

An Alternate Proof of Near-Optimal Light Spanners

Received Mar 7, 2024
Accepted Nov 24, 2024
Published Jan 10, 2025

Key words and phrases
Extremal Graph Theory,
Sparsifiers, Spanners, Lightness,
Hiker Lemmas

Greg Bodwin^a  

^a University of Michigan, USA

ABSTRACT. In 2016, a breakthrough result of Chechik and Wulff-Nilsen [SODA '16] established that every n -node graph G has a $(1+\varepsilon)(2k-1)$ -spanner of lightness $O_\varepsilon(n^{1/k})$, and recent followup work by Le and Solomon [STOC '23] generalized the proof strategy and improved the dependence on ε . We give a new proof of this result, with the improved ε -dependence. Our proof is a direct analysis of the often-studied greedy spanner, and can be viewed as an extension of the folklore Moore bounds used to analyze spanner sparsity.

1. Introduction

We study *spanners*, which are a graph-theoretic primitive with applications in graph algorithms, network design, and sketching [1].

DEFINITION 1.1 (Spanners [26, 25]). Given a graph G , a t -spanner is an edge-subgraph H that satisfies $\text{dist}_H(u, v) \leq t \cdot \text{dist}_G(u, v)$ for all vertices u, v .

The usual goal is to design a spanner with a favorable tradeoff between its stretch t and its *size*. There are two different ways that spanner size is commonly measured. One is by the sparsity (number of edges) of the spanner. The stretch/sparsity tradeoff has long been understood, thanks to the following classic theorem by Althöfer et al. [3] Here and throughout the paper, all graphs are undirected and may have arbitrary positive edge weights.

THEOREM 1.2 ([3]). *For all positive integers k, n , every n -node graph G has a $(2k-1)$ -spanner H on $|E(H)| \leq O(n^{1+1/k})$ edges. This tradeoff is best possible, assuming the girth conjecture [18].*

This is an invited paper from SOSA 2024 [5]. This work was supported by NSF:AF 2153680.

The other popular way to measure spanner size is by the *total edge weight* in the spanner $w(H)$. In general, the total edge weight required for a t -spanner might be unbounded, say by scaling up the edge weights of the input graph. So in order to prove stretch/weight tradeoffs, we typically normalize the spanner weight by the weight of a minimum spanning tree (MST) of the original graph. This size measure is called *lightness*:

DEFINITION 1.3 (Spanner Lightness). The lightness of a subgraph H of a graph G is the quantity

$$\ell(H | G) := \frac{w(H)}{w(\text{MST}(G))}$$

where $\text{MST}(G)$ is a minimum spanning tree of G .¹ For brevity we also write $\ell(H) := \ell(H | H)$.

There has been a long line of work on the stretch/lightness tradeoff; see Table 1 for the progression of results. A key result in this sequence, by Chechik and Wulff-Nilsen [13], established the following analog of Theorem 1.2:

THEOREM 1.4 ([13]). For all $\varepsilon > 0$ and positive integers k, n , every n -node graph G has a $(1 + \varepsilon)(2k - 1)$ -spanner H of lightness $\ell(H | G) = O_\varepsilon(n^{1/k})$.

The stretch/lightness tradeoff in Theorem 1.4 is best possible, assuming the girth conjecture [18], and up to its dependence on ε . The theorem is proved using an ingenious framework for hierarchical graph clustering. A followup paper by Le and Solomon [22] refined and vastly generalized the scope of this hierarchical clustering method, improving the hidden ε -dependence in the lightness bound of Theorem 1.4, and also gaining broad applications to the study of light spanners in some more specific graph classes.

Meanwhile, perhaps the most popular spanner construction in the literature is the *greedy algorithm* (see Algorithm 1). The greedy algorithm is ubiquitous because it is simple, easy to prove correct, and its stretch/sparsity and stretch/lightness tradeoffs are both known to be *existentially optimal* [3, 19]. That is, the stretch/lightness tradeoff achieved by any algorithm – including the hierarchical clustering method [13, 22] – is automatically achieved by the greedy algorithm as well. This has motivated interest in spanner size bounds that are proved by analyzing the greedy algorithm directly. In the context of spanner *sparsity*, there is indeed a simple proof of Theorem 1.2 that works by directly analyzing the greedy spanner. This proof is called the Moore bounds and it is considered folklore; we recap the proof in Section 2. In the context of spanner *lightness*, there are some arguments that directly analyze the greedy spanner [3, 12, 15], but they all show suboptimal lightness bounds that do not quite match the one in Theorem 1.4.

¹ Throughout this paper, we will assume where convenient that graphs are connected, so that $\text{MST}(G)$ exists. Otherwise, a minimum spanning forest may be used.

Stretch	Lightness	Analyzes Greedy Spanner?	Citation
$2k - 1$	$O(n/k)$	✓	[3]
$(1 + \varepsilon) \cdot (2k - 1)$	$O_\varepsilon(k \cdot n^{1/k})^*$	✓	[12]
$(1 + \varepsilon) \cdot (2k - 1)$	$O_\varepsilon\left(\frac{k}{\log k} \cdot n^{1/k}\right)^*$	✓	[15]
$(1 + \varepsilon) \cdot (2k - 1)$	$O(\varepsilon^{-(3+2/k)} n^{1/k})$		[13]
$(1 + \varepsilon) \cdot (2k - 1)$	$O(\varepsilon^{-1} n^{1/k})^*$		[22]
$(1 + \varepsilon) \cdot (2k - 1)$	$O(\varepsilon^{-1} n^{1/k})$	✓	this paper
$(1 + \varepsilon) \cdot (2k - 1), \varepsilon = n^{-\frac{1}{2k-1}}$ only	$\Omega_k(\varepsilon^{-1/k} n^{1/k})$	n/a	[6]

Table 1. Work on the stretch/lightness tradeoff for spanners. Lightness bounds marked with * have an *unconditionally* optimal dependence on n , i.e., they automatically improve if the girth conjecture fails.

The contribution of this paper is a new proof of Theorem 1.4, with the improved ε -dependence from [22], which directly analyzes the greedy spanner (or, more accurately, which directly analyzes graphs of high weighted girth [15]; see Section 3.1). Our proof follows the template of the Moore bounds, and so it may also have an advantage in conceptual familiarity to a reader who is primarily comfortable with the literature on spanner sparsity.

1.1 Other Related Work

In subsequent work to the current paper, the author and Flics [6] constructed a lower bound graph on which any $(1 + \varepsilon)(2k - 1)$ -spanner has lightness $\Omega_k(\varepsilon^{-1/k} n^{1/k})$, for some particular choice of ε (as usual, this result is conditional on the girth conjecture). This implies that the dependence of ε^{-1} in the upper bound of [22] and the current paper cannot be *completely* removed, although it remains an interesting open problem to tighten the ε -dependence on either the upper or the lower bounds side.

The greedy algorithm runs in time $O(mn^{1+1/k})$. There has been a line of work on alternate algorithms that achieve similar lightness guarantees, but which run faster. Elkin and Solomon [16] constructed spanners in near-linear time whose lightness is suboptimal by a factor of k . Alstrup, Dahlgaard, Filtser, Stöckel, and Wulff-Nilsen [2] constructed spanners with near-optimal lightness (up to dependence on ε), in time approximately $O(n^{2+1/k})$. Finally, the recent work of Le and Solomon [22] tightened the ε -dependence, and achieved near-linear runtime.

Light spanners have also been studied intensively in various specific graph classes. This most notably includes Euclidean spaces (see e.g. [19, 22, 24, 17] and references within), but

other domains have also received attention, such as doubling metrics [21, 23, 7, 11, 10, 9, 20], minor-free graphs [8], and polygonal domains [4].

1.2 Organization

A lot of this paper is optional “warmup” content rather than the main proof, and the enterprising reader can get the full proof by reading Sections 3 and 5 only. The surrounding warmup proofs, discussions, and puzzles are recommended to help build intuition.

2. Warmup 1: The Moore Bounds for Spanner Sparsity

We will begin by recapping the proof of the stretch/sparsity tradeoff given in Theorem 1.2. This theorem analyzes the following simple *greedy algorithm*:

```

1: Input: Graph  $G = (V, E, w)$ , stretch  $t$ 
2:
3: Let  $H = (V, \emptyset, w)$  be the initially-empty spanner
4: foreach  $(u, v) \in E$  in order of nondecreasing weight do
5:     if  $\text{dist}_H(u, v) > t \cdot w(u, v)$  then
6:         add  $(u, v)$  to  $H$ 
7: return  $H$ 

```

Algorithm 1. The Greedy Spanner Algorithm [3]

We first observe that the output spanner H of the greedy algorithm with stretch parameter $2k - 1$ has girth (shortest cycle length) $> 2k$ [3]. (We omit this proof, as it is standard, but we note that it is implied by Lemma 3.2 to follow.) Theorem 1.2 then follows from the *Moore bounds*, which limit the maximum possible number of edges in a high-girth graph:

THEOREM 2.1 (Moore Bounds). *For any positive integers n, k , every n -node graph H with girth $> 2k$ has $O(n^{1+1/k})$ edges.*

The proof of the Moore bounds is a counting argument over the *edge-simple k -paths* of H . Recall that an edge-simple path is one that does not repeat edges. One part of this counting argument is the following *dispersion lemma*, implying that these paths are “dispersed” around the graph rather than having several of them concentrated on a pair of endpoint nodes.

