

Are There Graphs Whose Shortest Path Structure Requires Large Edge Weights?

Aaron Bernstein ✉

Rutgers University, New Brunswick, NJ, USA

Greg Bodwin ✉

University of Michigan, Ann Arbor, MI, USA

Nicole Wein ✉

Simons Institute, Berkeley, CA, USA

Abstract

The *aspect ratio* of a (positively) weighted graph G is the ratio of its maximum edge weight to its minimum edge weight. Aspect ratio commonly arises as a complexity measure in graph algorithms, especially related to the computation of shortest paths. Popular paradigms are to interpolate between the settings of weighted and unweighted input graphs by incurring a dependence on aspect ratio, or by simply restricting attention to input graphs of low aspect ratio.

This paper studies the effects of these paradigms, investigating whether graphs of low aspect ratio have more structured shortest paths than graphs in general. In particular, we raise the question of whether one can generally take a graph of large aspect ratio and *reweight* its edges, to obtain a graph with bounded aspect ratio while preserving the structure of its shortest paths. Our findings are:

- Every weighted DAG on n nodes has a shortest-paths preserving graph of aspect ratio $O(n)$. A simple lower bound shows that this is tight.
- The previous result does not extend to general directed or undirected graphs; in fact, the answer turns out to be *exponential* in these settings. In particular, we construct directed and undirected n -node graphs for which any shortest-paths preserving graph has aspect ratio $2^{\Omega(n)}$.

We also consider the *approximate* version of this problem, where the goal is for shortest paths in H to correspond to approximate shortest paths in G . We show that our exponential lower bounds extend even to this setting. We also show that in a closely related model, where approximate shortest paths in H must also correspond to approximate shortest paths in G , even DAGs require exponential aspect ratio.

2012 ACM Subject Classification Theory of computation → Shortest paths

Keywords and phrases shortest paths, graph theory, weighted graphs

Digital Object Identifier 10.4230/LIPIcs.ITCS.2024.12

Related Version *Full Version:* <https://arxiv.org/abs/2308.13054>

Funding *Aaron Bernstein:* Funded by NSF CAREER grant 1942010 and Google Research Grant.

Greg Bodwin: Funded by NSF:AF 2153680

1 Introduction

In modern graph algorithms, a popular strategy is to *simplify* graphs in preprocessing before proceeding to the main part of the algorithm. Given an input graph G , the high-level goal is to reduce to solving the problem on a different graph H that faithfully encodes the important structural properties of G (and hence solving the problem over H instead of G yields a good solution), but where H has improved complexity measures relative to G in whatever sense is important for the runtime of the algorithm to follow. Some successful examples of this method in the literature include:



© Aaron Bernstein, Greg Bodwin, and Nicole Wein;
licensed under Creative Commons License CC-BY 4.0

15th Innovations in Theoretical Computer Science Conference (ITCS 2024).

Editor: Venkatesan Guruswami; Article No. 12; pp. 12:1–12:22

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

12:2 Are There Graphs Whose Shortest Path Structure Requires Large Edge Weights?

- **Edge Sparsifiers**, where the goal is for H to have substantially fewer edges than G . Well-studied examples of edge sparsifiers include spanners [25, 24], preservers [15, 1], flow/cut/spectral sparsifiers [29, 5], etc.
- **Vertex Sparsifiers**, where the goal is for H to have substantially fewer vertices than G . Well-studied examples of vertex sparsifiers include mimicking networks [11, 18], terminal minor sparsifiers [23], etc.
- **Hopsets and Shortcut Sets**, in which H has smaller *hop-diameter* than G while preserving its shortest path distances or reachabilities. Hop-diameter is an important complexity measure in parallel or distributed contexts [30, 31].
- **Expander Decomposition**, which reduces to solving problems on graphs with a favorable *expansion* parameter [21, 29].

