Distributed Approaches to Triangulation and Embedding [*]

Aleksandrs Slivkins[†]

June 2004
Revised: November 2004, July 2006.

Abstract

A number of recent papers in the networking community study the distance matrix defined by the node-to-node latencies in the Internet and, in particular, provide a number of quite successful distributed approaches that embed this distance into a low-dimensional Euclidean space. In such algorithms it is feasible to measure distances among only a linear or near-linear number of node pairs; the rest of the distances are simply not available. Moreover, for applications it is desirable to spread the load evenly among the participating nodes. Indeed, several recent studies use this ’fully distributed’ approach and achieve, empirically, a low distortion for all but a small fraction of node pairs.

This is concurrent with the large body of theoretical work on metric embeddings, but there is a fundamental distinction: in the theoretical approaches to metric embeddings, full and centralized access to the distance matrix is assumed and heavily used. In this paper we present the first fully distributed embedding algorithm with provable distortion guarantees for doubling metrics (which have been proposed as a reasonable abstraction of Internet latencies), thus providing some insight into the empirical success of the recent *Vivaldi* algorithm [7]. The main ingredient of our embedding algorithm is an improved fully distributed algorithm for a more basic problem of *triangulation*, where the triangle inequality is used to infer the distances that have not been measured; this problem received a considerable attention in the networking community, and has also been studied theoretically in [19].

We use our techniques to extend ϵ-relaxed embeddings and triangulations to infinite metrics and arbitrary measures, and to improve on the approximate distance labeling scheme of Talwar [36].

1 Introduction

A number of recent papers in the networking community study the distance matrix defined by the node-to-node latencies in the Internet[1] (e.g. [9, 13, 8, 20, 37]) and, in particular, provide a number of quite successful distributed approaches that embed this distance into a low-dimensional Euclidean space [30, 7, 32, 5, 25]. In such algorithms it is feasible to measure distances among only a linear or near-linear number of node pairs; the rest of the distances are simply not available. For instance, the Global Network Positioning (*GNP*) algorithm of Ng and Zhang [30] uses the *beacon-based* approach where a small number of nodes (’beacons’) are selected uniformly at random in the network so that every node only measures distances to the beacons. Using only these measurements (and allowing some processing at the beacons) *GNP* empirically achieves low distortion for all but a small fraction of node pairs.

[*] Preliminary version of this paper has appeared in *16th ACM-SIAM SODA*, 2005

[†] Department of Computer Science, Cornell University, Ithaca, NY 14853. Email: *slivkins@cs.cornell.edu*. Supported by a David and Lucile Packard Foundation Fellowship of Jon Kleinberg.

[1] For Internet latencies the triangle inequality is not always observed; however, recent networking research indicates that severe triangle inequality violations are not widespread enough so that the node-to-node latencies can be usefully modeled by metrics.

The drawback of the beacon-based approach is the high load placed on the beacons. Indeed, the *Vivaldi* algorithm [7] and other works [32, 5, 25] that followed *GNP* provide embedding algorithms with similar empirical performance where the load on *every* node is small; here the load includes computation, communication, storage and the completion time. Informally, we call such algorithms *fully distributed*. It is an important challenge to find provable guarantees for these fully distributed embedding algorithms.

Distributed metric embeddings As pointed out by Kleinberg et al. [19], the above Internet-related setting gives rise to a new set of theoretical questions not covered by the existing rich theory of algorithmic metric embeddings (e.g. [18]) where a full access to the distance matrix is assumed and heavily used. In particular, in the embedding algorithm of Bourgain [3] and Linial et al. [26], the coordinates are formed by measuring the distance from a point to a *set*; these sets can be as large as a constant fraction of nodes in a way that would not be feasible to implement for all nodes in the context of Internet.

It is easy to see that a beacon-based embedding cannot guarantee good distortion on *all* node pairs [19]. Accordingly, [19] formulated the notion of *ϵ-relaxed embedding* where for each node u there are at least $(1-\epsilon)n$ node pairs uv such that the uv-distance is embedded with the given distortion. To provide some theoretical insight into the empirical success of *GNP*, [19] came up with beacon-based algorithms that (for any fixed $\epsilon > 0$) use a small number of beacons and provably compute a low-distortion ϵ-relaxed embedding into low-dimensional L_p, $p \geq 1$ as long as the *doubling dimension* of the metric is small.

Here the doubling dimension of a metric is defined as the smallest k such that every ball can be covered by 2^k other balls of half the radius (see [2, 12]); a metric with this property is also called *2^k-doubling* or just *doubling* if k is a constant. Doubling metrics, which generalize the distance matrices of low-dimensional point sets in L_p, have been studied recently in the context of metric embeddings, nearest neighbour search and other problems [12, 24, 36, 23, 22, 19, 28, 4]. At the same time, several recent studies suggest the bounded growth rate of balls as a useful way to capture the structural properties of the Internet distance matrix (see e.g. [34, 8, 30, 31, 41, 15]).

Our contributions: In this paper we settle a question left open by [19]: we show that there exists a fully distributed algorithm that embeds a doubling metric into a low-dimensional Euclidean space with low distortion, thus providing the first provable guarantees for the fully distributed embedding problem introduced in the networking community. Specifically, given an s-doubling metric, our algorithm computes an ϵ-relaxed embedding into any L_p, $p \geq 1$ with distortion and dimension $f(\epsilon, s)$, so that the per-node load is at most $f(\epsilon, s)(\log n)^4$, where n is the number of nodes. The main technical ingredient of our algorithm is a fully distributed triangulation algorithm that improves upon the one in [19] and is of independent interest; we discuss it next.

We assume that the network provides the following functionality. Firstly, every node can, at unit cost, communicate with (and, in particular, measure distance to) any other node given its ID. Secondly, every node can select a node ID independently and uniformly at random among all nodes in the network [21, 38]. Such operation induces load on multiple nodes; to account for it, let us assume that when each node selects one random node ID, this induces a per-node load of $(\log n)$. We call such networks *uar-addressable*.

Distributed triangulation Predating the *GNP* algorithm, in the networking literature there were the IDMaps [9] of Francis et al., and several other beacon-based approaches [17, 13, 20] that used the triangle inequality to infer the distances that have not been measured. In particular, in [17, 13] the uv-distance d_{uv} is estimated by $\min(d_{ub} + d_{vb})$, where the minimum is taken over all beacons b.

With this motivation in mind we define a *triangulation* of order k as a labeling of the nodes such that a label of a given node u consists of upper and lower bounds on distances from u to each node in a set S_u of at most k other nodes; for each $b \in S_u$ we denote these bounds by D^+_{ub} and D^-_{ub} respectively [19]. Then any two nodes uv can exchange their labels and use the triangle inequality to upper-bound the uv-distance by

$D^+_{uv} = \min(D^+_{ub} + D^+_{vb})$, and lower-bound it by $D^-_{uv} = \max(D^-_{ub} - D^+_{vb}, D^-_{vb} - D^+_{ub})$, where the $\max$ and $\min$ are taken over all $b \in S_u \cap S_v$. An (ϵ, δ)-*triangulation* is a triangulation such that $D^+_{uv}/D^-_{uv} \leq 1 + \delta$ for all but an ϵ-fraction of node pairs uv. Note that either bound can be seen as a $(1+\delta)$-approximate estimate on the uv-distance, and, moreover, these bounds provide a ”quality certificate” for the estimate.

An (ϵ, δ)-triangulation of an s-doubling metric can be achieved if each node measures distances to $f(\epsilon, \delta, s)$ beacons selected in advance uniformly at random in the network [19]. Moreover, for a uar-addressable network the same paper obtains such triangulation by a fully distributed algorithm with a per-node load at most $f(\epsilon, \delta, s)(\log n)^{O(\log s)}$. Actually, this algorithm provides somewhat stronger guarantees: for each node u the desired triangulation property holds for at least $(1-\epsilon)n$ node pairs uv; we call it a *strong* (ϵ, δ)-triangulation.

Our contributions: we improve the per-node load for a strong (ϵ, δ)-triangulation to $f(\epsilon, \delta, s)(\log n)^4$.

Distance labeling We extend our techniques to obtain approximate distance labeling schemes [10] for doubling metrics. Specifically, for any fixed $\delta > 0$ we obtain a $(0, \delta)$-triangulation of order $O(\log^2 n)$ (which is of independent interest) and convert it to a $(1+\delta)$-approximate distance labeling scheme with $O(\log^2 n)(\log n + \log\log \Delta)$ bits per label, where Δ is the aspect ratio of the metric.[2] We show that it is the best possible dependence on Δ. Since Δ can be arbitrarily large with respect to n, this improves over the labeling scheme of Talwar [36] that uses $O(\log \Delta)$ bits per label. Moreover, in our labeling scheme, unlike the one in [36], given the labels for u and v we can not only estimate the uv-distance but also verify the quality of this estimate.

Our techniques In a fully distributed triangulation algorithm each node u only measures distances to a small set of other nodes, called the *neighbours* of u. In [19] these neighbours are simply selected uniformly at random in the entire network, whereas in this paper the neighbour selection is much more elaborate and, in fact, is the key ingredient of our algorithm. In particular, we make sure that in any ball of sufficiently large cardinality and radius r any two points are connected by a neighbour-to-neighbour path that has at most $O(\log n)$ hops and metric length $O(r)$.

Since the set of neighbor pairs can be seen as an overlay network, our construction is similar in spirit to the overlay topologies constructed for locality-aware distributed hash tables and distributed nearest neighbor selection (e.g. [34, 16, 15, 41]), most notably to the topology constructed in [39]. However, our construction is quite different on the technical level since it is designed to yield provable guarantees on doubling metrics, and is tailored to the specific problem of triangulation.

After the neighbours are selected, some nodes elect themselves as virtual beacons and propagate this information using a neighbour-to-neighbour gossiping. The gossiping protocol ensures that *without any new distance measurements* each node gets bounds on distances to beacons that are sufficient to simulate a beacon-based triangulation of the desired quality. This protocol is more complicated than the one in [19]; its performance relies on the ”quality” of the set of neighbour pairs produced by our algorithm.

To extend our triangulation to an embedding we simulate a beacon-based algorithm which builds on the techniques of Bourgain [3] and Linial et al. [26]. The analysis is considerably more difficult since instead of the actual distance function we use the upper bound D^+ from the triangulation, which is not necessarily a metric. In particular, in our proof D^+ cannot be replaced by an arbitrary function that approximately obeys the triangle inequality: it will be essential that D^+ is close to the specific metric. Moreover, our embedding algorithm has to use the same set of virtual beacons as our triangulation algorithm.