LEMMA 2.2 (Unweighted Dispersion Lemma). *H may not have two distinct edge-simple k -paths with the same endpoints s, t .*

PROOF. Suppose for contradiction that π_a, π_b are two distinct $s \rightsquigarrow t$ edge-simple k -paths in H . The subgraph $\pi_a \cup \pi_b$ is not a tree, since it contains two distinct $s \rightsquigarrow t$ paths, and so it contains a cycle C . This cycle must have $|C| \leq |\pi_a| + |\pi_b| = 2k$ edges, which contradicts that H has girth $> 2k$. ■

The dispersion lemma implies an upper bound of $O(n^2)$ on the number of edge-simple k -paths in H . The other part of the argument is a *counting lemma*, which gives a corresponding lower bound. Only the last “full” counting lemma in the following sequence is used, but the proof strategy is to bootstrap it by starting with weaker versions.

LEMMA 2.3 (Unweighted Weak Counting Lemma). *If $|E(H)| \geq n$, then H contains an edge-simple k -path.*

PROOF. Since $|E(H)| \geq n$, H contains a cycle C . Since H has girth $> 2k$, there are $> 2k$ edges in C . Thus, any subpath of C of length k is an edge-simple k -path. ■

LEMMA 2.4 (Unweighted Medium Counting Lemma). *H contains at least $|E(H)| - n$ edge-simple k -paths.*

PROOF. Repeat the following process until no longer possible: find an edge-simple k -path π , record it, and then delete any **one** edge in π from H to ensure that we don’t re-record π in a future round. By the weak counting lemma, we may repeat this process for at least $|E(H)| - n$ rounds. ■

LEMMA 2.5 (Unweighted Full Counting Lemma). *Let $d := |E(H)|/n$. If $d \geq 2$, then H contains $n \cdot \Omega(d)^k$ edge-simple k -paths.*

PROOF. Let H' be a random edge-subgraph of H , obtained by keeping each edge independently with probability $2/d$. Let p, p' be the number of edge-simple k -paths in H, H' , respectively. On one hand, for any edge-simple k -path π in H , the probability that π survives in H' is $\Theta(d)^{-k}$, and so $\mathbb{E}[p'] = p \cdot \Theta(1/d)^k$. On the other hand, we have

$$\begin{aligned} \mathbb{E}[p'] &\geq \mathbb{E}[|E(H')| - n] && \text{Medium Counting Lemma} \\ &= |E(H)| \cdot \frac{2}{d} - n \\ &= 2n - n \\ &= n. \end{aligned}$$

Combining these inequalities, we get $n \leq p \cdot \Theta(1/d)^k$, and rearranging gives $p \geq n \cdot \Theta(d)^k$. ■

We are now ready to complete the proof of the Moore bounds. Let $d := |E(H)|/n$. If $d < 2$ then $|E(H)| = O(n)$ and we are done. Otherwise, if $d \geq 2$, then by the full counting lemma H has $n \cdot \Omega(d)^k$ edge-simple k -paths. Meanwhile, the dispersion lemma implies that H has $O(n^2)$ edge-simple k -paths. Comparing these estimates, we get

$$n \cdot \Omega(d)^k \leq O(n^2).$$

Rearranging terms in this inequality, we get $d \leq O(n^{1/k})$, and so $|E(H)| = O(n^{1+1/k})$.

3. Lightness Reductions from Prior Work

3.1 The Weighted Girth Framework

In our previous proof of the Moore bounds, the first step is to observe that the output spanner of the greedy algorithm has high girth. For lightness, Elkin, Neiman, and Solomon [15] formalized the analogous method, which we will use in this paper.

DEFINITION 3.1 (Normalized Weight and Weighted Girth [15]). For a cycle C in G , we define its normalized weight to be

$$w^*(C) := \frac{w(C)}{\max_{e \in C} w(e)}.$$

The weighted girth of G is the minimum value of $w^*(C)$ over all cycles C in G .

LEMMA 3.2 ([15]). *The greedy algorithm with parameter t returns a graph H with weighted girth $> t + 1$.*

PROOF. Let C be a cycle in G of normalized weight $w^*(C) \leq t + 1$. It suffices to argue that not all edges of C will be added to H . Let (u, v) be the last edge in C considered by the greedy algorithm, and suppose that all previous edges in C were added to H . Then there is a $u \rightsquigarrow v$ path in H , through the other edges in C , of total weight

$$w(C) - w(u, v) \leq (t + 1) \cdot w(u, v) - w(u, v) = t \cdot w(u, v).$$

This inequality means that the greedy algorithm will reject the edge (u, v) , rather than adding it to H , and so the cycle C does not survive in H . ■

Also note that the greedy algorithm essentially contains a run of Kruskal's algorithm within it [14], and so the output spanner H contains an MST of G . This means that instead of bounding $\ell(H | G)$, we may equivalently bound $\ell(H)$ over all graphs H of weighted girth $> (1 + \varepsilon)2k$.

3.2 Reduction to Unit-Weight Spanning Cycles

It will be convenient in the main proof to reduce to the setting where the spanner H has a very particular structure for its MST:

DEFINITION 3.3 (Unit-Weight Spanning Cycles). We say that a cycle C in H is a unit-weight spanning cycle if C is Hamiltonian (it contains each node exactly once), all edges in C have weight 1, and all edges in $E(H)$ have weight ≥ 1 .

So, for example, any tree created by deleting any one edge from a unit-weight spanning cycle C is an MST. We will reduce to the case where H has a unit-weight spanning cycle; the parts of this reduction all appear implicitly or explicitly in prior work [13, 22, 15].

LEMMA 3.4. *Let H be a non-forest n -node graph with weighted girth $> t$ and lightness ℓ . Then there exists a graph H' with $O(n)$ nodes, weighted girth $> t$, lightness $\Omega(\ell)$, and a unit weight spanning cycle.*

PROOF. We split the reduction into two steps: first we modify to a unit-weight MST, and then we change the MST into a spanning cycle.

Reduction to Unit-Weight MST.

- Rescale the edge weights of H so that the average edge weight in $\text{MST}(H)$ is 1. Rescaling affects neither the weighted girth nor the lightness of H .
- For all edges $e \in \text{MST}(H)$ of weight $w(e) > 1$, add new nodes to subdivide e into a path of $\lceil w(e) \rceil$ edges. Each new edge is assigned weight $w(e)/\lceil w(e) \rceil$. Notice that:
 - The total weight in $\text{MST}(H)$ is $n - 1$ after the previous rescaling step, and therefore we add at most $n - 1$ new nodes to H in this step.
 - Additionally, the weighted girth of H does not change as we subdivide edges. This holds since for each cycle C the total weight of C is unchanged, and also C must have a heaviest edge that is not in $\text{MST}(H)$; since this heaviest edge is not subdivided, the normalized weight $w^*(C)$ remains the same.
- Finally, for all edges $e \in E(H)$ of weight < 1 , increase $w(e)$ to 1 (this step is applied to both MST and non-MST edges). Notice that:
 - By the previous step every edge in $\text{MST}(H)$ has weight in the range $[1/2, 1]$, and so this step increases $w(\text{MST}(H))$ by at most a factor of 2, and
 - The weighted girth of H is nondecreasing in this step. This is because for each cycle C , $w(C)$ is increasing, and moreover: (1) if C has an edge of weight ≥ 1 then its heaviest edge weight does not change, and (2) if all edges in C have weight ≤ 1 , then after this step all edges have weight 1, so its normalized weight is $w^*(C) = |C| > t$.

Reduction to Spanning Cycle. Next, we reduce to the setting where H has a spanning cycle C .

A *tour* T of $\text{MST}(H)$ is a circularly-ordered sequence of nodes, with repeats, of the form $T = (v_0, v_1, \dots, v_{2n-2} = v_0)$, which is the node sequence of a closed walk on $\text{MST}(H)$ that uses every edge exactly twice with opposite orientations. Fix a tour T of $\text{MST}(H)$, and then construct H' as follows:



Figure 1. (Left) For the pictured $\text{MST}(H)$, the red line around the outside of the tree traces the tour $T = (a, b, c, b, d, b, a, e, f, e, g, e, a)$. (Right) We construct H' by mapping the tour T to a spanning cycle on $2n - 2$ nodes, making copies of nodes from H as needed.

