completes the proof. $\square$

2.2 Altering H for Bounded Degree

In this section we show how to modify H so as to bound the degree of each vertex by a constant. Lemma 7 shows that an oriented version of H, namely $\overrightarrow{H}$, has bounded out-degree. Next we describe a method that carefully replaces some directed edges in $\overrightarrow{H}$ by others so as to guarantee constant bound on the in-degree as well, without increasing the out-degree. The replacement procedure is similar to the one used in [1], slightly adjusted to work with UDGs. Assume without loss of generality that $\varepsilon \leq \frac{1}{2}$; otherwise, if $\varepsilon > \frac{1}{2}$, we proceed with $\varepsilon = \frac{1}{2}$. We use the fact that $\varepsilon \leq \frac{1}{2}$ in the proof of Lemma 11. Let ℓ be the smallest positive integer such that $\frac{1}{\alpha^{\ell-1}} \leq \varepsilon$. Thus $\ell = O(\log_\alpha \frac{1}{\varepsilon})$.

Edge Replacement Procedure. Let u be an arbitrary point in V and let $M(u,i)$ be the set of all vertices $v \in V_i$ such that $\{v,u\}$ is an edge in E_i directed from v to u in $\overrightarrow{H}$. Let $I(u) = \langle i_1, i_2, \ldots \rangle$ be the increasing sequence of all indices i_k for which $M(u,i_k)$ is nonempty. For $1 \leq k \leq \ell$, we do not disturb any of the edges from points in $M(u,i_k)$ to u. For each $k > \ell$ such that $i_k \leq h - \delta - 2$, edges $\{v,u\}$ connecting $v \in M(u,i_k)$ to u are replaced by other edges. Specifically, an edge $\{v,u\}$, $v \in M(u,i_k)$ is replaced by an edge $\{v,w\}$, where w is an arbitrary vertex in $M(u,i_{k-\ell})$. The replacement can be equivalently viewed as happening in H or its oriented version $\overrightarrow{H}$. In $\overrightarrow{H}$, we replace the directed edge (v,u) by the directed edge (v,w). In the next two lemmas, our arguments will use $\overrightarrow{H}$ or H, as convenient.

Let $\widetilde{E}$ be the resulting set of edges. By our construction, $|\widetilde{E}| \leq |\widehat{E}|$. An important observation here is that the replacement procedure above is carried out only for edges in E_i, with $i \leq h - \delta - 2$ (that is, only edges of length no greater than $1/\alpha^2$). This is to ensure that only real edges get replaced and no virtual edges get added, a guarantee that is shown in the following lemma.

Lemma 9 *$\widetilde{E} \setminus \widehat{E}$ contains no virtual edges.*

Proof: Let $\{v, u\}$ be an edge that gets replaced by $\{v, w\}$, with $v \in M(u, i_k)$ and $w \in M(u, i_{k-\ell})$. Recall that $k > \ell$ and $i_k \leq h - \delta - 2$. Using the definitions of E_{i_k} and $E_{i_{k-\ell}}$ and the fact that $\frac{1}{\alpha^{\ell-1}} \leq \varepsilon$, it follows that $d(w, u) \leq \varepsilon \cdot d(v, u)$. By the triangle inequality, $d(v, w) \leq d(w, u) + d(v, u) \leq (1 + \varepsilon)d(v, u)$. Now note that $d(v, u) \leq 1/\alpha^2$. This is because edges in E_{i_k} have length no greater than $\gamma \cdot r_{i_k} \leq 1/\alpha^2$, for any $i_k \leq h - \delta - 2$. Therefore $d(v, w) \leq (1+\varepsilon)/\alpha^2 \leq 1$, for any $\alpha^2 \geq (1+\varepsilon)$. □

Let $J = (V, \widetilde{E})$. First we show that J indeed has bounded degree (Lemma 10). Second we show that the metric distance d_J induced by shortest paths in J is a good approximation of d_H (Lemma 11). A consequence of this is that J remains connected, and maintains spanner paths between endpoints of real edges.