This paper introduces a new paradigm for graph simplification, in which the goal is to minimize the complexity measure of *aspect ratio*. The aspect ratio of a graph is the multiplicative spread among its edge weights:

► **Definition 1 (Aspect Ratio)**. *Let $G = (V, E, w)$ be a graph with positive edge weights. Then its aspect ratio is the quantity $\frac{\max_e w(e)}{\min_e w(e)}$.*

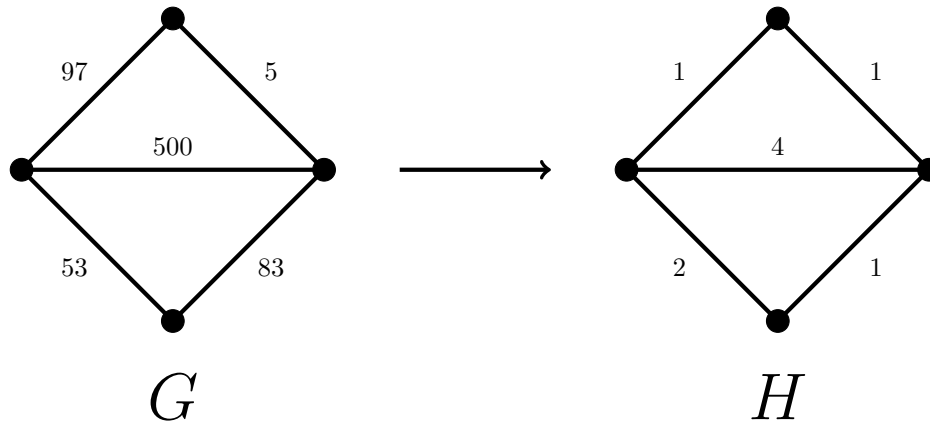
Aspect ratio often enters the picture for graph algorithms where the state-of-the-art solutions for *unweighted* graphs substantially outperform the solutions for *weighted* graphs. For these problems, one can sometimes extend the unweighted solution to graphs that are “close to unweighted.” Specifically, this could mean either (1) that the algorithm extends to the setting of weighted input graphs, but that the runtime or solution quality suffers a dependence on aspect ratio (say $\text{polylog } r$, for an input graph of aspect ratio r), or (2) more simply, that the algorithm only extends to the setting of weighted input graphs of bounded aspect ratio (say $\text{poly}(n)$, for n -node input graphs). These paradigms are especially common for problems related to computation of shortest paths or distances, for several reasons: when aspect ratio is bounded one can use bucketing methods to group shortest paths by length, and algebraic techniques based on fast matrix multiplication only give speedups when the associated matrix has bounded entries. Some concrete examples of shortest-path-related problems that have incurred a dependence on aspect ratio for these reasons include $(1 + \epsilon)$ -hopsets [7, 22], roundtrip spanners [26, 32, 10], the All-Pairs Shortest Paths (APSP) problem [33, 28], dynamic shortest paths (e.g. [19, 6, 12]), distributed shortest paths (e.g. [17, 9]), and many others. In light of this, we believe that the following two questions are natural:

Main Questions (Informal)

- *Is it generally possible to decrease the aspect ratio of a graph without changing the structure of its shortest paths?*
- *Do graphs of bounded aspect ratio have more structured shortest paths than general graphs?*

A positive resolution to either question would shed light on aspect-ratio-sensitive graph algorithms. In fact, as we point out next, some kind of structure is guaranteed: exactly one of the two questions must be answerable in the affirmative. To explain this point, let us formalize our model:

► **Definition 2** (Shortest-Paths Preserving Graph – See Figure 1). *Given a graph $G = (V, E, w)$, a reweighted graph on the same vertex and edge set $H = (V, E, w_H)$ is shortest-paths preserving if, for every shortest path π in H , the sequence of nodes and edges along π is also a shortest path in G .*¹



■ **Figure 1** Here H has the same shortest paths as G , but its aspect ratio is improved from 100 to 4.

For an example of a potential use case for shortest-paths preservers, we would get a valid solution to APSP on a graph G if we instead computed APSP over a shortest-paths preserving graph H , and then used the output as a solution for G (and we could imagine leveraging the improved aspect ratio of H to improve the computation). We can restate our main questions formally using this definition:

Main Questions (Formal)

- Does every n -node graph G admit a shortest-paths preserving graph H with aspect ratio $\text{poly}(n)$?
- Are there n -node graphs G whose shortest-paths system² cannot be realized by any graph of aspect ratio $\text{poly}(n)$?