- A tour of an n -node tree always contains exactly $2n - 2$ nodes. We will take these $2n - 2$ nodes as the vertex set of H' ; that is, some nodes in H have several copies in H' .
- The tour T will be the spanning cycle of H' , meaning that for each pair of adjacent nodes along T , we include the corresponding edge in $E(H')$ with weight 1.
- For each non-spanning-cycle edge $(u, v) \in E(H \setminus T)$, we choose an arbitrary copy $u_i, v_j \in V(H')$ of u, v respectively. Then we include $(u_i, v_j) \in E(H')$ with the same weight as (u, v) .

It is immediate from the construction that H' has $O(n)$ nodes and a unit-weight spanning cycle, and that $w(H') \geq w(H)$, and so lightness only changes by a constant factor. Moreover, we only create one new cycle when we move from H to H' , which is the spanning cycle T itself. We have $w^*(T) = 2n - 2$, and since we have assumed that H is not a forest, it has a cycle and therefore its weighted girth is $\leq n$. It follows that the weighted girth of H' is at least as large as the weighted girth of H . ■

One of the advantages of reducing to the case where H has a unit-weight spanning cycle is that we can limit its maximum edge weight:

LEMMA 3.5. *Let H be an n -node graph with weighted girth $> t$ and a unit weight spanning cycle C . Then all edges in H have weight $< \frac{n}{2(t-1)}$.*

PROOF. Consider an edge (u, v) , and consider the cycle formed by (u, v) and the shorter $u \rightsquigarrow v$ path through the spanning cycle, which uses at most $n/2$ edges. The normalized weight of this cycle is at most

$$\frac{w(u, v) + n/2}{w(u, v)} = 1 + \frac{n}{2w(u, v)}.$$

By the weighted girth of H , this quantity must be $> t$. Rearranging, we get $w(u, v) < \frac{n}{2(t-1)}$. ■

4. Warmup 2: Lightness Bounds via Monotone Paths

We will next prove a weaker version of our main result, with an additional suboptimal k factor, in order to introduce some of our new proof ideas.

THEOREM 4.1 (Warmup). *Let $\varepsilon > 0$, let k, n be positive integers, and let H be an n -node graph with a unit-weight spanning cycle C and weighted girth $> (1 + 2\varepsilon) \cdot 2k$. Then*

$$w(H) = O\left(\varepsilon^{-1}kn^{1+1/k}\right).$$

The 2ε term in the weighted girth, rather than ε , is purely for convenience in the analysis to follow; by reparametrizing $\varepsilon \leftarrow \varepsilon/2$ it does not affect the theorem statement. We will also assume for convenience that all non-spanning-cycle edges in H have distinct weights; if not, any tiebreaking method will work, e.g. the lexicographically smaller edge is considered lighter. Finally, we arbitrarily choose one direction around the spanning cycle C to be the forward direction, and the reverse to be backward.

4.1 Edge-Safe Paths and Warmup Proof Intuition

Our first step is to define the kind of paths that will be the focus of our Moore-bound-like counting argument. The following kind of path is analogous to a *single edge* in the Moore bounds; we will ultimately focus on paths made of k concatenated copies of this basic building block.

DEFINITION 4.2 (Edge-Safe Paths). A path π is *safe* for an edge (u, v) if, for some integer $0 \leq s \leq \varepsilon w(u, v)$, it has the following structure: it starts with a prefix of exactly s forward spanning cycle edges, then it uses the edge (u, v) , and then it ends with a suffix of exactly s backward spanning cycle edges. We will say that π is *extra-safe* for (u, v) if $s \leq \varepsilon w(u, v)/2$.

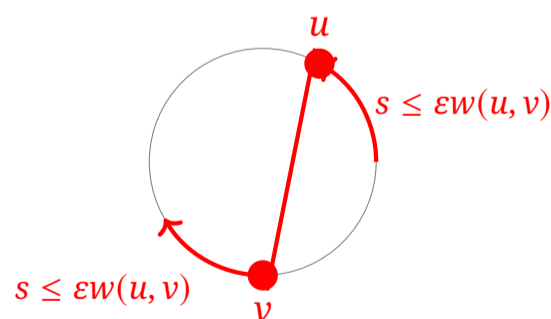


Figure 2. A path that is safe for the edge (u, v) (considering the counterclockwise direction around the spanning cycle to forward, and clockwise to be backward)

The detail of extra-safety in this definition can be ignored for now; it will be useful later for a minor technical reason in the counting lemma. The requirement that an edge-safe path uses the same number of forward and backward spanning cycle edges enables the following simple yet important technical claim:

CLAIM 4.3. *Let q, q' be paths in H that are safe for edges e, e' respectively, and which share an endpoint node y . If $q \neq q'$, then $e \neq e'$.*

PROOF. We prove the contrapositive. Suppose $e = e' =: (u, v)$. By Lemma 3.5 we have $w(u, v) < n/2$, and so no path safe for (u, v) can use more than n spanning cycle edges. Let $0 \leq s < n$ be the number of backward steps along the spanning cycle from v to y . Then both q, q' must end with a suffix of exactly s backward steps along the spanning cycle. So they also begin with a prefix of exactly s forward steps along the spanning cycle, ending at u , implying equality. ■

In the same way that the Moore bounds focus on paths made up of k edges, a natural proof attempt to bound lightness would be to focus on paths made up of k edge-safe subpaths:

DEFINITION 4.4 (Safe k -Paths). A path π in H is a *safe k -path* if it can be partitioned into k subpaths $\pi = q_1 \circ \dots \circ q_k$, where each path q_i is safe for an edge e_i . We say that π is an *extra-safe k -path* if each path q_i is extra-safe for e_i .

Unfortunately, this natural attempt breaks. Specifically, the dispersion lemma fails: it is possible to have two edge-simple safe k -paths that share endpoints, without implying that H has a cycle of small normalized weight. This specifically occurs in the following figure:

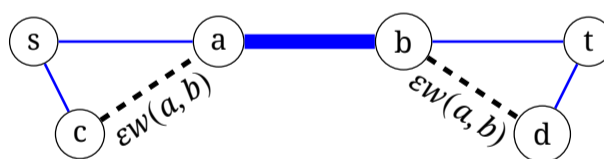


Figure 3. A counterexample to the dispersion lemma for safe k -paths

In Figure 3, the solid blue lines represent non-spanning-cycle edges, and we imagine that (a, b) is much heavier than the other four such edges. The dashed lines represent spanning cycle subpaths of length $\varepsilon w(a, b)$ each. This graph contains two different safe 3-paths with endpoints s, t , which are

$$\pi_1 = (s, a, b, t), \pi_2 = (s, c, \dots, a, b, \dots, d, t).$$

However, neither of the two cycles

$$C_1 = (s, a, c, \dots, s), C_2 = (t, b, \dots, d, t)$$

necessarily have small normalized weight. For example, the normalized weight of C_1 is

$$w^*(C_1) = \frac{w(s, c) + w(s, a) + \varepsilon w(a, b)}{\max\{w(s, c), w(s, a)\}}.$$

This quantity can be unbounded, specifically if $w(a, b) \gg \max\{w(s, c), w(s, a)\}$.

The conceptual point made by [12, 15] is essentially that the counterexample in Figure 3 can be avoided if we narrow our focus to paths that do not have one edge (a, b) that is much heavier than the others. Indeed, the dispersion lemma holds for safe k -paths whose non-spanning-cycle edge weights differ by at most a factor of 2 [12], or even by a factor of k [15]. Our following warmup proof makes the conceptual point that this counterexample can also be avoided if we narrow our focus to “monotone” paths, in which the non-MST edge weights may differ by a lot, but they must be increasing along the path.

4.2 Monotone Safe Paths and the Dispersion Lemma

Recall in the following definition that we have assumed for convenience that the non-spanning-cycle edges of H have distinct weights.

DEFINITION 4.5 (Monotone Safe k -Paths). Let π be a safe k -path, which can be partitioned into subpaths $\pi = q_1 \circ \dots \circ q_k$ where each q_i is safe for an edge e_i . We say that π is *monotone* if these edges are increasing in weight, that is, $w(e_1) < \dots < w(e_k)$.

LEMMA 4.6 (Monotone Dispersion Lemma). H may not have two distinct monotone safe k -paths with the same endpoints s, t .

PROOF. Seeking contradiction, let π^a, π^b be two distinct monotone safe k -paths with the same endpoints s, t . Let the decomposition of π^a into safe paths be $\pi^a = q_1^a \circ \dots \circ q_k^a$, where each subpath q_i^a is safe for the edge e_i^a . We use similar notation for π^b .