Lemma 10 *Every vertex in $J = (V, \widetilde{E})$ has degree bounded by $(\frac{1}{\epsilon})^{O(\rho)}$.*

Proof: Let A be the maximum out-degree of a vertex of $\overrightarrow{H}$. By Lemma 7, $A \leq (\frac{1}{\epsilon})^{O(\rho)}$. Let B be the largest of $|N_i(u)|$, for all i and all u. Then, by Lemma 3, $B \leq (\frac{1}{\epsilon})^{O(\rho)}$. The edge-replacement procedure replaces a directed edge (v, u) by a directed edge (v, w). So the out-degrees of vertices remains unchanged by the edge-replacement procedure, and continue to be bounded above by $(\frac{1}{\epsilon})^{O(\rho)}$. Thus, we can simply focus on the in-degrees of vertices. We bound these by accounting for the in-degree of an arbitrary vertex x with respect to old edges (in $\widetilde{E} \cap \widehat{E}$) and with respect to new edges (in $\widetilde{E} \setminus \widehat{E}$); we show that both in-degrees are bounded above by $(\frac{1}{\epsilon})^{O(\rho)}$.

In-degree of x w.r.t. $\widetilde{E} \cap \widehat{E}$. Out of the edges in $\overrightarrow{H}$ that come into x, at most $B(\ell + \delta + 2)$ remain in $\widetilde{E}$. More specifically, at most B edges at each of the first ℓ levels $i_1, i_2, \ldots, i_\ell$ in $I(x)$, plus at most B edges in each of E_i, $i = h - \delta - 1, h - \delta, \ldots, h$, remain in $\widetilde{E}$. Any other edge directed into x gets replaced by an edge not incident to x. We end this case by noting that $B(\ell + \delta + 2) = (\frac{1}{\epsilon})^{O(\rho)}$.

In-degree of x w.r.t. $\widetilde{E} \setminus \widehat{E}$. Vertex x has a new in-coming edge whenever it plays the role of w in the edge-replacement procedure. For each edge (w, u), there are at most B qualifying edges (v, u) directed into u. Furthermore, there are A edges (w, u) outgoing from w. This gives an upper bound of $AB = (\frac{1}{\epsilon})^{O(\rho)}$ on the in-degree of x. □

It remains to show that d_J is a good approximation of d_H. Intuition for this is provided by the proof of Lemma 3. In that proof, it is shown that when $\{v, w\}$ replaces $\{v, u\}$, $d(w, u) \leq \varepsilon \cdot d(v, u)$ and $d(v, w) \leq (1 + \varepsilon) \cdot d(v, u)$. Thus, if the path $\langle v, w, u \rangle$ existed in $\widetilde{E}$, this path would have length at most $(1 + 2\varepsilon) \cdot d(v, u)$. However, edge $\{w, u\}$ may not exist in $\widetilde{E}$, since it may itself have been replaced. Thus a shortest path from w to u in $\widetilde{E}$ may be longer than $d(w, u)$. However, since $d(w, u) \leq \varepsilon \cdot d(v, u)$, the extra cost of replacing $\{w, u\}$ is marginal and the eventual sum of all of these lengths is still bounded above by $(1 + 2\varepsilon) \cdot d(v, u)$. Thus we have the following lemma, whose proof appears in the appendix (due to space restrictions).

Lemma 11 *$d_J \leq (1 + 2\varepsilon)d_H$.*

2.3 Eliminating Virtual Edges

The only impediment in having $J = (V, \widetilde{E})$ serve as a spanner for the input UDG G is the presence of virtual edges in J. Recall that these are edges of length greater than one and clearly do not exist in G. In this section we show that there exist real edges that can take over the role of virtual edges in J, without violating the properties J is expected to have.