[2] The conference version of this paper erroneously claimed $O(\log^2 n)(\log\log \Delta)$ bits per label.

Extensions and related work For infinite metrics the notion of ϵ-relaxed embedding is not well-defined; we redefine it with respect to an arbitrary measure μ and call it an (ϵ,μ)-*relaxed embedding* (so that an ϵ-relaxed embedding is (ϵ,μ)-relaxed with respect to a uniform metric μ). We show that for any infinite complete doubling metric space with an arbitrary measure μ there exists an (ϵ,μ)-relaxed embedding into any L_p, $p \geq 1$ with finite dimension and distortion. We also obtain similar guarantees for triangulations.

We have already discussed the connection of our work to distance labeling. It is also related to the work on property testing in metric spaces (see [35] for a survey); however, this work considers problems quite different from what we study here, and makes use of different sampling models and objective functions. Finally, our use of triangulation corresponds to the notion of *triangle inequality bounds smoothing* in distance geometry [6], but our setting is different.

Preliminaries Denote the uv-distance by d_{uv}. Let n be the number of nodes. Call a set of node pairs a ϵ-*dense* if for each node u it contains at least $(1-\epsilon)n$ pairs uv. Let $B_u(r)$ be a ball of radius r around u; by default all balls are closed. Let $r_u(\epsilon)$ be the smallest radius r such that $B_u(r)$ contains at least ϵn points. For every $x>0$ denote the set $[x] := \{0,1,2\ldots\lceil x\rceil -1\}$.

Let n be the cardinality of V, and let σ_{unif} be the uniform distribution on V. Say a distribution τ on V is *near-uniform* if $\|\sigma_{\text{unif}} - \tau\|_\infty \leq \frac{1}{2n}$. We can define near-uniform distributions on any given subset of nodes in a similar fashion.

Some of our guarantees are *with high probability*, which in this paper means that the failure probability is at most $1/n^c$, for a sufficiently high constant c.

Organization of the paper. in Sections 2 we present our triangulation algorithm; we extend it to embedding in Section 3. The neighbour selection scheme is deferred to Section 4. Extentions to infinite metrics and distance labeling are described in Sections 5 and 6, respectively. Some relevant facts on tail inequalities and expander graphs are placed in Appendix A and Appendix B, respectively. A useful theorem on random node selection in a network is stated in Appendix C.2.

2 Fully distributed triangulation

Throughout the paper we will assume that the distance function is a metric. For simplicity we define a per-node load in terms of computation, communication and storage; communication includes pings used for latency measurements. Specifically, say a distributed algorithm imposes load k on a given node if during the execution this node stores, sends and receives at most k bits, and computes for at most k cycles.

We start with a fully distributed algorithm for strong (ϵ,δ)-triangulation.

Theorem 2.1. *Consider a uar-addressable network with an s-doubling distance metric of aspect ratio Δ. There is a fully distributed algorithm that for any given (ϵ,δ,s) with high probability constructs a strong (ϵ,δ)-triangulation of order at most $O(x\log x)$, $x=\frac{1}{\epsilon}s^{O(\log 1/\delta)}$. For this algorithm, the per-node load is at most $O(x\log^2 n + T)$, where $T = s^{O(1)}\log^6 n$. The total running time is $O(x\log^2 n + T\log\Delta)$.*

Each node u stores the addresses of, and distances to (a small number of) other nodes, which are called the *neighbours* of u. A pair of neighbours can be treated as an undirected edge. We call a path *neighbour-only* if each edge in the path corresponds to a pair of neighbours. A path is called r-*telescoping* if its i-th hop has length at most $2r/2^i$ (so that the total metric length of the path is at most $2r$). A uv-path is called (r,k)-*zooming* if for some intermediate node w the subpaths uw and vw are r-telescoping and consist of at most k hops each. The set E of edges is an ϵ-*frame* if for any ball of size at least ϵn and radius r any two points in this ball are connected by a $(3r,\log n)$-zooming path in E. The crux of our algorithm is a fully distributed way to select neighbours so that the neighbour pairs form an ϵ-frame.

Theorem 2.2. *Consider a uar-addressable network with an s-doubling distance metric of aspect ratio Δ. There is fully distributed algorithm that for any $\epsilon > 0$ selects neighbours so that with high probability the set of pairs of neighbours is an ϵ-frame, and each node has at most $s^{O(1)}\, O(\log^2 n + \frac{1}{\epsilon}\log n)$ neighbours. The per-node load is most $O(\frac{1}{\epsilon}\, s^{O(1)} \log^2 n + T)$, where $T = s^{O(1)} \log^6 n$. The total running time is at most $O(\frac{1}{\epsilon}\, s^{O(1)} \log^2 n + T \log \Delta)$.*

We defer the proof of this theorem to Section 4 and proceed with the rest of the algorithm. Once the neighbours are selected, some nodes elect themselves as beacons, and use broadcasting to announce themselves to the network. This is our basic broadcasting protocol.

Lemma 2.3. *There is a broadcasting protocol such that any node u can broadcast a message M_{urk} which reaches a given node via an (r,k)-zooming neighbour-only path whenever such path exists. During such broadcast every node stores $O(k)$ bits, and every pair of neighbours exchanges at most $O(k)$ messages.*
Proof: First let's specify the protocol. Node u sends a message $M_{urk}(1)$ to all neighbours within distance r. If some node $v \neq u$ receives a message $M_{urk}(i)$, it does the following:

- If $i > 0$ and v does not store any $M_{urk}(\cdot)$ then for each j, $0 < j \leq k$ it forwards $M_{urk}(1-j)$ to all neighbours that lie within distance $r/2^j$.
- If $0 \leq |i| \leq k$ and v does not store $M_{urk}(i)$, v saves it and then, if the inequality is strict, v forwards $M_{urk}(i+1)$ to all neighbours that lie within distance $r/2^{|i|}$ from v.

This completes the specification. It is easy to see that the storage and the number of messages exchanged are as required, and that the messages propagate along (r,k)-zooming paths. If u and v are connected by such a path, then by definition there exists an r-telescoping uw path P_1 and an r-telescoping vw path P_2, of length at most k each. Then using induction on i one shows that the i-th node of P_1 will receive $M_{urk}(1+i)$, and then similarly that the i-th node of P_2 will receive $M_{urk}(1-i)$. □

Recall that we are given arbitrary (ϵ, δ) and we'd like to construct a strong (ϵ, δ)-triangulation. Letting $\epsilon' = \epsilon\delta^{2\log s}/(2s)$, we select neighbours using the algorithm in Theorem 2.2 so that the set of neighbour pairs is an $\epsilon'/2$-frame. In particular, every node u will have $N = \Theta(\log n)/\epsilon'$ neighbours selected uar in the entire network, call them the *designated neighbours of* u. Then every node elects itself as a *beacon* independently at random with probability N_{beac}/n, where N_{beac} is the desired number of beacons which we will specify later. Once self-elected, each beacon b announces itself by broadcasting the message $M(b, r'_b, \log n)$ as in Lemma 2.3, where r'_b is the $(\epsilon' N/\sqrt{2})$-th smallest distance from b to its designated neighbours.

Whenever a node u receives a message M_{brk} from a beacon b via a path of length x, this message certifies that b lies within distance x from u; accordingly, node u updates D^+_{ub} if necessary. Moreover, it sends a message $M'(b,x)$ to every node v such that u is a designated neighbour of v. For node v such message certifies that d_{vb} lies within $d_{uv} \pm x$; accordingly, v updates $D^{\pm}_{vb}$ if necessary. This completes the algorithm.

During the broadcasting phase of this algorithm, every pair of neighbours exchanges at most $O(\log n)$ messages per each beacon. Letting $r_u = r_u(\epsilon')$, the quality of the bounds $D^{\pm}_{ub}$ is as follows.

Claim 2.4. *Whp every node u forms, for every beacon b, bounds $D^{\pm}_{ub}$ that lie within $d_{ub} \pm O(r_b)$.*
Proof: Fix a node u. Since the ball $B_b(r_b(\epsilon'/2))$ has size at least $\epsilon' n/2$, with high probability u has a neighbour in it, call it v. Since by Chernoff bounds with high probability $r_b(\epsilon'/2) \leq r'_b \leq r_b$, by definition of ϵ-frame there exists an $(r'_b, \log n)$-zooming bv-path. By Lemma 2.3 a message from b will reach v by traversing such a path (call it ρ), and then v will send $M'(b,x)$ directly to u, where x is the metric length of ρ. Upon receiving this message, u will bound d_{ub} by $d_{uv} \pm x$. The claim follows since x is at most $O(r_b)$, and d_{uv} lies within $d_{ub} \pm r_b$. □

Let $N_{\text{beac}} = (\log n)/\epsilon'$. In [19] it was proved that for an ϵ-dense set of node pairs uv there is a ball B of radius δd_{uv} either around u or around v which has cardinality at least $\epsilon' n$; say this is a ball around u. In particular, with high probability this ball contains a beacon, call it b. Then $B_b(2r_u)$ contains B, hence has size at least ϵn, so $r_b \leq 2r_u \leq 2\delta d_{uv}$. Therefore (omitting some details) by Claim 2.4 nodes u and v can use $d^{\pm}_{ub}$ and $d^{\pm}_{vb}$ to bound d_{uv} by $d_{uv} \pm O(\delta d_{uv})$ as desired.

This gives an strong (ϵ, δ)-triangulation of order $\Theta(N_{\text{beac}})$. With some more work we can make N_{beac} independent of n: set $N_{\text{beac}} = \Theta(s^3/\epsilon') \log(s/\epsilon')$ and use Lemma 5.2 to guarantee that with high probability every ball $B_u(6r_u)$ contains a beacon. Then, as before, for an ϵ-dense set of node pairs uv, nodes u and v can bound d_{uv} by $d_{uv} \pm O(\delta d_{uv})$ as desired. This completes the proof of Theorem 2.1.

3 Fully distributed embeddings

In this section we extend the algorithm from Section 2 to a fully distributed algorithm that computes an ϵ-relaxed embedding into any L_p, $p \geq 1$ with dimension and distortion that depend only on (ϵ, s), not on n.