Let j be the largest index for which $q_j^a \neq q_j^b$. Since the subpaths following q_j^a, q_j^b are equal, q_j^a, q_j^b must share an endpoint node, which we will call y . By Claim 4.3, q_j^a, q_j^b are safe for distinct edges $e_j^a \neq e_j^b$. Assume without loss of generality that $w(e_j^a) > w(e_j^b)$. By monotonicity, it follows that $e_j^a \notin \pi_b[s \rightsquigarrow y]$. We can therefore find a cycle

$$C \subseteq \pi^a[s \rightsquigarrow y] \cup \pi^b[s \rightsquigarrow y]$$

in which e_j^a is the heaviest edge. We can bound the normalized weight of C as:

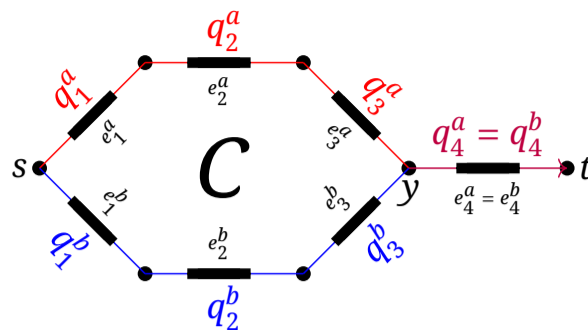


Figure 4. The setup in our proof of the monotone dispersion lemma; each colorful segment is an edge-safe path. The proof in this case takes $j = 3$ and argues that $w^*(C) \leq (1 + 2\varepsilon) \cdot 6$.

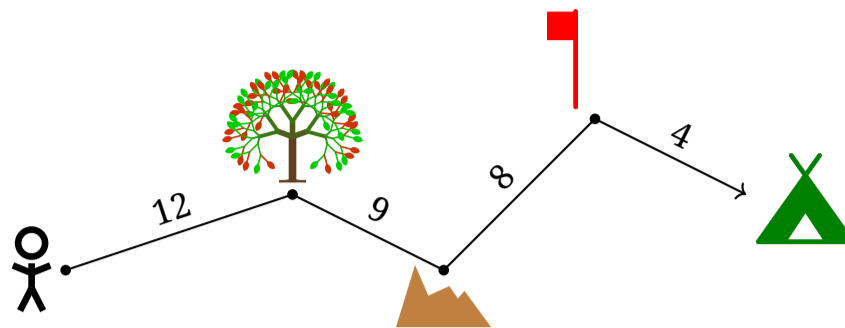
$$\begin{aligned}
 w^*(C) &\leq \frac{\sum_{i=1}^j w(q_i^a) + w(q_i^b)}{w(e_j^a)} \\
 &\leq \frac{\sum_{i=1}^j (1 + 2\varepsilon) (w(e_i^a) + w(e_i^b))}{w(e_j^a)} \\
 &\leq \frac{(j)(1 + 2\varepsilon) (2w(e_j^a))}{w(e_j^a)} \\
 &\leq (1 + 2\varepsilon) \cdot 2j.
 \end{aligned}$$

Since $j \leq k$, this contradicts that H has weighted girth $> (1 + 2\varepsilon) \cdot 2k$, completing the proof. ■

4.3 Hiker Paths and the Counting Lemma

The following is a famous puzzle in graph theory:

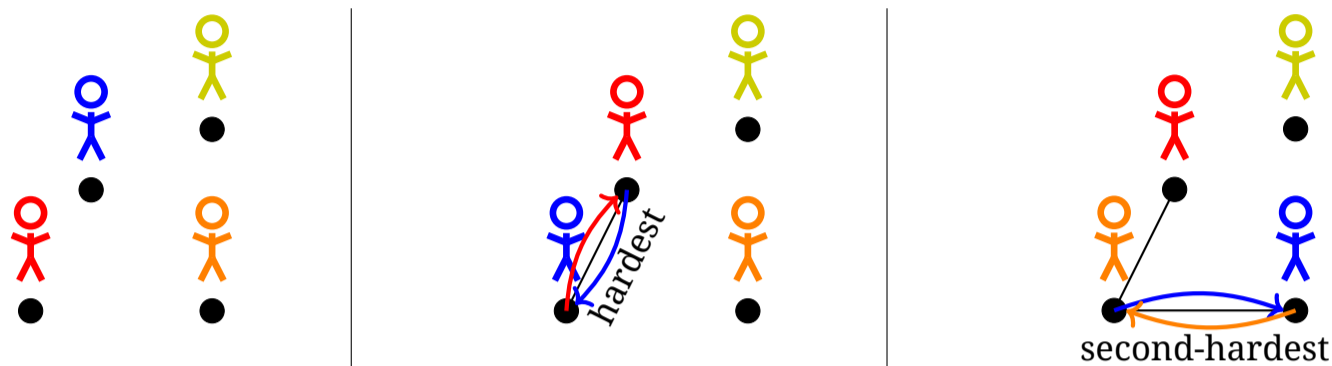
PUZZLE 4.7 (The Hiker Puzzle). We are vacationing in Graph National Park, which has n landmarks (nodes) and m trails (undirected edges) connecting pairs of landmarks. Each trail has a difficulty rating (weight). We would like to hike as many trails as possible, without repeating any trails. However, we will get increasingly tired as we hike, and so we are only willing to hike trails in nonincreasing order of difficulty. That is, after we hike a trail t_i , our next trail t_{i+1} must depart from the endpoint of t_i and its difficulty rating may not be higher than t_i 's difficulty rating. **Prove:** If we start at the right landmark, then we can hike at least $2m/n$ trails.



A valid hiker path, 4 trails long.

(This space is intentionally left free to not spoil the solution to Puzzle 4.7.)

SOLUTION 4.8. This problem, like so many others in life, is solved by asking our friends for help. We invite friends to join our vacation until we have gathered a group of n total hikers, and then we start with one hiker standing at each landmark. Consider the trails $t = (u, v)$ in descending order of difficulty, and ask the hiker currently standing at u and the hiker currently standing at v to hike the trail, switching places with each other.



In total, our n hikers will hike $2m$ trails, and so there must exist a hiker who hiked a path of length at least $2m/n$. This hiker hiked their trails in descending order of difficulty, and so their path satisfies the puzzle.

We will now prove counting lemmas for monotone safe k -paths. The medium and full counting lemmas from the Moore bounds generalize easily from Section 2, but the weak counting lemma needs a new idea, which is inspired by the hiker puzzle:

LEMMA 4.9 (Warmup Weak Counting Lemma). *If $w(H \setminus C) \geq \varepsilon^{-1}kn$, then H contains a monotone extra-safe k -path.*

PROOF. Start by placing a hiker at each node of H . Then, consider the non-spanning-cycle edges of H in order of increasing weight. When an edge (u, v) is considered, for each path π that is extra-safe for (u, v) , we ask the two hikers at either endpoint of π to hike π , thus switching places with each other.²

There are at least $\varepsilon w(u, v)/2$ paths that are extra-safe for each edge (u, v) , and two hikers hike each such path (one in each direction). Thus, after all non-spanning-cycle edges of H are considered, in total our n hikers have hiked at least $\varepsilon w(H \setminus C) \geq kn$ extra-safe paths, and the path walked by each hiker is a monotone extra-safe path. Thus, there exists a hiker who hiked a monotone extra-safe path of length at least k . ■

The bootstrapping process from the weak to the medium and full counting lemmas essentially works exactly as in the Moore bounds, with minor adjustments.

LEMMA 4.10 (Warmup Medium Counting Lemma). *H contains at least $\Theta(\varepsilon) \cdot (w(H \setminus C) - \varepsilon^{-1}kn)$ monotone safe k -paths.*

² A technical detail here is that each hiker walks at most one path π per edge (u, v) . This follows easily from Lemma 3.5, establishing that $w(u, v) < n/2$ and so any two extra-safe paths for (u, v) have disjoint endpoints.

PROOF. Repeat the following process until no longer possible. Find a monotone extra-safe k -path π , with decomposition

$$\pi = q_1 \circ \cdots \circ q_k$$

where each q_i is extra-safe for some edge e_i . Notice that, for any integer $0 \leq s \leq \varepsilon w(e_1)/2$, we can record a (not-necessarily-extra-)safe k -path by modifying π by adding s additional forward spanning cycle edges to the start of every path q_i , and also adding s additional backward spanning cycle edges to the end of every path q_i . We record $\Theta(\varepsilon w(e_1))$ monotone safe k -paths in this way.