Let $\{u, v\} \in E$ be an arbitrary edge and let i be such that $r_i \leq d(u, v) < r_{i+1}$. Let q be as in the proof of Lemma 6: the smallest integer such that $\frac{4\lambda}{\alpha^q} \leq \varepsilon < \frac{8\lambda}{\alpha^q}$. As mentioned before, the proof of Lemma 6 implies a certain uv-path of length at most $(1+\varepsilon)\cdot d(u,v)$ in $H=(V,\widehat{E})$. If $i \le q-1$, this path is just the edge $\{u,v\}$, because $\{u,v\}$ is guaranteed to exist in $\widehat{E}$. The Edge Replacement Procedure (Section 2.2) ensures that only real edges are replaced and each real edge is replaced by a path consisting only of real edges. This ensures that even in $\widetilde{E}$ there is a uv-path of length at most $(1+2\varepsilon)\cdot d(u,v)$, consisting of real edges only. If $i \ge q$, the uv-path in H implied by Lemma 6 may have more than one edge. Let $s=i-q$ and (s,u^*) (respectively, (s,v^*)) be the level-s ancestor of the leaf $(0,u)$ (respectively, the leaf $(0,v)$) in the net tree $\langle V_0, V_1, \ldots, V_h\rangle$. Then edge $\{u^*,v^*\}$ is guaranteed to be present in $\widehat{E}$ and the uv-path implied by Lemma 6 starts at $(0,u)$, goes up to the net tree via parents to (s,u^*), goes to (s,v^*), and then follows the unique path down the tree from (s,v^*) to $(0,v)$. It is easy to check that of all the edges in this path, only $\{u^*,v^*\}$ may be virtual. Specifically, when the edge $\{u,v\}$ is long enough to guarantee that $i \ge h-\delta+1+q$, then $s=i-q\ge h-\delta+1$ and the edge $\{u^*,v^*\}$ may belong to E_s. Recall that for $j \ge h-\delta+1$, edges in E_j may not be real and in particular $\{u^*,v^*\}$ may be a virtual edge. Since the uv-path implied by Lemma 6 passes through edge $\{u^*,v^*\}$, one has to be careful in replacing $\{u^*,v^*\}$ by a real edge. Our virtual edge replacement procedure is given below.

For any node (i,v) in the net tree, let $T(i,v)$ denote the set of all vertices $u\in V$, such that the subtree of the net tree rooted at (i,v) contains a copy of u. In other words, $T(i,v)=\{u\in V \mid (i,v)$ is an ancestor of (j,u) for some $j\le i\}$.

Virtual Edge Replacement Procedure. For each virtual edge $\{u,v\}\in E_i$, if there is a real edge $\{x,y\}$ already in the spanner H, with $x\in T(i,u)$ and $y\in T(i,v)$, then there is nothing to do. Similarly, if there is no such real edge $\{x,y\}$ in the input graph G, then there is nothing to do. Otherwise, find a real edge $\{x,y\}\in E$, $x\in T(i,u)$ and $y\in T(i,v)$, and replace $\{u,v\}$ by $\{x,y\}$.

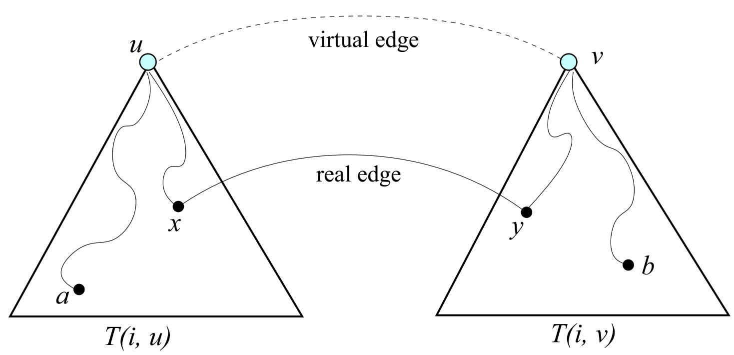

Figure 2: A short ab-path passes through virtual edge $\{u,v\}$. After replacing virtual edge $\{u,v\}$ by real edge $\{x,y\}$, there is a short ab-path through $\{x,y\}$.

The reason why this replacement procedure works can be intuitively explained as follows. A virtual edge $\{u,v\}\in E_i$ is important for pairs of vertices $\{a,b\}$, with $a\in T(i,u)$ and $b\in T(i,v)$, for which all ab-paths of length at most $(1+\varepsilon)\cdot d(a,b)$ pass through $\{u,v\}$. Replacing $\{u,v\}$ by $\{x,y\}$ provides the following alternate ab-path that is short enough: starting at the leaf a, go up the tree rooted at (i,u) via parents until an ancestor common to a and x is reached, then come down to x, take edge $\{x,y\}$, go up the tree rooted at (i,v) until an ancestor common to b and y is reached, and finally go down to b. See Figure 2 for an illustration. Note that this entire path consists only of real edges.