Theorem 3.1. *For a uar-addressable network with an s-doubling distance metric there is a fully distributed algorithm that for any given (ϵ, s) with high probability constructs an ϵ-relaxed embedding into any L_p, $p \geq 1$ with distortion $O(\log k)$ and dimension $O(k \log k)$, where $k = (\frac{s}{\epsilon})^{O(\log\log(s/\epsilon))}$. In this algorithm the per-node load and the total completion time are at most $O(k^2 \log^3 n)$.*

Fix (ϵ, s), and fix a constant c to be specified later. Note that there exists $k = (\frac{s}{\epsilon})^{O(\log\log(s/\epsilon))}$ such that for $\delta = c/\log k$ the algorithm in Section 2 with high probability computes an (ϵ, δ)-triangulation with at most k and at least $\Omega(k)$ beacons; the proof is a simple but tedious computation which we will omit.

The high-level algorithm is simple. First we let $\delta = c/\log k$ and compute an (ϵ, δ)-triangulation using the algorithm from Section 2. Then the beacons measure distances to one another and broadcast them to the entire network using a uniform gossip [33]; in this phase each beacon broadcasts one message of size $O(k)$, the total per-node load being at most $O(k^2 \log n)$. Upon receiving this information nodes update the bounds D^+ on their distances to beacons accordingly, by running a shortest-paths algorithm. (Note that in this step D^+ can only decrease, but not below the true distance; in particular, Claim 2.4 still holds.) Finally, run the embedding algorithm in Theorem 3.3 of [19] *on the same beacon set*, using the upper bounds D^+ instead of the latent true distances to the beacons.

Our proof outline follows that of Theorem 3.3 of [19]: first we bound the distortion on node-to-beacon distances, then use those to bound distances between other node pairs. The details are very different, though, since we are embedding D^+ which is not necessarily a metric. In particular, in our proof D^+ is more than just a function that approximately obeys the triangle inequality: it it will be essential that D^+ is close to a specific metric, as expressed by Lemma 2.4. we will use this lemma to reason about the embedded distances to beacons, which is why we use the same set of beacons for both triangulation and embedding.

For completeness let's restate the embedding algorithm (which is adapted from [19] and is closely related to the algorithms of Bourgain [3] and Linial et al. [26]). Let S_{beac} be the beacon set from the (ϵ, δ)-triangulation; for simplicity assume there are exactly k beacons. For each $i \in [\log k]$ choose $\Theta(k)$ random subsets of S_{beac} of size 2^i each; let S_{ij} be the j-th of those. These subsets are broadcasted to the entire network using a uniform gossip [33]: one message of size $O(k^2)$ is broadcasted, incurring a per-node load at most $O(k^2 \log n)$. Then every node u embeds itself into L_p so that each dimension ij is defined as $D^+(u, S_{ij})/\Theta(k)$, where $D^+(u, S)$ is the smallest D^+_{uv} such that $v \in S$.

Note the differences with the algorithm of [26]. Firstly, the beacon sets S_{ij} are sampled from S_{beac}, not from the entire network. Essentially, we embed S_{beac} using the algorithm of [26], and then embed the rest of the nodes using the same beacon sets. While embedding S_{beac} we use $\Theta(k)$ beacon sets of each size scale, not $\Theta(\log k)$ as [26] does. This is necessary to guarantee the following claim from [19]:

Claim 3.2. *with high probability for any $i \in [\log k]$ and any pair of disjoint subsets $S, S' \subset S_{beac}$ of size at least $k/2^i$ and at most $2k/2^i$, respectively, it is the case that at least $\Omega(k)$ sets S_{ij} hit S and miss S'.*

Then, letting d'_{uv} be the embedded uv-distance, we can bound the embedded node-to-beacon distances:

Claim 3.3. *Whp for each node u and every beacon b we have $d_{ub} \leq d'_{ub} \leq O(\log k) D^+_{ub}$.*

Now, as we saw in the proof of Theorem 2.1, with high probability for an ϵ-dense set of node pairs uv there is a beacon b within distance $6\delta d_{uv}$ from u or v (say, from v) such that $r_b \leq 12\delta d_{uv}$. Therefore by Claims 2.4 and 3.3 we have

$$(1-6\delta) d_{uv} \leq d'_{ub} \leq O(\log k) d_{uv}$$

and $d'_{vb} \leq O(\log k)\delta d_{uv}$, so it follows that

$$d_{uv}/2 \leq d'_{ub} - d'_{vb} \leq d'_{uv} \leq d'_{ub} + d'_{vb} \leq O(\log k) d_{uv}$$

as long as the constant c that defines δ is small enough.

To complete the proof of Theorem 3.1 it remains to prove Claim 3.3. For simplicity consider the case $p = 1$ first. For a node set S and any pair uv of nodes define $D^+_{uv}(S) = |D^+(u,S) - D^+(v,S)|$. Then the embedded uv-distance d'_{uv} is equal to the sum $\sum D^+_{uv}(S_{ij})$ over all beacon sets S_{ij}. In order to establish the desired upper bound on d'_{ub} it suffices to prove that if u is a node, b is a beacon and S is a set of beacons then $D^+_{ub}(S) \leq 2D^+_{ub}$. It will follow by a standard argument from the following claim: $|D^+_{ub'} - D^+_{bb'}| \leq 2D^+_{ub}$ for any two beacons b, b'.

Let's prove this claim. Consider the beacon b_u that is closest to u with respect to D^+; let $x = D^+(u, b_u)$ and $y = d(b', b_u)$. The beacons measure distances to each other, so $D^+_{bb'} = d_{bb'}$. Node u has updated $D^+_{ub'}$ according to these measurements, so it is at most $x + y$; obviously, it is at least $d_{ub'}$, which is lower-bounded by $y - x$. Therefore $|D^+_{ub'} - y| \leq x$, so, completing the proof,

$$|D^+_{ub'} - D^+_{bb'}| \leq |y - d_{bb'}| + |D^+_{ub'} - y| \leq d(b, b_u) + x \leq d_{bu} + 2x \leq 3D^+_{ub}.$$

It remains to establish the lower bound in Claim 3.3, which we will accomplish by a version of Bourgain's telescoping sum argument. Let $S_u(r)$ be the set of beacons b such that $D^+_{ub} \leq r$. For a fixed node u and beacon b, let $\rho_i = \min(\rho_u(i), \rho_v(i), d_{ub}/2)$, where $\rho_u(i)$ is the smallest r such that $S_u(r)$ contains at least $k/2^i$ beacons.

We claim that for each given i the sum $X_i = \sum_j D^+_{ub}(S_{ij})$ is at least $\Omega(k)(\rho_{i-1} - \rho_i)$. Indeed, fix i and without loss of generality assume that $\rho_u(i) \leq \rho_b(i)$. Note that the sets $S = S_u(\rho_i)$ and the interior S' of $S_b(\rho_{i-1})$ are disjoint since if a node v belongs to both S and S' then

$$d_{ub} \leq d_{uv} + d_{bv} \leq D^+_{uv} + D^+_{bv} < \rho_i + \rho_{i-1} \leq d_{ub},$$

contradiction. Therefore by Claim 3.2 with high probability for each i at least $\Omega(k)$ sets S_{ij} hit S and miss S', thus contributing at least $\rho_{i-1} - \rho_i$ each to X_i. This proves the claim.

Let $t = \lfloor \log k \rfloor$ and note that by definition $\rho_b(t) = 0$ (since $S_b(0)$ contains at one beacon, namely b itself), so $\rho_t = 0$. Summing up the X_i's we get $d'_{ub} \geq \Omega(k)(\rho_1 - \rho_t) = \Omega(k) d_{ub}$ as desired, as long as $\rho_1 \geq d_{ub}/4$. Now suppose $\rho_1 < d_{ub}/4$ and assume that $\rho_u(1) < \rho_b(1)$ (the case $\rho_u(1) \geq \rho_b(1)$ is treated similarly). Then the sets $S = S_u(d_{ub}/4)$ and $S' = S_{\text{beac}} \setminus S$ are disjoint and have size at least $n/2$ and at most $n/2$, respectively. Therefore by Claim 3.2 with high probability at least $\Omega(k)$ sets S_{1j} hit S and miss S', thus contributing at least $D^+_{ub}/2 = \Omega(d_{ub})$ each to X_i, so that $d'_{ub} \geq \Omega(k) d_{ub}$ as desired. This completes the proof of Claim 3.3 for $p = 1$. We can extend it to general $p \geq 1$ following [26]; we omit the details. This completes the proof of Theorem 3.1.

4 Finding good neighbours

In this section we discuss our neighbour selection algorithm and prove Theorem 2.2. We start with an overview. On the very basic level our algorithm will only guarantee properties that are local to every given node; we call them *invariants*. To prove that we indeed construct an ϵ-frame we will need to "connect" these invariants via certain global structures. First for every ball B we establish such a structure – essentially, a nesting sequence of balls with lots of nice properties. Then we will define the invariants and show that if they hold and B is large enough then every node in every ball in the sequence will have a neighbour in the next ball of the sequence; this (denoting the radius of B by r) will guarantee the required $(3r, \log n)$-zooming neighbour-only path connecting every pair of nodes in B. Finally, we will specify the algorithm and prove that it satisfies the invariants.

Now we will describe the proof in detail. Assume the metric is s-doubling.

Definition 4.1. A subset of a ball $B_u(r)$ that contains the corresponding open ball and at least one boundary point is called a *fuzzy ball*. For any such ball B and a scalar $\alpha > 0$, define a shorthand $\alpha B := B_u(\alpha r)$. A fuzzy ball B is called (α, β)-*padded* $|\alpha B| \leq s^{\beta}|B|$.

Lemma 4.2. *For any fuzzy ball B there exists a $(16, 5)$-padded fuzzy ball B^* such that*

$$32B^* \subset 2\tfrac{1}{16}\, B \;\; \textit{and} \;\; |B^*| = \left\lceil\, |B|/8s^5 \,\right\rceil. \tag{1}$$

Proof. Suppose not. Let $x = |B|$ and let u be the center of B. Consider the smallest ball around u that has cardinality at least $x/8$, call it $B_u(r)$. By the doubling property of the metric, $B_u(r)$ can be covered by s^5 balls of radius $r/32$. At least one of them, centered at (say) v, has cardinality at least $x/(2s^5)$; note that $d_{uv} \leq \frac{33}{32}r$. A fuzzy ball around v of cardinality exactly $\lceil x/(8s^5) \rceil$ (call it B') has radius at most $r/32$; obviously, $B_v(r)$ lies within $B_u(2\frac{1}{16}r)$. Therefore $B_v(r/2)$ has cardinality at least $x/8$, since otherwise B' is $(16, 5)$-padded, contradiction.