We can now formally point out that these questions are complements: a graph G that resists a low aspect ratio shortest-paths preserver is precisely one that cannot be realized by a low aspect ratio graph. Hence, such graphs (if they exist) imply that restricting to the setting of low aspect ratio confers additional structure on the shortest path systems to be analyzed. Our choice to focus on aspect ratio $\text{poly}(n)$ comes from the fact that, for many problems, the specific dependence on aspect ratio W is $\text{polylog } W$ (e.g. [7, 22, 26, 32, 10, 9, 17, 19, 12]), and so this translates to a $\text{polylog}(n)$ dependence.

In addition to the potential algorithmic insights surveyed above, this paper fits into a recent line on the combinatorics of *shortest path structures* [3, 2, 13, 14, 8]; it can be viewed as an investigation of the extremal shortest path systems that can only be induced by high

¹ If G has several tied-for-shortest $s \rightsquigarrow t$ paths, one could consider two different models: that *at least one* shortest $s \rightsquigarrow t$ path in G must be shortest in H (which is Definition 2), or that *all* shortest $s \rightsquigarrow t$ paths in G must be shortest in H . All of our upper and lower bounds work for both of these definitions, so we somewhat arbitrarily use the first one.

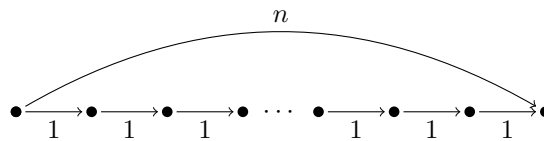
² Formally, the “shortest-paths system” associated to a graph G is the set of all vertex sequences that form shortest paths in G [13, 14, 8].

aspect ratio graphs. From this standpoint, an upper bound on aspect ratio would represent a new fact about the combinatorics of shortest path systems. Meanwhile, lower bounds would imply that *low-aspect-ratio* shortest path systems are fundamentally different objects than *general* shortest path systems, meaning their combinatorial properties could be independently investigated. We discuss this perspective in more depth in Section 1.4. More generally, there is virtually no current understanding of the interaction between shortest paths structure and aspect ratio, and the goal of this paper is to initiate this line of study.

1.1 Exact Shortest Path Preservers

A priori, our main questions can be asked independently in different graph classes: e.g., we could envision settling the dichotomy differently for directed graphs, undirected graphs, DAGs, etc. Our findings are that this is in fact the case: the answer does in fact change between these three graph classes.

Let us begin our discussion in the setting of DAGs. As a warmup observation, there is a simple way to construct n -node graphs that require aspect ratio $\Omega(n)$ for any shortest-paths preserving graph (see Figure 2). The lower bound is a path of n nodes and $n - 1$ edges of weight 1, plus an additional long edge of weight n that jumps from the start to the end of the path. Since the path of unit-weight edges is a shortest path, in reweighting, one cannot reduce the weight of the long edge below $n - 1$ times the minimum edge weight on the path. This implies that any shortest-paths preserving graph has aspect ratio $\Omega(n)$.



■ **Figure 2** Any shortest-paths preserving reweighting of this graph has aspect ratio $\Omega(n)$.

Our first result is that this simple lower bound is actually tight, over the class of DAGs:

► **Theorem 3** (Linear upper bound for DAGs). *Every n -node DAG has a shortest-paths preserving graph of aspect ratio $O(n)$.*

The fact that nontrivial aspect ratio upper bounds are possible might give hope that they generalize to an upper bound of $O(n)$, or perhaps $\text{poly}(n)$, for *all* directed graphs. However, our next result refutes this possibility: DAGs are special, and in general *exponential* aspect ratio is sometimes necessary. That is, despite the fact that only n^2 shortest paths need to be preserved, we discover that the interactions between these paths can compound on one another to force a chain reaction of edges with multiplicatively larger and larger weights. The consequence is a graph with “inherently” exponential aspect ratio.