We then delete the edge e_1 , to ensure that we do not re-record any of these paths in a future round. By the weak counting lemma, we may repeat this process at least until $w(H \setminus C) < \varepsilon^{-1}kn$. It follows that we will record at least $\Theta(\varepsilon) \cdot (w(H \setminus C) - \varepsilon^{-1}kn)$ monotone safe k -paths before halting. ■

LEMMA 4.11 (Warmup Full Counting Lemma). *Let $d := w(H \setminus C)/n$. If $d \geq 2k\varepsilon^{-1}$, then H contains at least $kn \cdot \Omega(\varepsilon d/k)^k$ monotone safe k -paths.*

PROOF. Let H' be a random edge-subgraph of H obtained by keeping the spanning cycle C deterministically, and keeping each non-spanning-cycle edge independently with probability $2k\varepsilon^{-1}/d$. Let p, p' be the number of monotone safe k -paths in H, H' , respectively. On one hand, monotonicity implies that any monotone safe k -path π uses k *distinct* non-spanning-cycle edges. Thus, the probability that π survives in H' is $\Theta(k\varepsilon^{-1}/d)^k$, and so

$$\mathbb{E}[p'] = p \cdot \Theta\left(\frac{k\varepsilon^{-1}}{d}\right)^k.$$

On the other hand, we have

$$\begin{aligned} \mathbb{E}[p'] &\geq \mathbb{E}\left[\Theta(\varepsilon) \cdot (w(H' \setminus C) - \varepsilon^{-1}kn)\right] && \text{Medium Counting Lemma} \\ &= \Theta(\varepsilon) \cdot (\mathbb{E}[w(H' \setminus C)] - \varepsilon^{-1}kn) \\ &= \Theta(\varepsilon) \cdot \left(w(H \setminus C) \cdot \frac{2k\varepsilon^{-1}}{d} - \varepsilon^{-1}kn\right) \\ &= \Theta(\varepsilon) \cdot (2kn\varepsilon^{-1} - \varepsilon^{-1}kn) \\ &= \Theta(\varepsilon) \cdot (\varepsilon^{-1}kn) \\ &= \Theta(kn). \end{aligned}$$

Comparing the two previous bounds on $\mathbb{E}[p']$, we get

$$\Theta(kn) \leq \mathbb{E}[p'] \leq p \cdot \Theta\left(\frac{k\varepsilon^{-1}}{d}\right)^k.$$

Rearranging this inequality gives our desired inequality of

$$p \geq kn \cdot \Theta \left(\frac{d}{k\varepsilon^{-1}} \right)^k. \quad \blacksquare$$

We are now ready to complete the proof of Theorem 4.1. Let $d := w(H)/n$. If $d < 2k\varepsilon^{-1}$, then we have $w(H) = O(kn\varepsilon^{-1})$ and we are done. Otherwise, if $d \geq 2k\varepsilon^{-1}$, then we may apply the full counting lemma to say that H has $kn \cdot \Omega(\varepsilon d/k)^k$ monotone safe k -paths. Meanwhile, the dispersion lemma implies that H has $O(n^2)$ such paths. Comparing these estimates, we get

$$kn \cdot \Omega \left(\frac{d}{k\varepsilon^{-1}} \right)^k \leq O(n^2).$$

Rearranging terms in this inequality to isolate d , we get

$$d \leq O \left(\varepsilon^{-1} k^{(k-1)/k} n^{1/k} \right) = O \left(\varepsilon^{-1} k n^{1/k} \right),$$

and thus $w(H) = w(H \setminus C) + w(C) = nd + n = O(\varepsilon^{-1} k n^{1+1/k})$.

5. Full Proof: Light Spanners via Bucket-Monotone Paths

We are now ready to prove our main theorem:

THEOREM 5.1 (Main Theorem). *Let $\varepsilon > 0$, let k, n be positive integers, and let H be an n -node graph with a unit-weight spanning cycle C and weighted girth $> (1 + 4\varepsilon) \cdot 2k$. Then*

$$w(H) = O \left(\varepsilon^{-1} n^{1+1/k} \right).$$

By the reductions in Section 3, this implies Theorem 1.4. As in the warmup proof, we use 4ε rather than ε in the weighted girth purely for convenience, and we will arbitrarily define a forward and backward direction around the spanning cycle.

5.1 Bucket-Safe Paths and Full Proof Intuition

Our full proof will save a factor of k in lightness over the previous warmup proof. Before proceeding to this main proof, let us give some intuition on where exactly the previous proof is losing a factor of k . We will consider two extreme opposing cases for the structure of H , and see how our previous proof loses this k -factor in either one. In the following discussion we define the *bucket* $B_i \subseteq E(H)$ to be the set of edges in $H \setminus C$ with weights in the range $[2^i, 2^{i+1})$.

Special Case 1: Well-Separated Edge Weights. Let us imagine that we are in an extreme special case in which no two edges in $H \setminus C$ lie in the same bucket; in particular, any two edges e_1, e_2 have weights $w(e_1), w(e_2)$ that differ by a factor of at least 2. In this case, the k -factor loss from our previous argument is because we are too stingy in our definition of edge-safe

paths. We allowed only $s \leq \varepsilon w(e)$ spanning cycle edges before and after each spanning cycle edge e , but in fact our dispersion lemma would still hold if we allowed approximately $\varepsilon k w(e)$ such edges. This is because the separation of edge weights implies that the total number of non-spanning-cycle edges will *telescope* over the monotone path; e.g., revisiting the proof of Lemma 4.6 there will still be only $O(\varepsilon k w(e_j^a))$ in the cycle C that we analyze. These ultimately cost only a negligible $+O(\varepsilon k)$ term in the bound on weighted girth. This leads to our first piece of intuition on how to improve the argument:

We can budget $O(\varepsilon k 2^i)$ spanning cycle edges per bucket B_i , rather than $O(\varepsilon w(e))$ spanning cycle edges per edge e .

Special Case 2: Bucketed Edge Weights. Let us now imagine the opposite special case, in which all edges in $H \setminus C$ lie in the same bucket B_i . In this case, as discussed earlier, *monotonicity* is not needed in the proof; we can use safe k -paths that use edges from B_i in any order. For these (not-necessarily-monotone) safe k -paths, the weak counting lemma can be improved by a factor of k : they are guaranteed to exist once $w(H \setminus C) \geq \varepsilon^{-1}n$, rather than $w(H \setminus C) \geq \varepsilon^{-1}kn$. This leads to our second piece of intuition on how to gain a k -factor:

The non-spanning-cycle edges within any bucket B_i do not need to occur in monotone order.

Bucket-Safe Paths. Our full proof will use *bucket-safe paths*, which replace the *edge-safe paths* from the previous warmup proof:

DEFINITION 5.2 (Bucket-Safe Paths). A path π in H is safe for bucket B_i if it is non-backtracking (meaning that it does not repeat any edge twice in a row), all of its non-spanning-cycle edges are in B_i , and for some integer $0 \leq s \leq \varepsilon k 2^i$ it contains exactly $2s$ spanning cycle edges, where the first s are used in the forward direction and the last s are used in the backward direction. We say that π is extra-safe if in fact $s \leq \varepsilon k 2^{i-1}$.

We note that any empty (single-node) path is considered to be safe for every bucket. Bucket-safe paths incorporate both pieces of intuition: they allow the edges within bucket B_i to occur in any order, and they budget $\varepsilon k 2^i$ non-spanning-cycle edges for this bucket. The following technical claim extends Claim 4.3 in the natural way to bucket-safe paths, and it has essentially the same proof:

CLAIM 5.3. *Let q, q' be bucket-safe paths in H , let σ, σ' be the sequences of non-spanning-cycle edges used by q, q' respectively, and suppose that q, q' share an endpoint node y . If $q \neq q'$, then $\sigma \neq \sigma'$.*

PROOF. We will prove the contrapositive. Suppose that

$$\sigma = \sigma' =: ((u_1, v_1), \dots, (u_j, v_j)).$$

For all pairs of nodes v_i, u_{i+1} , and also for v_j, y , there are two possible spanning cycle paths between these nodes (going around C in either direction), and at least one of these paths must use $\geq n/2$ spanning cycle edges. By Lemma 3.5 the maximum edge weight in H is $W < \frac{n}{4\epsilon k}$, and thus any bucket-safe path uses at most $2\epsilon kW < n/2$ spanning cycle edges. Thus q, q' must choose the same spanning cycle paths between each of these pairs of nodes.

This shows that q, q' are identical on their suffix from u_1 to y . In particular q, q' use the same number of forward and backward steps following u_1 . By definition bucket-safe paths must use the same number of forward and backward steps overall, which implies that they also use the same spanning cycle paths as prefixes from their start node up to u_1 . This implies $q = q'$. ■

Our plan is to use bucket-safe paths as building blocks to construct *bucket-monotone paths*, which are formed by concatenating bucket-safe paths in monotonic order of bucket weight, exactly like we previously constructed monotone safe paths by concatenating edge-safe paths.

5.2 Bucket-Monotone Paths and the Dispersion Lemma

We focus on the following paths in our proof:

DEFINITION 5.4 (Bucket-Monotone Safe k -Paths). A path π in H is a bucket-monotone safe k -path if it has exactly k non-spanning-cycle edges in total, and it can be partitioned into (possibly empty) subpaths $\pi = q_0 \circ \dots \circ q_j$, where each subpath q_i is safe for bucket B_i (and so these bucket weights are increasing along π). We say that π is extra-safe if each subpath q_i is extra-safe for B_i .