Iterating this argument i times, we come up with a node v such that $d_{uv} \leq \frac{33}{32}r(2 - 2^{-i})$ and either there is a $(16, 5)$-padded fuzzy $(v, r/2^{i+4})$-ball which is zooming with respect to B, or $B_v(r/2^i)$ has cardinality at least $x/8$. However, the latter cannot be true when i is large enough, e.g. when $r/2^i$ is less than the minimal distance. □

If two fuzzy balls B and B^* satisfy (1), let us say that B^* *is zooming with respect to* B. The constant 8 in (1) is tailored to the forthcoming arguments.

A sequence of fuzzy balls is called *zooming* if each ball is zooming with respect to the previous one, and the last ball consists of only one node. By the above lemma, for any ball B there is a zooming sequence $\sigma(B)$ of $(16, 5)$-padded fuzzy balls starting with a ball B^* which is zooming with respect to B. Every node u will select enough neighbours independently and uniformly at random from the entire network so that if B contains at least ϵn nodes then with high probability u has a neighbour in B^*. Our neighbour selection algorithm will make sure that for every such B, every node in every ball in the sequence $\sigma(B)$ has a neighbour in the next ball of the sequence. Since the last ball in $\sigma(B)$ consists of a single node w, for any nodes $u, v \in B$ this gives (letting r be the radius of B) a $2\frac{1}{16}r$-telescoping uw- and vw-paths of length at most $\log n$ each, as required.

We distinguish *out-neighbors* and *in-neighbors*: node v is an out-neighbor of node u if and only if v is an in-neighbor of u. Two nodes are neighbors if and only if one is an out-neighbor of another. Each node u selects $s^5 k/\epsilon$ *global* out-neighbours independently and uniformly at random from the entire network, incurring a per-node load of $O(\frac{1}{\epsilon}\, s^{O(1)} \log^2 n)$. Moreover, node u will have *local* out-neighbors that are distributed on smaller balls around u. Let us say that node u is a global/local in-neighbor of node v if v is a global/local out-neighbor of u. Note that by Chernoff Bounds with high probability every node has at most $O(\frac{1}{\epsilon} s^{O(1)} \log n)$ global in-neighbours.

Local neighbors of node u are partitioned into some number $l_u \leq \log n$ *levels* numbered $0, 1, 2, \ldots$, so that for each $i \in [l_u]$ there are exactly k level-i neighbours; here $k = c\, s^{15} \log n$, where the constant c is large enough to use the Union Bound where appropriate. All level-i neighbors of node u are contained in some ball $B_{ui} := B_u(\rho_{ui})$, where ρ_{ui} is the *characteristic radius* specified by the algorithm. By abuse of notation we will also call B_{ui} the i-th level of node u. Level-0 neighbors of each node u are selected independently and uniformly at random from the entire network; accordingly, we set $\rho_{u0} = \Delta$. We will maintain the following invariant:

Invariant 1. *For every level B_{ui}, ball B_{ui} is $(2,5)$-padded and $|B_{ui}| \leq \frac{1}{2}|B_{(u,i-1)}|$.*

Our construction is highly randomized. To analyze it, we consider the probability space induced by the random choices in the construction. In particular, intuitively we would like all level-i neighbours of a given node u to be distributed independently and uniformly (or near-uniformly) in the corresponding ball B_{ui}. What we actually ensure is a somewhat weaker condition, which however suffices for our purposes.

Essentially, we condition on levels with larger characteristic radius. We treat each level-i neighbor of a given node u as a random variable distributed on the corresponding ball B_{ui}. For each $j \in [\log \Delta]$ let us say that the *stage-j levels* are those with characteristic radii in the interval $[\Delta/2^j,\, 2 \times \Delta/2^j)$. Let $\mathcal{F}_j^{\text{nbrs}}$ be the family of all random variables corresponding to neighbors in stage-j levels. For convenience let us define $\mathcal{F}_{-1}^{\text{nbrs}} = \emptyset$.

Invariant 2. *For each $j \in [\log \Delta]$, the family $\mathcal{F}_j^{nbrs}$ is conditionally independent given $\cup_{l<j} \mathcal{F}_l^{nbrs}$. Moreover, if random variable $X \in \mathcal{F}_j^{nbrs}$ corresponds to a level-i neighbor of node u, then given $\cup_{l<j} \mathcal{F}_l^{nbrs}$ this X has a near-uniform distribution on the corresponding ball B_{ui}.*

Let us bound the number of local in-neighbors using the above invariants.

Claim 4.3. *If Invariants (1-2) hold then with high probability every node has at most $O(s^{O(1)} \log^2 n)$ local in-neighbours.*

Proof. Fix a node u and define the family of levels

$$\mathcal{F}_u := \{\text{all levels } B_{wi'} \,:\, i' \geq 0, w \in V, u \in B_{wi'}\}.$$

Let us treat the levels in $\mathcal{F}_u$ as balls. For each $i \in [\log n]$ let $\mathcal{F}_{ui}$ be the family of balls from F_u that have cardinality in the interval $(2^i, 2^{i+1}]$. Let B^* be a ball in $\mathcal{F}_{ui}$ with the largest radius. Then $2B^*$ contains the centers of all balls in $\mathcal{F}_{ui}$. Since B^* is $(2,5)$-padded and $\mathcal{F}_{ui}$ contains at most one ball with a given center, it follows that $|\mathcal{F}_{ui}| \leq s^5\, 2^{i+1}$. Therefore

$$\sum_{B \in \mathcal{F}_u} 1/|B| = \sum_i \sum_{B \in \mathcal{F}_{ui}} 1/|B| \leq \sum_i |\mathcal{F}_{ui}|/2^i \leq 2s^5 \log n. \tag{2}$$

So the expected number of non-global reverse neighbours of u is at most $2ks^5 \log n$. Using generalized Chernoff bounds (Theorem A.2) we get the desired high probability result. □

While Invariants (1-2) specify the properties of the levels once they exist, we also need guarantees as to when they actually exist. To formulate those, we need a few definitions.

Definition 4.4. Call level B_{ui} *healthy* if for any $v \in B_{ui}$ the smallest ball B_{vj} such that $\rho_{vj} \geq \min(2\rho_{ui},\, \Delta)$ has cardinality at most $8s^{10}$ times that of B_{ui}.

Definition 4.5. Say a ball B centered at node u is *friendly* for a ball B' if $x/8 \leq |B| \leq x$, where x is the cardinality of the smallest ball around u that contains B'.

Invariant 3. *If level B_{ui} is healthy then for any $(2,5)$-padded ball $B \subset B_{ui}$ around u of cardinality at least $1/(2s^{10})$ that of B_{ui} there is a friendly level $B_{(u,\cdot)}$.*

Before fully specifying the algorithm, we will show that if the invariants are satisfied then the required neighbour-only paths exist. As we have seen, it suffices to prove that with high probability for any ball B_0 of size at least ϵn, every node in every fuzzy ball in the sequence $\sigma(B_0)$ has a neighbour in the next fuzzy ball. Note that although there can be exponentially many fuzzy balls, only a polynomial number of them is used in sequences $\sigma(B_0)$, so we can achieve a high-probability result just by increasing the constant in the definition of k.

Let $\sigma(B_0) = (B_1^*, B_2^*, B_3^*, \ldots)$ and let $B_i = 2\frac{1}{16}\, B_i^*$ for all i. Note that each ball B_i is $(7,5)$-padded and the sequence $(B_1, B_2, B_3, \ldots)$ is nesting; the latter can be seen by induction on the length of the sequence.

Claim 4.6. *If Invariants (1-3) hold then with high probability for each i and every $v \in B_i$ there is a healthy level $B_{(v,\cdot)}$ which is friendly for B_i.*

Proof: we will use induction on i. For the base case consider some $v \in B_1$. By Invariant (3) there exists a level $B_{(v,\cdot)}$ friendly for B_1. This level is healthy since B_1 has cardinality at least n_0. For the induction step, assume the claim holds for some B_i, and let $v \in B_{i+1}$. Then by induction hypothesis there is a level B_{vj} which is healthy and friendly for B_i. Note that, letting $x = |B_i^*|$, the cardinality of B_{vj} is at most xs^5.

The smallest ball B around v that contains B_{i+1} is $(2,5)$-padded. (Indeed, letting $B_{i+1} = B_u(\rho)$ and $B = B_v(r)$, we have $r \leq 2\rho$ and therefore $B_v(2r) \subset B_u(5\rho)$ is small enough.) Since $|B| \geq |B_{i+1}| \geq \frac{1}{2}x/s^5$, applying Invariant (3) to B_{vj} shows that if $B \subset B_{vj}$ then (since B_{vj} is healthy) there is a level B_{vl} friendly for B. If $B \not\subset B_{vj}$ then $B_{vj} \subset B$ and (since it is easy to see that $B \subset B_i$) level B_{vj} itself is friendly for B. So there is a level B_{vl}, $l \geq j$ that is friendly for B, and hence friendly for B_{i+1}.

It remains to show that B_{vl} is healthy. Fix $w \in B_{vl}$. Since B_{vl} is contained in B_{i+1} and has cardinality at least $x/(8s^5)$, it suffices to find a level $B_{(w,\cdot)}$ of cardinality at most xs^5 such that $\rho_{wt} \geq 2r$. Since $w \in B_i$, applying induction hypothesis yields a level B_{wt} which is friendly for B_i, hence has cardinality at most xs^5 and at least $x/2$. Now, since $B_{i+1} = B_u(\rho)$ is $(7,5)$-padded and $B_w(2r) \subset B_v(3r) \subset B_u(7\rho)$, it follows that $|B_w(2r)| \leq s^5 |B_{i+1}^*| = x/2 \leq |B_{wt}|$, so $\rho_{wt} \geq 2r$. Therefore B_{wt} is the required level. □

Claim 4.7. *If Invariants (1-3) hold then with high probability in each ball B_i every node $v \in B_i$ has a neighbour in the ball B_{i+1}.*

Proof. Fix $v \in B_i$. Recall that $B_i \subset B_{i-1}$. By Claim 4.6 since $v \in B_{i-1}$ there exists a level B_{vj} which is friendly for B_{i-1}. We claim that $B_{i+1} \subset B_{vj}$.