► **Theorem 4** (Exponential lower bound for directed graphs). *There are n -node directed graphs G such that any shortest-paths preserving graph H has aspect ratio $2^{\Omega(n)}$.*

Next, we consider the undirected setting. The undirected setting is at least as easy as the directed setting: there is a black-box reduction showing that, if every directed graph has a shortest-paths preserver of aspect ratio at most $\alpha(n)$, then the same is true for undirected graphs [8].³ Nonetheless, we extend our lower bound to show that exponential aspect ratio is necessary for general undirected graphs as well.

³ To sketch the reduction: given an undirected graph G , convert it to a directed graph G' by replacing

► **Theorem 5** (Exponential lower bound for undirected graphs). *There are n -node undirected graphs G such that any shortest-paths preserving graph H has aspect ratio $2^{\Omega(n)}$.*

1.2 Shortest Path Preservers with Stretch

Given our hardness results for general graphs, it is next natural to investigate whether they can be overcome by allowing an approximation factor. We will consider the following model:

► **Definition 6** (α -stretch shortest-paths preserving graph). *Given a graph $G = (V, E, w_G)$, a reweighted graph $H = (V, E, w_H)$ is α -stretch shortest-paths preserving if every shortest path in H is also an α -approximate shortest path in G . That is, if π is an $s \rightsquigarrow t$ shortest path in H , then $w_G(\pi) \leq \alpha \cdot \text{dist}_G(s, t)$.*

This definition is generally weaker than the (exact) shortest-paths preserving graphs discussed previously, which is the special case of stretch 1. It represents a natural attempt to bypass the lower bounds in Theorems 4 and 5: can we reduce aspect ratio to $\text{poly}(n)$ in a more forgiving model, where we allow a stretch factor α in the shortest paths of G ?

We describe modifications of our lower bounds that are robust even to this approximate version of the problem. In the directed setting, we construct graphs that force exponential aspect ratio for *any* finite α (even depending on n):

► **Theorem 7** (Exponential lower bound for approximation in directed graphs). *For any α , there are n -node directed graphs G such that any α -stretch shortest-paths preserving graph H has aspect ratio $2^{\Omega(n)}$.*

For undirected graphs, we are able to modify our construction to obtain an exponential lower bound for the $(1 + \varepsilon)$ -approximate version:

► **Theorem 8** (Exponential lower bound for approximation in undirected graphs). *For any $\varepsilon \leq 13/12$, there are n -node undirected graphs G such that any $(1 + \varepsilon)$ -stretch shortest-paths preserving graph H has aspect ratio $2^{\Omega(n)}$.*

We have not attempted to optimize the constant $13/12$ in this theorem: it is almost certainly improvable, but only to a point (e.g. our methods will likely not be able to rule out shortest-paths preserving graphs with $\text{poly}(n)$ aspect ratio and stretch 2). It is an interesting open question whether one can obtain $\text{poly}(n)$ aspect ratio and $O(1)$ stretch in the undirected setting.

1.3 Shortest Path Preservers with Two-Sided Stretch

In the previous settings, we imagine computing *exact* shortest paths over H , and then mapping them to exact or approximate shortest paths in G . One might also imagine computing *approximate* shortest paths in H , and designing H such that these map to approximate shortest paths in G . The following definition captures the property that H would need to have to enable this method.

each undirected edge $\{u, v\}$ with both directed edges $(u, v), (v, u)$. Let H' be a shortest-paths preserving graph of G' with aspect ratio $\alpha(n)$. Convert H' to an undirected graph H by recombining edges $(u, v), (v, u)$ into a single undirected edge $\{u, v\}$ of weight $w_{H'}(u, v) + w_{H'}(v, u)$. One can calculate that this recombination step does not increase aspect ratio, and it preserves shortest paths.