We note that, although each individual bucket-safe path q_i is non-backtracking, a bucket-monotone safe k -path may backtrack, e.g. when edges from q_{i+1} backtrack edges from q_i . This will be used in the counting lemma. The dispersion lemma for bucket-monotone safe k -paths is very similar to the one from the warmup proof.

LEMMA 5.5 (Dispersion Lemma). H may not have two distinct bucket-monotone safe k -paths with the same endpoints s, t .

PROOF. Seeking contradiction, let π^a, π^b be two distinct bucket-monotone safe k -paths with the same endpoints s, t . Let the decomposition of π^a into bucket-safe paths be $\pi^a = q_0^a \circ q_1^a \circ \dots$, and let σ_i^a be the (possibly empty) sequence of non-spanning-cycle edges used in q_i^a . We use similar notation for π^b .

Let j be the last index on which $q_j^a \neq q_j^b$. Note that q_j^a, q_j^b share an endpoint node, which we will call y . Then by Claim 5.3, we have $\sigma_j^a \neq \sigma_j^b$. Thus the $s \rightsquigarrow y$ prefixes of π^a, π^b are distinct, and in particular there exists a cycle

$$C \subseteq \pi^a[s \rightsquigarrow y] \cup \pi^b[s \rightsquigarrow y]$$

where C contains at least one edge from $\sigma_j^a \cup \sigma_j^b$. Let e^* be the heaviest edge in C . Our goal is now to bound $w(C)$, and it will be helpful to separately count the contribution of the spanning cycle and non-spanning-cycle edges.

Non-Spanning-Cycle Edges: Since π^a, π^b are bucket-safe k -paths, together they contain at most $2k$ non-spanning-cycle edges, each of weight $\leq w(e^*)$. So these contribute at most $2kw(e^*)$ to $w(C)$.

Spanning Cycle Edges: Since each subpath q_i^a, q_i^b is safe for bucket B_i , it contains at most $\varepsilon k 2^{i+1}$ spanning cycle edges. So the total number of spanning cycle edges in $\pi^a[s \rightsquigarrow y] \cup \pi^b[s \rightsquigarrow y]$ is at most

$$2 \cdot \sum_{i=1}^j \varepsilon k 2^{i+1} < \varepsilon k 2^{j+3}.$$

Finally, since e^* is in bucket B_j , we have $w(e^*) \geq 2^j$. Putting the parts together, we have

$$w^*(C) = \frac{w(C)}{w(e^*)} < \frac{2kw(e^*) + \varepsilon k 2^{j+3}}{w(e^*)} \leq 2k + 8\varepsilon k = (1 + 4\varepsilon) \cdot 2k.$$

This contradicts that H has weighted girth $> (1 + 4\varepsilon) \cdot 2k$, which completes the proof. ■

5.3 New Hiker Paths and the Counting Lemma

Much like the warmup proof, the medium and full counting lemmas generalize easily from the Moore bounds, but a new conceptual idea is needed for the weak counting lemma. For intuition, and for fun, we will make up an extension of the previous hiker puzzle that captures the gist of how our new weak counting lemma extends the one from the warmup proof.

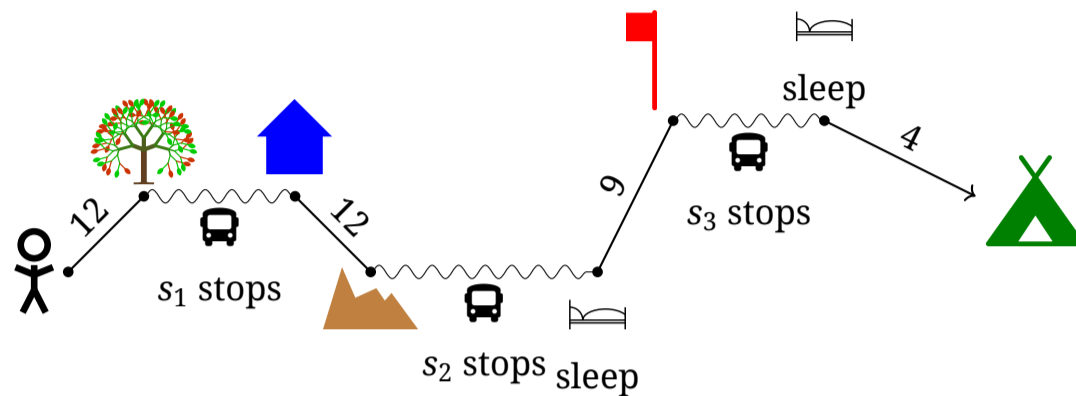
PUZZLE 5.6 (Another Hiker Puzzle). Graph National Park has n landmarks (nodes), m trails (undirected edges), and each trail has a difficulty rating (weight). It is possible for several trails to receive the same difficulty rating. The park has also installed a shuttle system, with a fleet of shuttles that drive in both the forward and backward direction in a loop (spanning cycle) around the landmarks. We have a Visitor's Pass that lets us ride the shuttle for up to $2t$ stops per day, which can be split across several trips if we like.

We are planning a multi-day backpacking trip to the park. At any time we may camp overnight at a landmark. We have the following constraints:

- To avoid boredom, we must hike at least one trail per day. We may also never *backtrack* a trail, meaning that after we hike a trail, our next action cannot be to immediately hike the same trail again in the reverse direction. (But we can otherwise repeat trails, e.g., by riding the shuttle back to the start of the trail and hiking it again.)

- Over the entire trip, we must hike all trails in nonincreasing order of difficulty. Additionally, each time we camp we wake up sore, and so the trails we hike on the following day must be *strictly* easier than all trails hiked on the previous day.

Prove: If we start at the right landmark, we can hike at least $2mt/n$ trails.



A valid hiker path, 4 trails long and split over 3 days, assuming $s_1 + s_2 \leq 2t$ and $s_3 \leq 2t$.

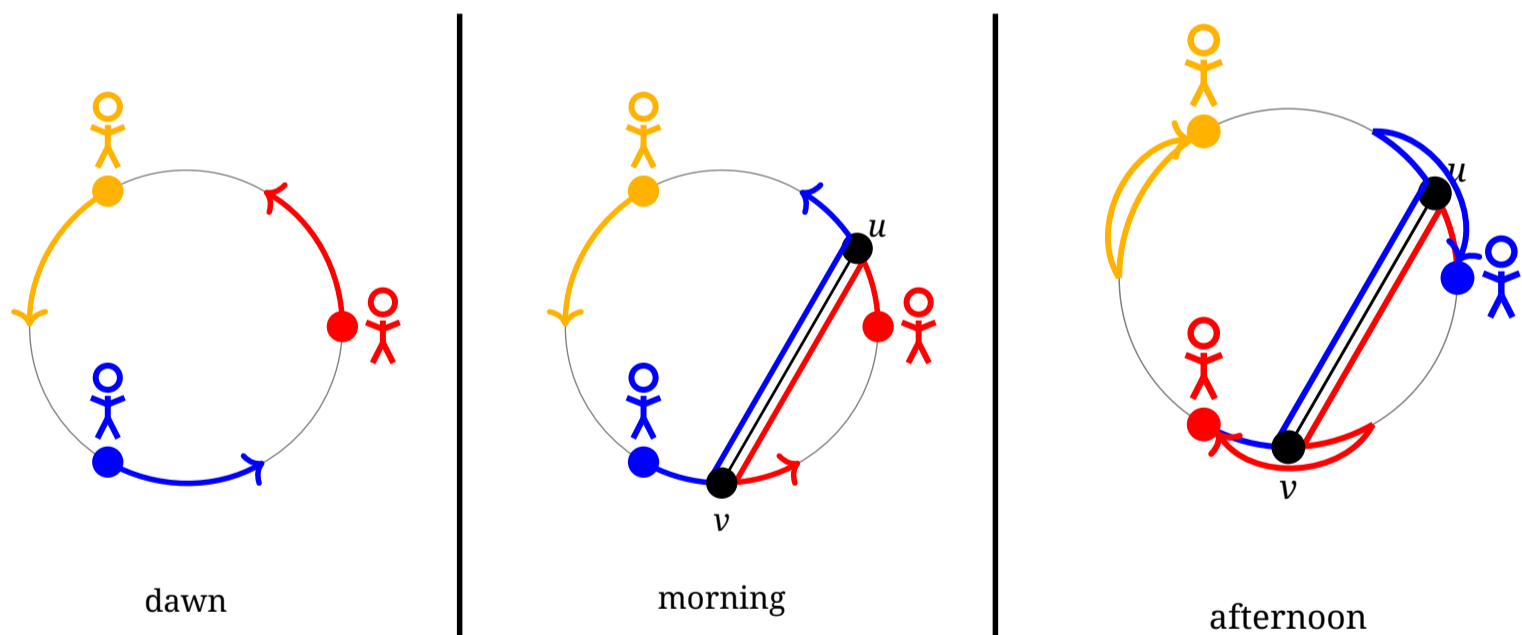
(This space is intentionally left free to not spoil the solution to Puzzle 5.6.)