Indeed, let B be the smallest ball around v that contains B_i. Since $B_{i+1} \subset B_i \subset B$, it suffices to prove that $B \subset B_{vj}$. First, note that $B \subset 2B_i \subset 4\frac{1}{8}\, B_i^*$. Since fuzzy ball B_i^* is $(16,5)$-padded, it follows that $|B| \leq s^5 |B_i^*|$. Since B_i^* is zooming with respect to B_{i-1}, we have $|B_i^*| \leq |B_{i-1}^*|/8s^5$. Since ball B_{vj} is friendly for ball B_{i-1}, we have $|B'|/8 \leq |B_{vj}| \leq |B'|$, where B' is the smallest ball around v containing B_{i-1}. Putting it all together,

$$|B| \leq s^5\, |B_i^*| \leq |B_{i-1}|/8 \leq |B'|/8 \leq |B_{vj}|.$$

Since B and B_{vj} are balls around the same node, it follows that $B \subset B_{vi}$, claim proved.

Moreover, it turns out that B_{i+1} is a sufficiently large subset of B_{vj}:

$$|B_{vj}| \leq |B'| \leq |2\, B_{i-1}| \leq s^5\, |B_{i-1}| \leq O(s^{15})\, |B_{i+1}|.$$

Now by Chernoff Bounds with high probability B_{i+1} contains a level-j neighbour of v. □

4.1 Finding good neighbors: the construction

We specify the neighbour selection algorithm and show that it is load-balanced and satisfies the invariants.

After a given level B_{ui} is constructed, node u eventually *explores* it by calling procedure *Explore*(B_{ui}). This procedure uses the level-i neighbors of node u to find possible smaller characteristic radii, and attempts to construct levels with these candidate characteristic radii via Theorem C.2 with $Q = B_{ui}$. Call a given level B_{ui} *unexplored* if procedure *Explore*(B_{ui}) has not been called yet.

For simplicity assume that the aspect ratio Δ is an exact power of two. The construction proceeds in globally synchronized stages numbered $0, 1, \ldots, \log\Delta$. Recall that the *stage-j levels* are those with characteristic radii in the interval $[\Delta/2^j,\ 2\times\Delta/2^j)$. We maintain the following invariant:

Invariant 4. *Each stage-j level is explored in stage j; no stage-j levels are explored outside stage j.*

Note that by Invariant (4) all stage-j levels are constructed before the end of stage j.

For a given node u and stage j, the pseudocode is quite simple: while there are unexplored stage-j levels, call *Explore*(B_{ui}), where B_{ui} is the unexplored stage-j level with the largest characteristic radius. Note that this pseudocode trivially satisfies Invariant (4).

Now let us specify procedure *Explore*(B_{ui}).

Definition 4.8. For a set S of numbers, let us define a δ-*median* of S as the $\lceil\delta|S|\rceil$-th number in the ascending ordering of S. For $\delta \leq 1$, let $\phi_{ui}(\delta)$ be the δ-median of distances from u to its i-level neighbours. To define a similar quantity $\phi_{ui}(\delta)$ for $\delta > 1$, let us choose the smallest ball B_{uj} such that $\rho_{ui} = \phi_{uj}(\delta')$ for some $\delta' \leq \frac{1}{\delta}$, and define $\phi_{ui}(\delta) = \phi_{uj}(\delta\delta')$.

Among the levels $B_{(u,\cdot)}$ constructed so far, let us choose the one with the smallest characteristic radius, call this radius r_0. Choose δ_0 such that $\phi_{ui}(\delta_0) = r_0$. We consider all $\delta \in [\delta_0/4; s^{-10}]$ such that $\delta = \delta_0/4^l$ for some $l \in \mathbb{N}$. For each such δ we check whether $2\phi_{ui}(\delta) \leq \phi_{ui}(4s^5\delta)$. If this is the case, we call $r := \phi_{ui}(\delta)$ a *candidate radius* and attempt to select neighbors for the *candidate level* with characteristic radius r. Note that by Chernoff Bounds Invariant (1) is satisfied.

We select neighbors for the candidate levels via Theorem C.2, so we need to tailor our setting to this theorem. Firstly, each node partitions its neighbors in every given level into two equal-size groups: *walk-neighbors* and *seed-neighbors*. The former are used for random walks in this theorem, and the latter are used for random seeds.

Let Q be the ball corresponding to level B_{ui}. For each $v \in Q$, consider the levels constructed for node v in stages 1 through j with characteristic radii at least $2r$. Among these levels, let B_v be the one with the smallest characteristic radii. Let G be the directed graph (possibly with loops and multiple edges) induced by the walk-neighbors in levels B_v, $v \in Q$. Specifically, whenever node $v \in Q$ has a walk-neighbor w in level B_v, we add a directed edge (v, w) to the graph G. Let G^* be the undirected version of G.

We will use Theorem C.2 for graph G and subset Q. Note that by Theorem A.2 $\deg(G^*) \leq O(k)$ with high probability. Accordingly, in Theorem C.2 we will take $d = d_Q = O(k)$.

Since for this theorem each node needs to know its G^*-neighbors (which are a subset of its reverse neighbors), each node v contacts all its neighbors so that each neighbor w of v learns (a) that node v is its reverse neighbor, (b) what are the the characteristic radii of all levels of v, and (c) in which level of v node w is a neighbor. To bound the per-node overhead of this operation, recall that by Claim 4.3 each node has at most $O(\frac{1}{\epsilon}s^{O(1)}\log^2 n)$ reverse neighbours. Note that for our purpoces each node $w \in Q$ should not only know all its reverse neighbors, but know which of them are its G^*-neighbors. In other words, for each reverse neighbor v of w we need to know whether node w is one of the neighbors in B_v. However, node w can determine this knowing r and the information that he receives from node v.

By Theorem A.2 if level B_{ui} is healthy then with high probability $\deg(G|Q) \geq 3\log n$ for each $v \in Q$. In this case by Theorem B.3(b) the graph $G^*|Q$ is an expander with high probability. Let $d_0 = 3\log n$.

Moreover, as required in Theorem C.2, the graph $G|Q$ is a (d_0, γ)-quasi-expander, for some constant γ; this is by Theorem B.3(a).

By Theorem A.1 with high probability at least one seed-neighbor in level B_{ui} lies in Q; we choose any such neighbor at random and designate it to be the random seed. Let algorithm $\mathcal{A}(B_{ui})$ be the construction in Theorem C.2 (for graph G and subset Q) with this random seed and $k_0 = \Theta(s^{10}\, k)$. Either algorithm $\mathcal{A}(B_{ui})$ aborts, or node u acquires k_0 nodes selected independently from a near-uniform distribution on Q. In particular, the latter happens if level B_{ui} is healthy, thus satisfying Invariant (3)

Suppose algorithm $\mathcal{A}(B_{ui})$ does not abort. Let $r_1, r_2, \ldots$ be the candidate radii in the decreasing order, and let $r_l = \phi_{ui}(\delta_l)$. Recall that δ_l's are exponentially decreasing and lower-bounded by s^{-10}. We randomly partition the k_0 neighbors returned by $\mathcal{A}(B_{ui})$ into sets of $\Theta(k/\delta_l)$ nodes, one for each candidate level l. In each such set by Theorem A.1 at least k nodes will land in $B_u(r_l)$ with high probability. We select k of those at random as the neighbors in the corresponding level. Note that Invariant (2) is satisfied since graph G depends only on $\cup_{l<j} \mathcal{F}_l^{\text{nbrs}}$.

This completes the description of the procedure *Explore*(B_{ui}).

Load-balancing. Fix node v. Let $Z_v(B_{ui})$ is 1 if node v is visited by algorithm $\mathcal{A}(B_{ui})$, and 0 otherwise; i.e. this is precisely the random variable Z_v from Theorem C.2. Define the family of levels containing v:

$$\mathcal{F} := \{\text{all levels } B_{ui} :\ i \geq 0, u \in V, v \in B_{ui}\}.$$

Recall that by Theorem C.2 for each stage-j level $B \in \mathcal{F}$ we have $Z_v(B) \leq 1$ and

$$E\left(Z_v(B) |\cup_{l<j} \mathcal{F}_l\right) \leq \mu_B := O(k_0 t/|B|),$$

where $t = O(k^2 \log n)$. Moreover, by (2) we have

$$\mu := \sum_{B\in\mathcal{F}} \mu_B = O(k_0 t) \sum_{B\in\mathcal{F}} 1/|B| \leq O(k_0\, t s^5 \log n) = O(s^{O(1)} \log^4 n).$$

By Theorem A.2 with high probability $Z_v := \sum_{B\in\mathcal{F}} Z_v(B) \leq O(\mu)$.

Let us consider the load induced on a given node u by the construction of local neighbors. Let us partition this load into *native load* induced by algorithms $\mathcal{A}_u(\cdot)$, and *foreign load* induced by algorithms $\mathcal{A}_v(\cdot)$, $v \neq u$. Native load is at most $O(k_0 d t)$ for each of at most $\log n$ instances of algorithm $\mathcal{A}_u(\cdot)$, for a total of $O(s^{O(1)} \log^5 n)$. To bound the foreign load, let S be the set of all local neighbors of u; recall that $|S|$ is upper-bounded by Claim 4.3. By Theorem C.2 the foreign load on u is

$$O\left(\textstyle\sum_{v\in S} Z_v\right) \leq O(\mu|S|) \leq O(s^{O(1)} \log^6 n).$$

Total running time. Note that in a given stage each message can be processed (in, essentially, a unit time) as soon as it is received. The only possible delay occurs due to contention, when a given node receives messages faster than it can process them. However, we assume that our construction happens in the backgroung, at a sufficiently slow pace so that such contention is negligeable. Specifically, we assume that before sending each message we wait for a random time interval, drawn from (say) a Gaussian with mean T. Then the total running time in a given stage is upper-bounded by T times the per-node load.

5 Triangulation and embedding for infinite doubling metrics

The beacon-based triangulations and embeddings of [19] are for finite metrics; ϵ-dense sets are defined with respect to the uniform measure. Here we extend them to infinite metrics and arbitrary probability measures.