We finally state our main result. Let G' be the graph obtained from J by replacing virtual edges using the Virtual Edge Replacement Procedure. Let $d_{G'}$ be the distance metric induced by shortest paths in $G'=(V,E')$.

Theorem 12 *$G'=(V,E')$ is a $(1+\varepsilon)$-spanner of G with degree bounded by $(\frac{1}{\varepsilon})^{O(\rho)}$ and weight bounded by $O(\log\Delta)\cdot(\frac{1}{\varepsilon})^{O(\rho)}\cdot wt(MST)$.*

A proof similar to that of Lemma 6 can be used the show the spanner property of G'. This high level ideas are also discussed earlier in this section. The fact that G' is lightweight simply follows from the fact that a virtual edge of length greater than one in J, either gets eliminated, or gets replaced by at most one real edge of length at most one in G'. The constant degree bound follows from the observation that, for a vertex x to acquire a new incident edge, there is an ancestor of x in the net tree at level $h-\delta+1$ or higher, that loses an incident edge at that level. There are a constant number of such ancestors and from Lemma 3, we know that any vertex has a constant number of incident edges at any particular level.

We conclude this section with a summary of our algorithm.

`Algorithm SPANNER(`(V,d)`, `ε`)`

Let $\sqrt{1+\varepsilon} < \alpha \le 1+\varepsilon/2$ be a constant, $\lambda = \frac{\alpha}{\alpha-1}$, $\gamma = 2\lambda\left(1+\frac{4\alpha}{\varepsilon}\right)$, and $\delta = \lceil \log_\alpha \gamma \rceil$.
Let h be the smallest integer such that $\frac{1}{\alpha^h}$ is smaller than the minimum inter-point distance.
Let $r_0 = \frac{1}{\alpha^h}$ and let $r_i = \alpha \cdot r_{i-1}$, for all $i > 0$.

Constructing a linear size $(1+\varepsilon)$-spanner $H = (V, \widehat{E})$.
1. Construct the net tree $\langle V_0, V_1, \dots, V_h \rangle$.
   [Let $i(u) = \max\{i \mid u \in V_i\}$.]
2. Construct the sets
   $$E_0 = \{\{u,v\} \in V_0 \times V_0 \mid \ d(u,v) \le \gamma \cdot r_0\},$$
   $$E_i = \{\{u,v\} \in V_i \times V_i \mid \gamma \cdot r_{i-1} < \ d(u,v) \le \gamma \cdot r_i\}$$
   for $1 \le i \le h$.
   [Let $\widehat{E} = \cup_i E_i$ and $H = (V, \widehat{E})$.]

Replacing edges to obtain a constant degree bound.
3. Orient each edge $\{u,v\} \in \widehat{E}$ from u to v if $i(u) \le i(v)$, breaking ties arbitrarily.
   [Let $M(u,i)$ denote the set of vertices $v \in V_i$ such that there is an edge from v to u.]
4. For each $u \in V$, construct the increasing sequence $I(u) = \langle i_1, i_2, \dots, \rangle$ of all i_k with $M(u, i_k) \ne \emptyset$
   [Let ℓ be the smallest integer with $\frac{1}{2^{\ell-1}} \le \varepsilon$.]
5. For each $u \in V$ and each $i_k \in I(u)$, with $k > \ell$ and $i_k \le h-\delta-2$ do
6. Replace directed edge (v,u), $v \in M(u,i_k)$ by edge (v,w) for arbitrary $w \in M(u, i_{k-\ell})$.
   [Let $J = (V, \widetilde{E})$ be the resulting graph, with distance metric d_J.]