► **Definition 9** ($(\alpha_H \rightarrow \alpha_G)$ -stretch shortest-paths preserving graph). *Given a graph $G = (V, E, w)$, a reweighted graph $H = (V, E, w_H)$ is said to be $(\alpha_H \rightarrow \alpha_G)$ -stretch shortest-paths preserving if every α_H -approximate shortest path in H is also an α_G -approximate shortest path in G .*

Thus, an α -stretch preserver in the previous sense is the same as a $(1 \rightarrow \alpha)$ -stretch preserver under this definition. We note that the problem is easier (more likely to admit an upper bound) as α_G increases, but harder as α_H increases. Thus, the previous lower bounds for directed and undirected graphs from Theorems 7 and 8 apply also to $(\alpha_H \rightarrow \alpha)$ -stretch preservers, for any $\alpha_H \geq 1$. However, the two-sided stretch version is generally incomparable in difficulty to the original (exact shortest paths) version of the problem.⁴ Thus, we can study this two-sided stretch question for DAGs: Do DAGs have $(\alpha_H \rightarrow \alpha_G)$ -stretch preservers with polynomial aspect ratio, for $\alpha_H, \alpha_G > 1$? Our final result is an answer to this question in the negative:

► **Theorem 10** (Exponential lower bound for two-sided approximation in DAGs). *For any $\alpha_H, \alpha_G > 1$, there is an n -node DAG G such that any $(\alpha_H \rightarrow \alpha_G)$ -stretch shortest-paths preserving graph H has aspect ratio $(\alpha_H)^{\Omega(\sqrt{n})}$.*

This implies a dichotomy: for the exact version on DAGs the answer is $O(n)$, while for the two-sided approximate version, the answer suddenly jumps to exponential.

1.4 Discussion: Aspect Ratio and Theory of Shortest Paths Structure

As mentioned previously, this work can be viewed as a study of the combinatorics of *shortest path systems*: we construct extremal shortest path systems that can only be expressed using large aspect ratio, or (in the case of DAGs) we prove that no such structures exist. In this sense, our work fits into a recent line of research in theoretical computer science and combinatorics that centers shortest path systems as objects exhibiting remarkable combinatorial structure. Some other work of this kind includes:

- Amiri and Wargalla [3] proved that, for DAGs G , it holds that every triple of nodes lies along a shortest path iff there is a *single* shortest path that covers all nodes in the graph. Extensions of this theorem to undirected and directed graphs were later established by Akmal and Wein [2].
- A developing line of work, initiated by Cizma and Linial [13, 14], studies the class of *geodesic graphs*, which are the graphs in which every complete consistent path complex⁵ can be induced as shortest paths by an edge weight function. Many graphs satisfying these condition have been discovered, as have obstructions to geodesy.
- Bodwin [8] classifies the combinatorial patterns that can or can't generally appear in the shortest path system induced by a graph with unique shortest paths.

Other results along these lines include [4, 16]. All of the above results study the combinatorial structure of shortest path systems that are induced by graphs with *arbitrary* weight functions. Our results suggest a way to incorporate edge weights into the discussion. In

⁴ On one hand, the exact version is harder because an exact shortest path in G must be an exact shortest path in H , whereas in the two-sided stretch version, an exact shortest path in G only needs to be an approximate shortest path in H . On the other hand, the two-sided stretch version is harder because *every* approximate shortest path in H must also be an approximate shortest path in G , whereas in the exact version there are no constraints on the approximate-but-not-exact shortest paths of H .

⁵ Formally, a path system is *complete* if it contains one path between each node pair, and it is *consistent* if it is closed under taking contiguous subpaths.

particular, for general directed and undirected graphs, our results show that the class of shortest paths systems induced by *low aspect-ratio* weight functions is a proper subset of the class of shortest path systems induced by *any* weight function. This therefore raises the open problem of how the structural results from prior work evolve if we restrict our attention to graphs with “simple” edge weights, in the sense of small aspect ratio. (Some other restrictions on the weight function could be interesting as well.)