Specifically, given a probability measure μ we define (ϵ, δ, μ)-*triangulation* and (ϵ, μ)-*relaxed embedding*; here the desired properties hold for all node pairs uv, $v \in S_u$ where $\mu(S_u) \geq 1 - \epsilon$. We aim for (ϵ, δ, μ)-triangulations of finite order, and (ϵ, μ)-relaxed embeddings with finite distortion and dimension.

Theorem 5.1. *Consider a (possibly infinite) complete s-doubling metric space (V, d). Then for any probability measure μ on V and any positive parameters ϵ, δ there exists:*

(a) an (ϵ, δ, μ)-triangulation of order $k = O(s^{10})(\frac{1}{\epsilon})(\frac{1}{\delta})^{2\log s}$, and

(b) an (ϵ, μ)-relaxed embedding into ℓ_p^k, $p \geq 1$ with distortion $O(\log k)$, where $k = (\frac{s}{\epsilon})^{O(\log\log(s/\epsilon))}$.

Suppose we can take independent random samples from distribution μ. Then after taking $O(k)$ such samples we can construct respectively node labels in (a) or node coordinates in (b), in time $O(k)$ per node.

Say S is an (ϵ, μ)-hitting set for the metric if it hits a ball of radius $6r_u(\epsilon)$ around every node u, where $r_u(\epsilon)$ is the radius of the smallest ball around u of measure at least ϵ. The crux of the proof of Theorem 5.1 is a lemma on the existence of finite (and small!) (ϵ, μ)-hitting sets, which we derive using a suitable analogue of Lemma 4.2.

Lemma 5.2. *Consider a complete s-doubling metric. Then for any probability measure μ and any $\epsilon > 0$ there is an (ϵ, μ)-hitting set of size $k = 2s^3/\epsilon$. Moreover, with probability at least $1 - \gamma$ a set of $k \log(k/\gamma)$ points sampled independently at random with respect to μ is (ϵ, μ)-hitting.*

Proof: Let $r_u = r_u(\epsilon)$. First we claim that for every node u either there is a node $b_u \in B_u(2r_u)$ of measure at least $\epsilon/2$, or there is a ball $B_u \subset B_u(3r_u)$ of measure at least $\epsilon/(2s^3)$ such that the ball B'_u with the same center and four times the radius has measure at most $\epsilon/2$. Indeed, following the proof of Lemma 4.2 we can show that for a fixed u either such B_u exists, or there is an infinite nesting sequence of balls $(S_1 \supset S_2 \supset S_3 \supset \ldots)$ of exponentially decreasing radii ρ_i and measure at least $\epsilon/2$ each, starting with $B_u(2r_u)$. Consider the sequence of centers of these balls. This is a Cauchy sequence, so (since the metric is complete) it has a limit, call it v. Then v lies in each S_i, so for each i the ball $B_v(2\rho_i)$ contains S_i, hence has measure at least $\epsilon/2$. Therefore v has measure at least $\epsilon/2$ as required. Claim proved.

For convenience define $B_u = \{b_u\}$ if such b_u exists. Let $\mathcal{F}$ be a maximal collection of disjoint balls B_u. We claim that for every node v some ball $B_u \in \mathcal{F}$ lies within $B_v(6r_v)$. Suppose not. Then $B_v \not\in \mathcal{F}$, so it overlaps with some ball $B_u \in \mathcal{F}$. If $B_u = \{b_u\}$ then we are done. Otherwise note that the ball $B_v(r_v)$ has measure ϵ and hence cannot lie within B'_u. Then since $B_v(3r_v)$ overlaps with B_u it follows that $4r_v \geq 3r$, where r is the radius of B_u. We come to a contradiction since B_u lies in $B_v(3r_v + 2r)$. Claim proved.

Lemma follows since any hitting set for $\mathcal{F}$ is an (ϵ, μ)-hitting set for the metric. □

Proof Sketch of Theorem 5.1: Fix ϵ. First we note that the proof of Theorem 2.2 of [19] actually shows that for each node u there exists a set S_u of measure at least $1 - \epsilon$ that has the following property: for every $v \in S_u$ a ball around u or v of radius δd_{uv} has measure at least $\epsilon_\delta = \frac{1}{2}\epsilon\delta^{2\log s}/s$. By Lemma 5.2 there exists an (ϵ_δ, μ)-hitting set H_δ of size $\Theta(s^3/\epsilon_\delta)$. Now using $H_{\delta/6}$ as the beacon set, the algorithm in Section 2 of [19] gives the desired (ϵ, δ)-triangulation; we omit the details. Similarly, the desired embedding is obtained using the algorithm from Section 3 of [19], with beacon set H_δ for small enough δ; again, we omit the details.

Note that the (ϵ, μ)-hitting set H_δ can be chosen at random according to Lemma 5.2. Therefore, the proof is algorithmic as long as one can sample at random with respect to μ. □

6 Approximate distance labeling via triangulation

In this section we show that for any fixed $\delta > 0$ any finite doubling metric has a $(0, \delta)$-triangulation of order $O(\log^2 n)$. This leads to a $(1 + \delta)$-approximate distance labeling scheme with $O(\log^2 n \log\log\Delta)$ bits per label, where Δ is the aspect ratio of the metric. we will use the notion of *doubling measure* [14].

Theorem 6.1. *For any $\delta > 0$ any s-doubling metric has a $(0, \delta)$-triangulation of order $O(\frac{1}{\delta})^{2 \log s} \log^2 n$. Moreover, such triangulation can be computed by an efficient centralized algorithm.*

Proof: As proved in [14], there exists a measure μ such that the measure of any ball $B_u(r)$ is at most s times the measure of the ball $B_u(r/2)$; moreover, such measure can be efficiently computed. Using this measure we select a set of 'beacons' (call them *μ-beacons*), partitioned into *levels*. Specifically, letting $K = 1/\min_u \mu(u)$ and $k = c \log n$ where c is a constant to be adjusted later, for each $j \in [\log K]$ select a set of level-j μ-beacons as follows: add each node u independently with probability $2^j k \mu(u)$. Moreover, we select another set of beacons (called *uar-beacons*), also partitioned into levels: for each $i \in [\log n]$ select a set of level-i *uar-beacons*: add each node independently with probability $2^i k/n$.

Now we will use beacons to define the distance labels of nodes. For each node u and each level $i \in [\log n]$, define level-i *uar-neighbours* of u as all level-i uar-beacons that lie within the ball $B_{ui} = B_u(r_{ui})$, where $r_{ui} = r_u(2^{-i})$. Define level-i *μ-neighbours* of u as all level-j μ-beacons that lie within the ball $B'_{ui} = B_u(2r_{ui}/\delta)$, for each j such that $\mu(B'_{ui}) < 2^{1-j}$. Finally, the label of u consists of distances to all its neighbours. Using Chernoff bounds it is easy to see that with high probability each node has $\Theta(k)$ neighbors on each level, for the total of $\Theta(k \log n)$ neighbors.

Fix a node pair uv and let $d = d_{uv}$. We need to show that a ball of radius δd around either u or v contains a beacon that is a neighbour of both u and v. Suppose there is no such beacon. Let $r = (1 + \delta)d$ and choose i such that $r_{ui} < r + d \leq r_{(u,i-1)}$. This "$u$-centric" choice yields some bounds on r_{vj}'s as well. Specifically, $r_{(v,i-1)} \geq r$ since $B_v(r)$ is contained in $B_u(d+r)$. Also, since $B_v(r+2d)$ contains $B_u(r+d)$, it has at least $n/2^i$ nodes, so $r_{vi} \leq r + 2d$.

Whp the ball B_{vi} contains a level-$(i-1)$ uar-beacon which is, by definition, a neighbour of v. If $B_{vi} \subset B_{(u,i-1)}$ then this beacon is also a level-$(i-1)$ neighbour of u, contradiction. Therefore we can assume $r_{vi} > \delta d$. Similarly, we can assume $r_{ui} > \delta d$. Therefore, in particular, the ball $B = B_u(\delta d)$ is contained in both B'_{ui} and B'_{vi}. Since the radii of the two bigger balls are $O(d)$, their measure is at most $c' \mu(B)$ where $c' = O(\frac{1}{\delta})^{2 \log s}$. Choose the largest j such that both $\mu(B'_{ui})$ and $\mu(B'_{vi})$ are at most $2/2^j$. Then for a sufficiently large $c = O(c')$ with high probability B contains a level-j μ-beacon, which is (by definition of a μ-neighbour) a level-i μ-neighbour of both u and v. □

It is easy to see that the above triangulation can be extended to a $(1 + \delta)$-approximate distance labeling scheme so that each neighbor is represented by a unique $\lceil \log n \rceil$-bit id, and each distance is encoded with $O(\log \frac{1}{\delta} + \log \log \Delta)$ bits, where Δ is the aspect ratio of the metric. Indeed, if following the literature on distance labeling (e.g. [10, 36]) we assume that the smallest distance is 1, then it suffices to use $O(\log \frac{1}{\delta})$ bits for the mantissa, and $O(\log \log \Delta)$ bits for the exponent. Since Δ can be arbitrarily large with respect to n, we improve over the labeling scheme of Talwar [36] that uses $O(\log \Delta)$ bits per label.

In fact, we can show that the $O(\log \log \Delta)$ worst-case dependence of the label length on Δ is optimal. Indeed, consider a metric d on three nodes $\{u, v, w\}$. Assume the smallest distance is $d_{uw} = 1$, and the largest distance is $d_{vw} = \Delta$. We need to encode d_{uv} with the node labels of u and v. Fix the encoding and suppose the two labels together take up at most k bits in the worst case. Then, obviously, at most 2^{k+1} different values of d_{uv} can be encoded. On the other hand, letting $c = (1 + \delta)^2$, the set of all values of d_{uv} that can be encoded must include a number in each interval $[c^{i-1}, c^i)$ such that $c \leq c^i \leq \Delta$. Since there are at least $\log \Delta$ such intervals, $k \geq \Omega(\log \log \Delta)$, claim proved.

It has come to our attention that very recently Mendel and Har-Peled [28] further sharpened the label length to $(\frac{1}{\delta})^{O(\log s)} (\log^2 n + \log n \log \log \Delta)$ bits. Their technique is different; in particular, it does not apply to $(0, \delta)$-triangulations.

Acknowledgments We thank Jon Kleinberg for useful discussions and help with the manuscript.

References

[1] N. Alon and J. Spencer. *The Probabilistic Method*. Wiley-Interscience Series in Discrete Mathematics and Optimization. John Wiley & Sons, New York, 2nd edition, 2000.