Replacing virtual edges by real ones.
7. For each $i \ge h-\delta+1$ and each virtual edge $\{u,v\} \in E_i$ do
8. If there is a real edge $\{x,y\} \in \widetilde{E}$, $x \in T(i,u)$ and $y \in T(i,v)$, then do nothing.
9. Otherwise, if there is a real edge $\{x,y\} \in E$, with $x \in T(i,u)$ and $y \in T(i,v)$, replace $\{u,v\}$ by $\{x,y\}$.
   [Let E' be the set of resulting edges. Output is $G' = (V, E')$.]

---

3 Leapfrog Property

In Lemma 8, we showed that $H = (V, \widehat{E})$ has total weight bounded above by $O(\log \Delta) \cdot \left(\frac{1}{\epsilon}\right)^{O(\rho)} \cdot wt(MST)$, where Δ is the aspect ratio of G. Thus, for fixed ε and constant doubling dimension ρ, the upper bound is within $O(\log \Delta)$ times the optimal value. In an attempt to show a bound that is within $O(1)$ times the optimal value, we use a tool that is widely used in the computational geometry literature [6, 4, 7]. In the context of building lightweight $(1+\varepsilon)$-spanners for Euclidean spaces, Das and Narasimhan [6] have shown that if the set of edges in the spanner satisfy a property known as the *leapfrog property*, then the total weight of the spanner is bounded above by $O(wt(MST))$. Below we state the leapfrog property precisely.

Leapfrog Property. For any $t \geq t' > 1$, a set F of edges has the (t', t)-leapfrog property if, for every subset $S = \{\{u_1, v_1\}, \{u_2, v_2\}, \ldots, \{u_s, v_s\}\}$ of F,

$$t' \cdot d(u_1, v_1) < \sum_{i=2}^{s} d(u_i, v_i) + t \cdot \Big(\sum_{i=1}^{s-1} d(v_i, u_{i+1}) + d(v_s, u_1) \Big). \tag{1}$$

Informally, this definition says that, if there exists an edge between u_1 and v_1, then any u_1v_1-path not including $\{u_1, v_1\}$ must have length greater than $t' \cdot d(u_1, v_1)$. Das and Narasimhan [6] show the following connection between the leapfrog property and the weight of the spanner.

Lemma 13 *Let $t \geq t' > 1$. If the line segments F in d-dimensional space satisfy the (t', t)-leapfrog property, then $wt(F) = O(wt(MST))$, where MST is a minimum spanning tree connecting the endpoints of line segments in F. The constant in the asymptotic notation depends on t, t' and d.*

It is well known that if a spanner is built "greedily", then the set of edges in the spanner satisfies the leapfrog property [6, 4, 7]. In [5] we showed that even a "relaxed" version of the greedy algorithm would ensure that the spanner edges have the leapfrog property. This was critical to showing that the spanner constructed in a distributed manner for UBGs in Euclidean spaces [5] had total weight bounded above by $O(wt(MST))$. Here we ask if it is possible to do the same for UBGs in metric spaces with constant doubling dimension. In an attempt to answer this question we show that, using a variant of the `SPANNER` algorithm (end of Section 2), we can build, for a given doubling UBG G, a $(1+\varepsilon)$-spanner with constant degree and with the (t, t')-leapfrog property, for some constants $t \geq t' > 1$. Note that this does not give us the desired $O(wt(MST))$ bound on the weight of the constructed spanner because we do know if the equivalent of Lemma 13 holds for non-Euclidean metric spaces. The proof of this lemma in [6] is quite geometric and does not suggest an approach to its generalization to metric spaces of constant doubling dimension.

To guarantee that the output spanner satisfies the (t', t)-leapfrog property, we need to make two modifications to the `SPANNER` algorithm. Let H_i denote the spanning subgraph of G induced by $E_0 \cup E_1 \cup \cdots \cup E_i$.