SOLUTION 5.7. As in our previous hiker puzzle, we assemble a squad of n hikers, and we start with one hiker standing at each landmark. We plan out hiker paths by the following process.

— First, we group the trails by difficulty rating. Let B_0 be all the trails that are tied for highest difficulty rating, let B_1 be the trails that are tied for second-highest difficulty rating, etc.

We will plan hiker paths for each day i using the following steps:

- **(Dawn)** At dawn, each hiker plans a path that consists of riding the shuttle t stops in the forward direction from their current position (but they do not yet walk their planned path). Notice an **invariant**: for each node v , and for each integer $1 \leq s \leq t$, there exists a hiker who reaches v as the endpoint of their s^{th} shuttle stop. As we modify paths in the following step, this invariant will remain.
- **(Morning)** Next, the hikers collectively consider the trails $(u, v) \in B_i$ one at a time, in arbitrary order. For each integer $1 \leq s \leq t$, consider the two hikers h_u, h_v who currently plan to reach the nodes u, v as their s^{th} shuttle stop. We insert (u, v) into the paths of these two hikers right after their s^{th} shuttle stops. Following (u, v) , the two hikers swap paths: h_u takes h_v 's previous path from v onward, and h_v takes h_u 's previous path from u onward.
- **(Afternoon)** Finally, in the afternoon, the hikers all travel the paths they have planned out. Then they end the day by riding the shuttle t steps backward and camping at the landmark they reach.



Each trail (u, v) is inserted into the paths of $2t$ hikers. So our n hikers hike $2mt$ trails in total, and so there is a hiker who hikes at least $2mt/n$ trails. There is one last detail: we are supposed to ensure that this hiker hikes at least one trail per day. Notice that, if a hiker does not hike any trails on day i , then their planned path for the day consists of riding the shuttle t stops forward, riding the shuttle t stops backward, and camping back at the same landmark where their day began. The hiker can therefore skip this day entirely in their itinerary, and so they hike at least one trail in each un-skipped day.

We now give our weak counting lemma, which is more or less a repeat of the hiker path protocol from the above puzzle, with some adaptations and slightly more formal exposition of some steps.

LEMMA 5.8 (Weak Counting Lemma). *If $w(H \setminus C) \geq 4\epsilon^{-1}n$, then H contains a bucket-monotone extra-safe k' -path, for some $k' \geq k$.*

PROOF. Start by placing a hiker at each node of H . We will define a protocol to generate paths for these hikers, which loops on three phases; each loop is called a **day**.

- **(Dawn)** On day i at dawn, each hiker plans a path that consists of walking $t_i := \lfloor \epsilon k 2^{i-1} \rfloor$ steps forward on the spanning cycle (but they do not walk these edges yet). We notice that the following invariant holds:

Invariant: for each node v , for each integer $1 \leq s \leq t_i$ there exists a hiker who plans to reach v as the endpoint of the s^{th} spanning cycle edge that they walk, and also there exists a hiker who plans to end their journey at v .

- **(Morning)** On day i in the morning, the hikers collectively consider the edges $(u, v) \in B_i$ one at a time, in arbitrary order. For each integer $1 \leq s \leq t_i$, by the invariant, we may let h_u, h_v be the two hikers who respectively reach u, v at the end of their s^{th} forward spanning cycle steps. We insert the edge (u, v) into the planned paths of these two hikers immediately after their s^{th} spanning cycle steps. Following this edge, these hikers plan to swap paths; h_u takes the path suffix previously planned by h_v from v onward, and h_v takes the path suffix previously planned by h_u from u onward.

Note that each suffix swap preserves the invariant, since each $(s+1), \dots, t_i$ spanning cycle edge used by h_u becomes the $(s+1), \dots, t_i$ spanning cycle edge used by h_v (and vice versa), and the node at which h_u plans to end their journey becomes the node at which h_v plans to end their journey (and vice versa).

- **(Afternoon)** Finally, for each hiker h , let f_h be the number of contiguous forward spanning cycle edges that form a suffix of their planned path. We replace this suffix with $t_i - f_h$ backward steps on the spanning cycle. We can think of this step as adding a suffix of t_i backward steps on the spanning cycle to the end of each hiker's path, and then repeatedly canceling adjacent forward and backward steps that backtrack each other.

Each hiker then walks their planned path, and then ends the day; we then proceed to day $i+1$ and continue until all buckets have been considered.

For correctness of this hiker protocol, we note that by the invariant, it remains true that one hiker stands at each node at the end of the day and thus the invariant holds again in the following day at dawn. This construction implies that the path hiked by each hiker on day i

is extra-safe for B_i . Thus, each hiker's overall journey (concatenating their path across each day) forms a bucket-monotone extra-safe path. The number of hikers who traverse an edge $(u, v) \in B_i$ is

$$2t = 2\lfloor \varepsilon k 2^{i-1} \rfloor \geq \frac{\varepsilon k 2^{i+1}}{4} \geq \varepsilon k \cdot \frac{w(u, v)}{4}.$$

Summing over all edges in $H \setminus C$, the total number of non-spanning-cycle edges traversed by our n hikers is therefore at least

$$\sum_{(u,v) \in E(H \setminus C)} \varepsilon k \cdot \frac{w(u, v)}{4} = \varepsilon k \cdot \frac{w(H \setminus C)}{4} \geq kn.$$

So there exists a hiker who hikes a bucket-monotone extra-safe k' -path, for some $k' \geq k$. ■

The medium and full counting lemmas can now be bootstrapped from the weak counting lemma by the same method as in our previous proofs.

LEMMA 5.9 (Medium Counting Lemma). *H contains at least $\Theta(\varepsilon) \cdot (w(H \setminus C) - 4\varepsilon^{-1}n)$ bucket-monotone safe k -paths.*

PROOF. Repeat the following process until no longer possible. Find a bucket-monotone extra-safe k' -path π , for some $k' \geq k$. We can use π to record a family of bucket-monotone (not-necessarily-extra-)safe k -paths by the following process:

- Let e_1, \dots, e_k be the first k non-spanning-cycle edges used by π . Truncate π immediately after e_k . After this truncation, the last path q_j in the decomposition might not end with a suffix of the appropriate number of backward edges, and so we add the appropriate number of backward edges as a suffix to π to restore the fact that q_j is extra-safe for B_j .
- Let B_i be the bucket that contains e_1 , and let B_j be the bucket that contains e_k . Thus, omitting empty paths at the beginning and end, we may write the decomposition of π into bucket-extra-safe paths as $\pi = q_i \circ \dots \circ q_j$, where each subpath q_x is extra-safe for bucket B_x .
- For each integer $0 \leq s \leq \lfloor \varepsilon k 2^{i-1} \rfloor$, notice that we can generate a bucket-monotone safe k -path by modifying π by adding a prefix of s additional forward spanning cycle edges, and a suffix of s additional backward spanning cycle edges, to each nonempty path q_x in the decomposition of π .

We record $\lfloor \varepsilon k 2^{i-1} \rfloor = \Theta(\varepsilon k w(e_1))$ bucket-monotone safe k -paths in this way. We then delete e_1 to ensure that we do not re-record these paths in a future round. By the weak counting lemma, we can repeat this process at least until $w(H \setminus C) < 4\varepsilon^{-1}n$. So we record at least $\Theta(\varepsilon) \cdot (w(H \setminus C) - 4\varepsilon^{-1}n)$ paths in total. ■

In the full counting lemma for our warmup proof, we used the fact that monotone safe k -paths necessarily contain k distinct non-spanning cycle edges, by monotonicity. For our new

full counting lemma we will need the analogous fact is also true for bucket-monotone safe k -paths, as established in the following claim:

CLAIM 5.10. *Every bucket-monotone safe k -path π uses k distinct non-spanning-cycle edges.*

PROOF. Suppose for contradiction that there is a repeated non-spanning-cycle edge (u, v) in π . Let B_i be the bucket that contains (u, v) , and let q_i be the subpath of π that is safe for B_i . Notice that q_i must contain both copies of (u, v) , and so it has a cycle C as a subpath between these copies. Let e^* be the heaviest edge in C . We may bound the cycle weight $w(C)$ as follows. First, there are at most k non-spanning-cycle edges in C , and so the total contribution of these edges to $w(C)$ is at most

$$\sum_{(u,v) \text{ non-sp-cyc edge in } C} w(u, v) \leq k \cdot w(e^*).$$

Meanwhile, the number of spanning cycle edges in C is at most $\varepsilon k 2^i$. Putting these together, we have

$$w^*(C) \leq \frac{w(C)}{w(e^*)} \leq \frac{k \cdot w(e^*) + \varepsilon k 2^i}{w(e^*)} \leq k + 2\varepsilon k = (1 + 2\varepsilon)k.$$

This contradicts the weighted girth of H . ■

LEMMA 5.11 (Full Counting Lemma). *Let $d := w(H \setminus C)/n$. If $d \geq 5\varepsilon^{-1}$, then H contains at least $n \cdot \Omega(\varepsilon d)^k$ bucket-monotone safe k -paths.*

PROOF. Let H' be a random edge-subgraph of H obtained by keeping the spanning cycle C deterministically, and keeping each non-spanning-cycle edge independently with probability $5\varepsilon^{-1}/d$. Let p, p' be the number of bucket-monotone safe k -paths in H, H' respectively.