[2] P. Assouad. Plongements lipschitziens dans $\mathbf{R}^n$. *Bull. Soc. Math. France*, 111(4):429–448, 1983.

[3] J. Bourgain. On Lipschitz embeddings of finite metric spaces in Hilbert space. *Israel J. of Mathematics*, 52(1-2):46–52, 1985.

[4] H. T.-H. Chan, A. Gupta, B. M. Maggs, and S. Zhou. On hierarchical routing in bounded-growth metrics. In *16th ACM-SIAM Symp. on Discrete Algorithms (SODA)*, pages 762–771, 2005. Full and updated version available as a Carnegie Mellon University ETR CMU-PDL-04-106.

[5] M. Costa, M. Castro, A. Rowstron, and P. Key. PIC: Practical Internet coordinates for distance estimation. In *24th IEEE Intl. Conf. on Distributed Computing Systems (ICDCS)* , 2004.

[6] G. Crippen and T. Havel. *Distance geometry and molecular conformation* . Wiley, New York, NY, 1988.

[7] F. Dabek, R. Cox, F. Kaashoek, and R. Morris. Vivaldi: A decentralized network coordinate system. In *ACM SIGCOMM*, 2004.

[8] M. Fomenkov, k. claffy, B. Huffaker, and D. Moore. Macroscopic Internet topology and performance measurements from the DNS root name servers. In *Usenix LISA*, 2001.

[9] P. Francis, S. Jamin, C. Jin, Y. Jin, D. Raz, Y. Shavitt, and L. Zhang. IDMaps: A global Internet host distance estimation service. *IEEE/ACM Transactions on Networking*, 9:525–540, October 2001. Preliminary version in IEEE INFOCOM 1999.

[10] C. Gavoille, D. Peleg, S. Perennes, and R. Raz. Distance labeling in graphs. *J. of Algorithms*, 53(1):85–112, 2004. (Preliminary version in *12th ACM-SIAM SODA*, 2001).

[11] C. Gkantsidis, M. Mihail, and A. Saberi. Random Walks in Peer-to-Peer Networks. In *IEEE INFOCOM*, 2004.

[12] A. Gupta, R. Krauthgamer, and J. R. Lee. Bounded geometries, fractals, and low–distortion embeddings. In *44th Symp. on Foundations of Computer Science (FOCS)* , pages 534–543, 2003.

[13] J. Guyton and M. Schwartz. Locating nearby copies of replicated Internet servers. In *ACM SIGCOMM*, 1995.

[14] J. Heinonen. *Lectures on analysis on metric spaces*. Universitext. Springer-Verlag, New York, 2001.

[15] K. Hildrum, J. Kubiatowicz, S. Ma, and S. Rao. A note on the nearest neighbor in growth-restricted metrics. In *15th ACM-SIAM Symp. on Discrete Algorithms (SODA)* , pages 560–561, 2004.

[16] K. Hildrum, J. Kubiatowicz, S. Rao, and B. Y. Zhao. Distributed object location in a dynamic network. In *14th ACM Symp. on Parallel Algorithms and Architectures (SPAA)* , pages 41–52, 2002.

[17] S. Hotz. *Routing information organization to support scalable interdomain routing with heterogeneous path requirements*. PhD thesis, University of Southern California, 1994.

[18] P. Indyk. Algorithmic applications of low-distortion geometric embeddings. In *42nd Symp. on Foundations of Computer Science (FOCS)*, pages 10–33, 2001. Invited survey.

[19] J. Kleinberg, A. Slivkins, and T. Wexler. Triangulation and Embedding Using Small Sets of Beacons. In *45th Symp. on Foundations of Computer Science (FOCS)* , pages 444–453, 2004.

[20] C. Kommareddy, N. Shankar, and B. Bhattacharjee. Finding close friends on the Internet. In *12th IEEE Intl. Conf. on Network Protocols (ICNP)* , 2001.

[21] D. Kostic, A. Rodriguez, J. R. Albrecht, A. Bhirud, and A. Vahdat. Using random subsets to build scalable network services. In *4th USENIX Symp. on Internet Technologies and Systems (USITS)* , 2003.

[22] R. Krauthgamer, J. Lee, M. Mendel, and A. Naor. Measured descent: A new embedding method for finite metrics. *Geometric and Functional Analysis (GAFA)* , 15(4):839–858, 2005. Preliminary version in *45th IEEE FOCS*, 2004.

[23] R. Krauthgamer and J. R. Lee. The black-box complexity of nearest neighbor search. In *31st Intl. Colloquium on Automata, Languages and Programming (ICALP)*, volume 3142 of *Lecture Notes in Computer Science*, pages 858–869, 2004.

[24] R. Krauthgamer and J. R. Lee. Navigating nets: simple algorithms for proximity search. In *15th ACM-SIAM Symp. on Discrete Algorithms (SODA)*, pages 798–807, 2004.

[25] L. Lehman and S. Lerman. PALM: Predicting Internet network distances using peer-to-peer measurements. Technical report, MIT, January 2004.

[26] N. Linial, E. London, and Y. Rabinovich. The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15(2):215–245, 1995. (Preliminary version in *35th IEEE FOCS*, 1994).

[27] N. Linial and A. Wigderson. Course notes: Expander graphs and their applications. http://www.math.ias.edu/ boaz/ExpanderCourse/, 2002.

[28] M. Mendel and S. Har-Peled. Fast construction of nets in low dimensional metrics, and their applications. In *21st ACM Symp. on Computational Geometry (SoCG)*, pages 150–158, 2005.

[29] R. Motwani and P. Raghavan. *Randomized algorithms*. Cambridge University Press, Cambridge, 1995.

[30] T. E. Ng and H. Zhang. Predicting Internet network distance with coordinates-based approaches. In *IEEE INFOCOM*, 2002.

[31] R. Percacci and A. Vespignani. Scale-free behavior of the Internet global performance. *The European Physical Journal B*, 32(4):411–414, 2003. Also appeared as *arXiv e-print cond-mat/0209619*, September 2002.

[32] M. Pias, J. Crowcroft, S. Wilbur, T. Harris, and S. Bhatti. Lighthouses for scalable distributed location. In *2nd Intl. Workshop on Peer-to-Peer Systems (IPTPS)*, 2003.

[33] B. Pittel. On spreading a rumor. *SIAM J. Appl. Math.*, 47(1):213–223, 1987.

[34] C. G. Plaxton, R. Rajaraman, and A. W. Richa. Accessing nearby copies of replicated objects in a distributed environment. *Theory Comput. Syst.*, 32(3):241–280, 1999. Preliminary version in *9th ACM SPAA*, 1997.

[35] D. Ron. Property Testing (a tutorial). In S. Rajasekaran, P. Pardalos, J. Reif, and J. Rolim, editors, *Handbook of Randomized Computing, Vol. II*, pages 597–649. Kluwer Academic, Dordrecht, Netherlands, 2001.

[36] K. Talwar. Bypassing the embedding: approximation schemes and compact representations for growth restricted metrics. In *36th ACM Symp. on Theory of Computing (STOC)*, pages 281–290, 2004.

[37] A. Vazquez, R. Pastor-Satorras, and A. Vespignani. Large-scale topological and dynamical properties of Internet. *Phys. Rev. E,* 65, 066130, 2002.

[38] V. Vishnumurthy and P. Francis. On Heterogeneous Overlay Construction and Random Node Selection in Unstructured P2P Networks. In *IEEE INFOCOM*, 2006.

[39] B. Wong, A. Slivkins, and E. G. Sirer. Meridian: A Lightweight Network Location Service without Virtual Coordinates. In *ACM SIGCOMM*, 2005.

[40] D. Y. Xiao. The Evolution of Expander Graphs. BA Thesis, Harvard University, 2003. Available from *http://www.cs.princeton.edu/ dxiao*.

[41] B. Zhao, L. Huang, S. Rhea, J. Stribling, A. Joseph, and J. Kubiatowicz. Tapestry: a resilient global-scale overlay for service deployment. *IEEE Journal on Selected Areas in Communications*, 22(1):41–53, 2004.

A Tools from Probability

We use the standard Chernoff bounds for bounded independent random variables, e.g. see [29].

Theorem A.1 (Chernoff bounds, folklore). *Let X be the sum of finitely many independent random variables $X_i \in [0, y]$, for some $y > 0$. Let $\epsilon \in (0, 1)$ and $\beta \geq 1$. Then:*

(a) $\Pr[X < (1-\epsilon)\mu] \leq e^{-\epsilon^2 \mu/2y}$*, for any* $\mu \leq E(X)$.

(b) $\Pr[X > \beta\mu] \leq \left[\frac{1}{e}(\frac{e}{\beta})^{\beta}\right]^{\mu/y}$*, for any* $\mu \geq E(X)$.

We also derive and use a version of Chernoff bounds that applies to near-independent random variables. While it is possible that this result appears in the literature, we have not been able to find a reference.