1. We modify Step (2) of the algorithm and place an edge $\{u, v\}$ into E_i only if $\{u, v\} \in V_i \times V_i$, $\gamma \cdot r_{i-1} < d(u, v) \leq \gamma \cdot r_i$, *and there is not already a uv-path of length at most* $(1+\varepsilon) \cdot d(u, v)$ *in* H_{i-1}.
2. Two edges $\{u, v\}$ and $\{u', v'\}$ in E_i are said to be *mutually redundant* if both of the following conditions hold:
   (a) $d_{H_{i-1}}(v, u') + d(u', v') + d_{H_{i-1}}(v', u) \leq (1+\varepsilon) \cdot d(u, v)$
   (b) $d_{H_{i-1}}(v', u) + d(u, v) + d_{H_{i-1}}(v, u') \leq (1+\varepsilon) \cdot d(u', v')$

   Note that these conditions imply that $H_i \setminus \{u, v\}$ contains a uv-path of length at most $(1+\varepsilon) \cdot d(u, v)$ and $H_i \setminus \{u', v'\}$ contains a $u'v'$-path of length at most $(1+\varepsilon) \cdot d(u', v')$. Thus, one of these can potentially be eliminated from H_i, without compromising the $(1+\varepsilon)$-spanner property of H_i. In fact, such mutually redundant pairs of edges need to be eliminated from H_i in order to show that H satisfies the leapfrog property.

These ideas and how they lead to a spanner that has the leapfrog property are discussed in detail in [5].

4 Distributed Implementation

In this section, we show that the `SPANNER` algorithm (end of Section 2) and its variant that ensures the leapfrog property (Section 3), both have distributed implementations that run in $O(\log^* n)$ rounds of communication. Here we focus on the `SPANNER` algorithm. It turns on that Step (1) of this algorithm takes $O(\log^* n)$ rounds, whereas the remaining steps take $O(1)$ additional rounds. We first examine Steps (2)-(9) of the algorithm.

It is easy to verify that in Steps (2)-(9), a node u needs to communicate only with other nodes that are either neighbors of u in G, or to which u is connected by a virtual edge. The main difficulty here is that the endpoints of a virtual edge $\{u, v\}$ may not be neighbors in the network. Consider a virtual edge $\{u, v\} \in E_i$. By definition of E_i, $d(u, v) \leq \gamma \cdot r_i \leq \gamma$. Even though the distance between u and v in the underlying metric space is bounded above by a constant, it is not necessary that the hop distance between u and v in G be similarly bounded above. Let us call a virtual edge $\{u, v\} \in E_i$, *useful*, if there exist $x \in T(i, u)$ and $y \in T(i, v)$ such that $\{x, y\} \in E$. Notice that only useful virtual edges need to be considered by our algorithm. If a virtual edge $\{u, v\}$ is not useful, then even though it is added in Step (2), it is eliminated in Steps (7)-(9). In the following lemma we show that the hop distance between endpoints of useful virtual edges is small.

Lemma 14 *The hop distance in G between the endpoints of any useful virtual edge $\{u, v\} \in E_i$ is at most* $2(2\lambda + 1)$.

Proof: By definition of a useful virtual edge, there are points $x \in T(i, u)$ and $y \in T(i, v)$ such that $\{x, y\}$ is an edge in G. Thus a path in G between u and v is the following: start at node (i, u) in the net tree and travel down to a copy of x, follow the edge $\{x, y\}$, and then travel up to node (i, v). Note that the edge $\{x, y\} \in E$, but it may not belong to $\widetilde{E}$. The length of this path is at most

$$2(1 + \frac{1}{\alpha} + \frac{1}{\alpha^2} + \ldots) + 1,$$

implying that $d_G(u, v) \leq 2\lambda + 1$. Now consider a shortest uv-path in G, say $\langle w_0 = u, w_1, \ldots, w_k = v\rangle$. Because G is a UBG and due to the triangle inequality, $d(w_i, w_{i+2}) > 1$ for all $0 \leq i \leq k - 2$. This yields a lower bound of $k/2$ on $d_G(u, v)$. Combining this with the upper bound of $2\lambda + 1$, we obtain that $k \leq 2(2\lambda + 1)$. $\square$

Thus, in Steps (2)-(9), a node only needs to communicate with nodes that at most $O(\lambda)$ hops away. This suggests a simply way of implementing Steps (2)-(9): after Step (1) is completed, each node u gathers neighborhood information and the values of $i(v)$ from all nodes v that are $O(\lambda)$ hops away. After this, node u can do all of its computation with no further communication.