On one hand, by Claim 5.10, every such path π uses k distinct non-spanning-cycle edges. Thus, the probability that π survives in H' is $\Theta(\varepsilon^{-1}/d)^k$, and so

$$\mathbb{E}[p'] = p \cdot \Theta\left(\frac{\varepsilon^{-1}}{d}\right)^k.$$

On the other hand, we have

$$\begin{aligned} \mathbb{E}[p'] &\geq \mathbb{E}\left[\Theta(\varepsilon) \cdot \left(w(H' \setminus C) - 4\varepsilon^{-1}n\right)\right] && \text{Medium Counting Lemma} \\ &= \Theta(\varepsilon) \cdot \left(\mathbb{E}[w(H' \setminus C)] - 4\varepsilon^{-1}n\right) \\ &= \Theta(\varepsilon) \cdot \left(w(H \setminus C) \cdot \frac{5\varepsilon^{-1}}{d} - 4\varepsilon^{-1}n\right) \\ &= \Theta(\varepsilon) \cdot \left(5\varepsilon^{-1}n - 4\varepsilon^{-1}n\right) \\ &= \Theta(\varepsilon) \cdot \left(\varepsilon^{-1}n\right) \\ &= \Theta(n). \end{aligned}$$

Comparing the two previous bounds on $\mathbb{E}[p']$, we get

$$\Theta(n) \leq \mathbb{E}[p'] \leq p \cdot \Theta\left(\frac{\varepsilon^{-1}}{d}\right)^k.$$

Rearranging now gives our desired inequality of $p \geq n \cdot \Theta(\varepsilon d)^k$. ■

We now complete the proof of Theorem 5.1 in the usual way. Let $d := w(H \setminus C)/n$. If $d < 5\varepsilon^{-1}$, then we have $w(H) = O(\varepsilon^{-1}n)$ and we are done. Otherwise, by the dispersion and full counting lemmas, the number of bucket-monotone safe k -paths in H is at least $n \cdot \Omega(\varepsilon d)^k$, and at most $O(n^2)$. So we have

$$\begin{aligned} n \cdot \Omega(\varepsilon d)^k &\leq O(n^2) \\ d &\leq O\left(\varepsilon^{-1}n^{1/k}\right) \end{aligned}$$

and thus $w(H) = w(H \setminus C) + w(C) = nd + n = O(\varepsilon^{-1}n^{1+1/k})$.

Acknowledgments

I am grateful to Michael Elkin for helpful technical comments on an earlier draft of this paper.

References

- [1] Reyan Ahmed, Greg Bodwin, Faryad Darabi Sahneh, Keaton Hamm, Mohammad Javad Latifi Jebelli, Stephen Kobourov, and Richard Spence. Graph spanners: a tutorial review. *Computer Science Review*, 37:100253, 2020. [DOI](#) (1)
- [2] Stephen Alstrup, Søren Dahlgaard, Arnold Filtser, Morten Stöckel, and Christian Wulff-Nilsen. Constructing light spanners deterministically in near-linear time. *Theoretical Computer Science*, 907:82–112, 2022. [DOI](#) (3)
- [3] Ingo Althöfer, Gautam Das, David Dobkin, Deborah Joseph, and José Soares. On sparse spanners of weighted graphs. *Discrete & Computational Geometry*, 9(1):81–100, 1993. [DOI](#) (1–4)
- [4] Sujoy Bhore, Balázs Keszegh, Andrey Kupavskii, Hung Le, Alexandre Louvet, Dömötör Pálvölgyi, and Csaba D Tóth. Spanners in planar domains via steiner spanners and non-steiner tree covers. *arXiv preprint arXiv:2404.05045*, 2024. [DOI](#) (4)
- [5] Greg Bodwin. An alternate proof of near-optimal light spanners. *2024 Symposium on Simplicity in Algorithms, SOSA 2024, Alexandria, VA, USA, January 8–10, 2024*, pages 39–55. SIAM, 2024. [DOI](#) (1)
- [6] Greg Bodwin and Jeremy Flics. A lower bound for light spanners in general graphs. *arXiv preprint arXiv:2406.04459*, 2024. [DOI](#) (3)
- [7] Glencora Borradaile, Hung Le, and Christian Wulff-Nilsen. Greedy spanners are optimal in doubling metrics. *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019*, pages 2371–2379. SIAM, 2019. [DOI](#) (4)
- [8] Glencora Borradaile, Hung Le, and Christian Wulff-Nilsen. Minor-free graphs have light spanners. *2017 IEEE 58th Annual Symposium on Foundations of Computer Science, FOCS 2017*, pages 767–778. IEEE, 2017. [DOI](#) (4)
- [9] T-H Hubert Chan and Anupam Gupta. Small hop-diameter sparse spanners for doubling metrics. *Discrete & Computational Geometry*, 41:28–44, 2009. [DOI](#) (4)
- [10] T-H Hubert Chan, Mingfei Li, and Li Ning. Sparse fault-tolerant spanners for doubling metrics with bounded hop-diameter or degree. *Algorithmica*, 71(1):53–65, 2015. [DOI](#) (4)
- [11] T-H Hubert Chan, Mingfei Li, Li Ning, and Shay Solomon. New doubling spanners: better and simpler. *SIAM Journal on Computing*, 44(1):37–53, 2015. [DOI](#) (4)

- [12] Barun Chandra, Gautam Das, Giri Narasimhan, and José Soares. New sparseness results on graph spanners. *Int. J. Comput. Geom. Appl.* 5:125–144, 1995. [DOI](#) (2, 3, 11)
- [13] Shiri Chechik and Christian Wulff-Nilsen. Near-optimal light spanners. *ACM Transactions on Algorithms (TALG)*, 14(3):1–15, 2018. [DOI](#) (2, 3, 7)
- [14] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. Introduction to algorithms. MIT press, 2022. (6)
- [15] Michael Elkin, Ofer Neiman, and Shay Solomon. Light spanners. *SIAM Journal on Discrete Mathematics*, 29(3):1312–1321, 2015. [DOI](#) (2, 3, 6, 7, 11)
- [16] Michael Elkin and Shay Solomon. Fast constructions of lightweight spanners for general graphs. *ACM Transactions on Algorithms (TALG)*, 12(3):1–21, 2016. [DOI](#) (3)
- [17] Michael Elkin and Shay Solomon. Optimal euclidean spanners: really short, thin, and lanky. *Journal of the ACM (JACM)*, 62(5):1–45, 2015. [DOI](#) (3)
- [18] Paul Erdős. Extremal problems in graph theory. *Proceedings of the Symposium on Theory of Graphs and its Applications*, page 2936, 1963. (1, 2)
- [19] Arnold Filtser and Shay Solomon. The greedy spanner is existentially optimal. *SIAM Journal on Computing*, 49(2):429–447, 2020. [DOI](#) (2, 3)
- [20] Lee-Ad Gottlieb and Liam Roditty. An optimal dynamic spanner for doubling metric spaces. *European Symposium on Algorithms, pages 478–489*. Springer, 2008. [DOI](#) (4)
- [21] Omri Kahalon, Hung Le, Lazar Milenković, and Shay Solomon. Can’t see the forest for the trees: navigating metric spaces by bounded hop-diameter spanners. *Proceedings of the 2022 ACM Symposium on Principles of Distributed Computing, PODC 2022*, pages 151–162, 2022. [DOI](#) (4)
- [22] Hung Le and Shay Solomon. A unified framework for light spanners. *Proceedings of the 55th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2023*. ACM, 2023. [DOI](#) (2, 3, 7)
- [23] Hung Le and Cuong Than. Greedy spanners in euclidean spaces admit sublinear separators. *ACM Transactions on Algorithms*, 20(3):1–30, 2024. [DOI](#) (4)
- [24] Giri Narasimhan and Michiel Smid. Geometric spanner networks. Cambridge University Press, 2007. [DOI](#) (3)
- [25] David Peleg and Jeffrey Ullman. An optimal synchronizer for the hypercube. *SIAM Journal on Computing (SICOMP)*, 18(4):740–747, 1989. [DOI](#) (1)
- [26] David Peleg and Eli Upfal. A trade-off between space and efficiency for routing tables. *Journal of the ACM (JACM)*, 36(3):510–530, 1989. [DOI](#) (1)