Theorem A.2. *Consider the sum* $X = \sum_{i=0}^{n} X_i$ *where* X_i *are positive integer-valued random variables upper-bounded by for* $k \in \mathbb{N}$*. Let* $\epsilon \in (0, 1)$ *and* $\beta \geq 1$*. Let* $\mathcal{F}_i = \sigma(X_0, X_1, \ldots, X_i)$ *and* $\mathcal{F}_{-1} = \emptyset$*. Then:*

(a) If $E(X_i|\mathcal{F}_{i-1}) \geq \mu_i$ *for all* i*, then* $\Pr[X < (1-\epsilon)\mu/k] \leq e^{-\epsilon^2\mu/2k}$ *for any* $\mu \leq \sum_i \mu_i$*.*

(b) If $E(X_i|\mathcal{F}_{i-1}) \leq \mu_i \leq 1$ *for all* i*, then* $\Pr[X > \beta\mu k] \leq \left[\frac{1}{e}(\frac{e}{\beta})^{\beta}\right]^{\mu}$ *for any* $\mu \geq \sum_i \mu_i$*.*

Proof. For part (a) we construct a family of independent 0-1 random variables $\{Y_0, Y_1, \ldots, Y_n\}$ such that $Y_i \leq X_i$ and $E(Y_i) = \mu_i/k$ for all i. Then by part (a) of Chernoff bounds for any $\mu \leq \sum_i \mu_i$ we have

$$\Pr[X < (1-\epsilon)\mu/k] \leq \Pr[\textstyle\sum Y_i < (1-\epsilon)\mu/k] \leq e^{-\epsilon^2\mu/2k}.$$

Similarly, for part (b) we construct a family of independent 0-1 random variables $\{Y_0, Y_1, \ldots, Y_n\}$ such that $k\, Y_i \geq X_i$ and $E(Y_i) = \mu_i$ for all i. Then by part (b) of Chernoff bounds for any $\mu \geq \sum_i \mu_i$ we have

$$\Pr[X > \beta\mu k] \leq \Pr[\textstyle\sum Y_i > \beta\mu] \leq \left[\frac{1}{e}(\frac{e}{\beta})^{\beta}\right]^{\mu}.$$

Let us start with part (b). For each i-vector $x = (x_0, x_1, \ldots, x_{i-1}) \in [k+1]^i$ let us define the event

$$A_x := \{\omega \in \Omega : X_0(\omega) = x_0,\ X_1(\omega) = x_1,\ \ldots,\ X_{i-1}(\omega) = x_{i-1}\}. \tag{3}$$

Let us define Z_x to be a 0-1 valued independent random variable with expectation

$$E(Z_x) = (\mu_i - p_x)/(1 - p_x), \text{ where } p_x := \Pr[X_i > 0 | A_x]. \tag{4}$$

This is well-defined because $\mu_i \geq E(X_i|A_x) \geq p_x$. Now let us define Y_i as a 0-1 valued random variable

$$Y_i := Y_i(X_0, X_1, \ldots, X_{i-1}), \text{ where } Y_i(x) := 1_{\{X_i > 0\}} \vee Z_x.$$

To define Y_0, for notational convenience set $x = \emptyset$, let Z_x be an independent 0-1 random variable with expectation defined by (3) with $A_x = \emptyset$, and define $Y_0 = 1_{\{X_0 > 0\}} \vee Z_x$. This completes the definition of the Y_i's. It remains to establish that these random variables have the desired properties.

Claim A.3. *For all* i *we have (a)* $k\, Y_i \geq X_i$ *and (b)* $E[Y_i] = \mu_i$*.*

Proof. For part (a) note that if $X_i > 0$ then by definition of Y_i we have $Y_i = 1 \geq X_i/k$. For part (b) note that for any i-vector $x = (x_0, x_1, \ldots, x_{i-1}) \in [k+1]^i$ if the event A_x happens then $Y_i = 1$ if and only if $X_i > 0$ or $Z_x = 1$. Therefore $\Pr[Y_i = 1|A_x] = p_x + (1 - p_x)\Pr[Z_x = 1] = \mu_i$. □

Claim A.4. *Random variables* $\{Y_0, Y_1, \ldots, Y_n\}$ *are independent.*

Proof. Fix some vector $y = (y_0, \ldots, y_n) \in \{0,1\}^{n+1}$. For each i let us define

$$\begin{aligned} B_i &= \{\omega \in \Omega : Y_0(\omega) = y_0,\, Y_1(\omega) = y_1, \ldots, Y_i(\omega) = y_i\}, \\ \alpha_i &= \mu_i y_i + (1-\mu_i)(1-y_i). \end{aligned}$$

To prove the lemma we need to show that $\Pr[B_n] = \prod_{i=0}^n \alpha_i$. To this end we will prove by induction on i that $\Pr[B_i] = \prod_{j=0}^i \alpha_j$ for all $i \leq n$. Indeed, for $i = 0$ this just follows from the definition of Y_0. Suppose the induction hypothesis is true for some i. Define the event $C = \{Y_{i+1} = y_{i+1}\}$ and the set of vectors

$$\{x = (x_0, \ldots, x_i) \in [k+1]^{i+1} : B_i \cap A_x \neq \emptyset\}.$$

Note that $\Pr[C|A_x \cap B_i] = \alpha_{i+1}$ by the proof of the Claim A.3. Therefore

$$\Pr[B_{i+1}] = \Pr[B_i \cap C] = \sum_{x \in S} \Pr[A_x \cap B_i] \times \Pr[C|A_x \cap B_i] = \Pr[B_i] \times \alpha_{i+1},$$

and the induction step follows. □

This completes the proof of part (b). For part (a), we proceed in a similar fashion; we borrow the definitions of events A_x and probabilities p_x. For each i-vector $x = (x_0, x_1, \ldots, x_{i-1}) \in [k+1]^i$ let us define Z^*_x as a 0-1 valued independent random variable with expectation $E(Z^*_x) = 1 - \mu_i / k\, p_x$. This is well-defined because $\mu_i \leq E(X_i|A_x) \leq k\, p_x$. Now for $i \geq 1$ we define Y^*_i as a 0-1 valued random variable

$$Y^*_i := Y^*_i(X_0, X_1, \ldots, X_{i-1}), \text{ where } Y^*_i(x) := 1_{\{X_i = 0\}} \wedge Z^*_x,$$

and we let $Y^*_0 = 1_{\{X_0 > 0\}} \vee Z^*_\emptyset$. This completes the definition of the Y^*_i's. It remains to prove the suitable analogs of Claims A.3 and A.4, namely that $Y_i \leq X_i$ and $E(Y_i) = \mu_i / k$ for all i and that the Y^*_i's are independent. The proofs are similar to those in part (b); we omit them here. □

B Constructing high-expansion graphs

For an undirected graph, the *expansion* is defined as $\min \frac{|\partial(S)|}{|S|}$, where the minimum is over all nonempty sets S of at most $n/2$ vertices, and $\partial(S)$ stands for the set of edges with exactly one endpoint in S. We can generalize this definition to *weighted* undirected graphs, or, equivalently, to symmetric non-negative matrices: we just define $\partial(S)$ to be the total weight of all edges with exactly one endpoint in S. We can further extend this definition to directed graphs (non-symmetric matrices) by considering the weight of all edges leaving S.

For a pre-defined absolute constant, *expander* is an undirected graph whose expansion is at least this constant. Expanders are well-studied and have rich applications, see [27, 1, 29, 40] for more background.

The following is a standard result on expanders, e.g. see p. 10 of [11] for a proof.

Theorem B.1 (Folklore). *Fix node set* V. *Suppose for each node* u *we choose three nodes idependently and uniformly at random from* V, *and create undirected links between* u *and these three nodes. Then the resulting graph is an expander with high probability.*

In a slightly stronger version of this theorem we select nodes from (and construct an expander on) any given subset Q of nodes, whereas we need the failure probability to be low in terms of n, not the size of Q. Hence we create $O(\log n)$ links per node instead of just three.

Theorem B.2. *Fix node set V of n nodes, and a subset $Q \subset V$. Suppose for each node $u \in Q$ we choose at least $3 \log n$ nodes idependently from a near-uniform distribution on Q, and create undirected links between u and these nodes. Then the induced graph on Q is an expander with high probability.*

For this paper we need a somewhat more complicated version of Theorem B.2 where the edge selection is not quite independent:

Theorem B.3. *Fix a set V of n nodes, and a subset $Q \subset V$ where the nodes are numbered from 1 to $|Q|$. Suppose for each node $i \in Q$ we choose at least $k = 3 \log n$ nodes at random from a set Q_i containing Q, and create directed links between u and these nodes. Let us denote these k nodes by $X_i = (X_{ij} : j \in [k])$, where we treat the X_{ij}'s as Q_i-valued random variables. Let G_Q be the induced directed graph on Q. We characterize the joint distribution of X_{ij}'s as follows:*

- *for every fixed node i, random variables X_{ij}, $j \in [k]$ are independent.*
- *each random variable X_{1j}, $j \in [k]$ has a near-uniform distribution on Q_i.*
- *for each $i \geq 2$ each random variable X_{ij}, $j \in [k]$ has a near-uniform distribution on Q_i conditional on any given values of the random vectors $(X_l : l < i)$.*

Then:

(a) with high probability graph G_Q is a (k, γ)-quasi-expander, for a constant γ.

(b) if $Q_i = Q$ for all i, then with high probability the undirected version of G_Q is an expander.

The proof of Theorem B.3(b) follows that of Theorem B.1, except we use Theorem A.2 instead of the standard Chernoff bounds.

C Random node selection in a network

An *undirected version* of a directed graph G is an undirected graph on the same node set, possibly with multiple edges, where each directed edge $uv \in G$ is replaced by an undirected edge uv.

Definition C.1. A directed graph $G = (V, E)$ is a (d_0, γ)-*quasi-expander* if it has the following property. Take any subset $S \subset V$ such that each node in S has out-degree at least d_0. Let E_S be the set of edges entering or leaving the nodes in S. Then there exists a constant-degree expander on node set $V \setminus S$, with edge set E^*, such that the undirected version G_S of the graph $(V, E_S \cup E^*)$ has expansion at most γ. Call this undirected graph G_S an (d_0, γ, S)-*extension* of G.

Theorem C.2. *Let G be a directed graph on an n-node set V; let G^* be its undirected version. Fix node u and consider a subset $Q \subset V$ Suppose that:*

- *for some (d_0, γ) graph $G|Q$ is a (d_0, γ)-quasi-expander,*
- *after pinging any node $v \in V$, node u can, at unit cost, determine whether $v \in Q$.*
- *node u knows numbers $d \geq \deg(G^*)$, $d_Q \geq \deg(G^*|Q)$, $t \geq (d_Q/\gamma)^2 (\log n)$ and d_0.*
- *node u is given a* random seed*: an address of some node.*

Then for any $k_0 \in \mathbb{N}$ there exists a randomized (u, G, G^*)-distributed algorithm such that:

(a) if $\deg(G|Q) \geq d_0$ then node u acquires addresses of k_0 nodes $X_i \in Q$, where the X_i's are independent random variables with a near-uniform distribution on Q; we say that the algorithm succeeds*. Else the algorithm either succeeds or aborts.*

(b) The total running time and the load on node u are $O(k_0 dt)$. The load on ry other node w is at most $O\left(\sum_{wv \in G} Z_v\right)$, where Z_v is 1 if node v is "visited" by algorithm, and 0 otherwise,[3] in particular it

[3] For each node v, the algorithm either does not touch the list of G-neighbors of v, or reads the entire list at once. In the latter case we say that the algorithm *visits* node v.

is 0 for all $v \notin Q$. If the random seed was selected independently from a near-uniform distribution τ on Q, then in the probability space induced by the algorithm and τ, $E(Z_v) = O(k_0 t/|Q|)$ for each $v \in Q$.

We omit the proof from this version of the paper.