The fact that Step (1) can be implemented in $O(\log^* n)$ rounds of communication follows from a clever argument in [11]. Suppose that we have computed the set V_{i-1}. The computation of the set V_i, which is an r_i-net of V_{i-1}, reduces to an maximal independent set (MIS) computation on a degree-bounded graph. To see this, create a graph, say G_i, whose vertex set is V_{i-1} and whose edges connect any pair of vertices $u, v \in V_{i-1}$, if $d(u, v) \leq r_i$. Then it is easy to see that an MIS in G_i is an r_i-net of V_{i-1}. Furthermore, the fact that G_i has bounded degree follows from the fact that the underlying metric space has bounded doubling dimension. There is a well-known algorithm due to Linial [13] for computing an MIS, that runs in $O(\log^* n)$ communication rounds on graphs with bounded degree. Using this algorithm, one can compute the r_i-net V_i of V_{i-1} in $O(\log^* n)$ rounds. Since there are $h + 1 = O(\log \Delta)$ such sets to compute, it seems like this approach will take $O(\log \Delta \cdot \log^* n)$ rounds. However, in [11] it is shown that in this algorithm, each node uses information only from nodes that are at most $O(\log^* n)$ hops away in G. Therefore, this algorithm has a $O(\log^* n)$-round implementation in which each node u first gathers information from nodes that are at most $O(\log^* n)$ hops away and then performs all steps of the `SPANNER` algorithm locally, using the collected information.

References

[1] Hubert T-H. Chan, Anupam Gupta, Bruce M. Maggs, and Shuheng Zhou. On hierarchical routing in doubling metrics. In *SODA '05: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 762–771, 2005.

[2] T-H. Hubert Chan. Personal Communication, 2006.

[3] T-H. Hubert Chan and Anupam Gupta. Small hop-diameter sparse spanners for doubling metrics. In *SODA '06: Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 70–78, 2006.

[4] A. Czumaj and H. Zhao. Fault-tolerant geometric spanners. *Discrete & Computational Geometry*, 32(2):207–230, 2004.

[5] Mirela Damian, Saurav Pandit, and Sriram Pemmaraju. Local approximation schemes for topology control. In *PODC '06: Proceedings of the twenty-fifth annual ACM SIGACT-SIGOPS symposium on Principles of distributed computing*, 2006.

[6] G. Das and G. Narasimhan. A fast algorithm for constructing sparse euclidean spanners. *Int. J. Comput. Geometry Appl.*, 7(4):297–315, 1997.

[7] J. Gudmundsson, C. Levcopoulos, and G. Narasimhan. Fast greedy algorithms for constructing sparse geometric spanners. *SIAM J. Comput.*, 31(5):1479–1500, 2002.

[8] S. Har-Peled and M. Mendel. Fast construction of nets in low dimensional metrics, and their applications. In *SCG'05: Proceedings of the 21st annual symposium on Computational geometry*, pages 150–158, 2005.

[9] R. Krauthgamer, A. Gupta, and J.R. Lee. Bounded geometries, fractals, and low-distortion embeddings. In *FOCS '03: Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*, pages 534–543, 2003.

[10] R. Krauthgamer and J.R. Lee. Navigating nets: simple algorithms for proximity search. In *SODA '04: Proceedings of the 15th annual ACM-SIAM symposium on Discrete algorithms*, pages 798–807, 2004.

[11] Fabian Kuhn, Thomas Moscibroda, and Roger Wattenhofer. On the locality of bounded growth. In *PODC '05: Proceedings of the twenty-fourth annual ACM SIGACT-SIGOPS symposium on Principles of distributed computing*, pages 60–68, 2005.

[12] Xiang-Yang Li and Yu Wang. Efficient construction of low weighted bounded degree planar spanner. *International Journal of Computational Geometry and Applications*, 14(1–2):69–84, 2004.

[13] Nathan Linial. Locality in distributed graph algorithms. *SIAM J. Comput.*, 21(1):193–201, 1992.

[14] R. Rajaraman. Topology control and routing in ad hoc networks: A survey. *SIGACT News*, 33:60–73, 2002.

[15] K. Talwar. Bypassing the embedding: algorithms for low dimensional metrics. In *STOC'04: Proceedings of the 36th annual ACM symposium on Theory of computing*, pages 281–290, 2004.

Appendix

Lemma 11. Proof: It suffices to show that, for each edge $\{v, u\} \in \widehat{E}$ that gets replaced, $d_J(v, u) \leq (1 + 2\varepsilon) \cdot d_H(v, u)$. Assume without loss of generality that edge $\{v, u\}$ directs into u, and let k be such that $v \in M(u, i_k)$. Then it must be that $k > \ell$ and $i_k < h - \delta$, otherwise $\{v, u\}$ would stay in $\widetilde{E}$.

Let $w_0 = v$, and assume that $\{w_0, u\}$ gets replaced by $\{w_0, w_1\}$. By construction, $w_1 \in M(u, i_{k-\ell})$. To avoid double-subscripts, through the remaining of this proof we use the index pair $\langle i, k\rangle$ to denote i_k. We now show that $d(w_1, u) \leq \varepsilon \cdot d(w_0, u)$ and $d(w_0, w_1) \leq (1+\varepsilon) \cdot d(w_0, u)$. This claim follows from the following observations:

1. $\langle i, k - \ell\rangle \leq \langle i, k\rangle - \ell$, since increasing indices in $I(u)$ are not necessarily incremental. This implies that $r_{\langle i,k-\ell\rangle} \leq r_{\langle i,k\rangle-\ell}$, which in turn implies that $d(w_1, u) \leq \gamma \cdot r_{\langle i,k\rangle-\ell} = \gamma \cdot r_{\langle i,k\rangle}/\alpha^{\ell}$.

2. $d(w_0, u) \geq \gamma \cdot r_{\langle i,k\rangle-1} = \gamma \cdot r_{\langle i,k\rangle}/\alpha$ (by definition). This along with the first observation implies that $d(w_1, u) \leq d(w_0, u)/2^{\ell-1} = \varepsilon \cdot d(w_0, u)$.

3. By the triangle inequality, $d(w_0, w_1) \leq d(w_1, u) + d(w_0, u) \leq (1 + \varepsilon) \cdot d(w_0, u)$.

So if $\{w_1, u\} \in \widetilde{E}$, then the claim of the lemma follows immediately from the observations above and the triangle inequality: $d_J(w_0, u) \leq d_J(w_0, w_1)+d_J(w_1, u) = d(w_0, w_1)+d(w_1, u) \leq (1+2\varepsilon)\cdot d(w_0, u)$. Otherwise, $\{w_1, u\} \in \widehat{E}$ in turn gets replaced by $\{w_1, w_2\} \in \widetilde{E}$, and the process repeats itself. Let $w_0, w_1, \ldots, w_r$ be a shortest path in J that leads to an edge $\{w_r, u\} \in \widetilde{E} \cap \widehat{E}$. It is not difficult to see that such a path always exists. This means that $\{w_0, w_1\}, \{w_1, w_2\}, \ldots, \{w_{r-1}, w_r\}$ are all new edges in $\widetilde{E} \cap \widehat{E}$. The three observations above translated to lower levels yield, for each $j = 1, 2, \ldots, r$, the following two inequalities: (i) $d(w_j, u) \leq \varepsilon \cdot d(w_{j-1}, u)$, and (ii) $d(w_{j-1}, w_j) \leq (1+\varepsilon)\cdot d(w_{j-1}, u)$. Repeated application of the first inequality yields $d(w_j, u) \leq \varepsilon^j \cdot d(w_0, u)$. Finally, we have:

$$\begin{aligned}
d_J(v, u) &\leq \sum_{j=1}^{r} d(w_{j-1}, w_j) + d(w_r, u) \\
&\leq (1 + \varepsilon) \sum_{j=1}^{r} \varepsilon^{j-1} d(w_0, u) + \varepsilon^r d(w_0, u) \\
&\leq d(w_0, u) \cdot (1 + \varepsilon)/(1 - \varepsilon) \\
&\leq (1 + 2\varepsilon) \cdot d(v, u)
\end{aligned}$$

This latter inequality follows from the fact that, for $0 < \varepsilon < 1/2$, $(1 + \varepsilon)(1 - \varepsilon) \leq 1 + 2\varepsilon$. $\square$