1.5 Open Problems and Future Directions

Besides the upper bound for DAGs (Theorem 3), the findings of this paper point towards the message that *low aspect ratio graphs have additional structure in their shortest paths*. We conclude our introduction by mentioning three places in which there is still potential to settle our main dichotomy in the other direction. First: although we have placed focus on directed graphs, undirected graphs, and DAGs, there are many other notable graph classes in which one could seek upper bounds:

► **Open Question 1.** *Are there other notable graph classes that always admit shortest-paths preserving graphs of aspect ratio $\text{poly}(n)$, besides DAGs?*

Our second problem was mentioned previously, in the context of shortest-paths preservers with stretch for undirected graphs. Although we have constructed graphs for which there is no $(1 + \varepsilon)$ -stretch preserver of aspect ratio $\text{poly}(n)$, such a preserver of stretch $O(1)$ is still conceivable:

► **Open Question 2.** *Does every n -node undirected graph have a constant-stretch shortest-paths preserving graph of aspect ratio $\text{poly}(n)$?*

Although we focus on aspect ratio in this paper, the same basic questions apply to other complexity measures associated to graph edge weights. Perhaps most naturally, one can re-ask our questions for graphs of integer edge weights, parametrized by maximum edge weight:

► **Open Question 3.** *Does every n -node DAG have a shortest-paths preserving graph H with **integer** edge weights in the range $[1, \dots, \text{poly}(n)]$? If not, does this hold if we allow stretch α ?*

We note that this question is harder (less likely to admit an upper bound) than the one for aspect ratio, since any graph with integer edge weights in the range $[1, \dots, \text{poly}(n)]$ has aspect ratio $\text{poly}(n)$. Thus, the answer to the question is *no* in the settings of general directed or undirected graphs. But an affirmative answer for DAGs is still possible, and would for example allow us to extend the parallel/distributed results in Corollary 1.8 and 1.9 of [27] to work for arbitrary weighted graphs (assuming H could be computed efficiently).

Finally: this paper focuses on *extremal* questions, investigating the extent to which aspect ratio can be reduced for *all* graphs. One could also study an instance-optimal version of the problem, in which the goal is to compute a shortest-paths preserving graph of minimum aspect ratio:

► **Open Question 4.** *Is there a polynomial-time algorithm that takes an input graph G on input and computes a shortest-paths preserving graph (possibly with stretch) of minimum or near-minimum aspect ratio? If not, does this problem admit an approximation algorithm?*

1.6 Paper Organization

In Section 2 we prove our exponential lower bound for directed graphs, which serves as a warm-up for our exponential lower bound for undirected graphs, which appears in Section 3. In Sections 2 and 3 we also include the extensions of our exponential lower bounds to the approximate α -stretch version of the problem for directed and undirected graphs respectively. In Section 4 we prove our $O(n)$ upper bound for DAGs. Finally, in Section 5 we prove our exponential lower bound for the two-sided stretch version of the problem for DAGs.

Instead of a centralized technical overview, we include a high-level overview of our approach at the beginning of several of the individual sections.

2 Warm Up: Exponential Lower Bounds for Directed Graphs

We start by proving our lower bounds for general directed graphs, as these are simpler. We begin with the exact version, and later move to the approximate version. In the next section, we turn to undirected graphs. We first prove Theorem 4, restated below.

► **Theorem 4** (Exponential lower bound for directed graphs). *There are n -node directed graphs G such that any shortest-paths preserving graph H has aspect ratio $2^{\Omega(n)}$.*

High-Level Approach

It is clear from the example in Figure 2 that one can easily find graphs where any shortest-paths preserving reweighting has aspect ratio $\Omega(n)$. This example consists of a shortest path P of weight n , and a not-shortest path P' (in this case just a single edge) with the same endpoints as P . The fact that P' must remain not shortest after reweighting provides a lower bound on the sum of the weights of the edge(s) in P' .

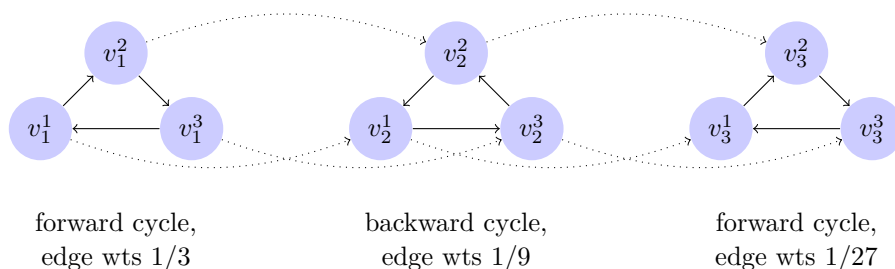
The key question towards getting a lower bound with larger aspect ratio is: can this approach be iterated? That is, we would like to carry out the following procedure:

1. Generate a collection of edges of at least some weight w .
2. Combine these edges into a collection \mathcal{P} of shortest paths where each path in \mathcal{P} has ℓ edges of weight w , for some ℓ .
3. Construct a collection \mathcal{P}' of *not* shortest paths with the same endpoints as the shortest paths in \mathcal{P} , such that each of the paths in \mathcal{P}' has $\ell' < \ell$ edges.
4. Now, in order for the paths in \mathcal{P}' to remain not-shortest after reweighting, their edge weights must be on average a multiplicative factor of ℓ/ℓ' larger than the weight w of the edges in \mathcal{P} . Now that we have generated a collection of edges of weight multiplicatively larger than w , we can return back to step 1 with a larger value of w .

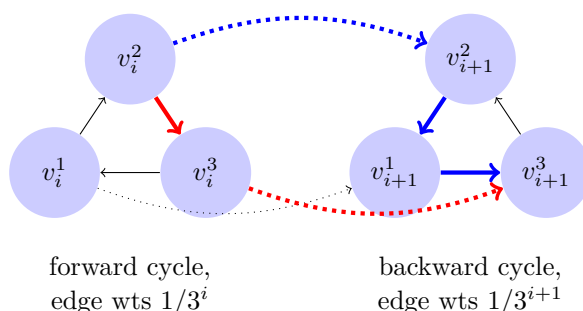
Several challenges arise when trying to construct a graph with these properties:

First, an edge constructed in some iteration i of this procedure has a very restricted set of properties: it must simultaneously be in a *not-shortest* path with the *same* endpoints as a shortest path from iteration i , and be in a *shortest* path in the next iteration $i + 1$. This requires a precise interleaving of various shortest and not-shortest paths, and it is unclear whether such a construction should exist.

Second, it is easy to imagine that if one uses k edges of weight w in the infrastructure of iteration i , one might only generate a smaller number, say $k/2$, edges of larger weight for the next iteration. This type of situation would not yield strong bounds because it would only allow $\log n$ iterations of the procedure, which would result in only polynomial aspect ratio. Thus, we need to make sure there is not too much loss in the number of edges we generate in each iteration (and in fact we achieve no loss).



■ **Figure 3** Our exponential aspect ratio lower bound for directed graphs (Theorem 4). The graph consists of $n/3$ cycles of length 3. The cross-cycle edges (gray, dotted) go between same-numbered nodes in adjacent cycles, and all have weight 0.



■ **Figure 4** Our analysis uses the fact that the shortest path between certain node pairs in adjacent cycles uses no edges from C_i and 2 edges from C_{i+1} (the blue path – see Lemma 11), rather than an alternate non-shortest path that uses 1 edge C_i and no edges from C_{i+1} (the red path). For example, in this diagram, the shortest $v_i^2 \rightsquigarrow v_{i+1}^3$ path is $(v_i^2, v_{i+1}^2, v_{i+1}^1, v_{i+1}^3)$ (thick, blue) and not $(v_i^2, v_i^3, v_{i+1}^3)$ (thick, red). This ultimately implies that, in any shortest-paths preserving graph, the blue path must remain shorter than the red path and so edge weights in C_i must be (on average) at least double of those in C_{i+1} (see Lemma 12).

We show how to construct a surprisingly simple graph that satisfies the constraints necessary to carry out the iterated procedure, and furthermore uses only 3 edges of average weight w (plus 3 helper edges) to generate 3 more edges whose average weight is at least $2w$. Because of this, we can perform $\Omega(n)$ iterations of the procedure, which yields an aspect ratio of $2^{\Omega(n)}$.

Now we will provide our construction and analysis in detail.

2.1 Construction

We construct the lower-bound graph $G = (V, E, w)$ as follows (see Figure 3)

- We assume that the number of vertices n is divisible by 3. The graph $G = (V, E, w)$ consists of $n/3$ cycles, each with exactly 3 vertices. Label the cycles $C_1, \dots, C_{n/3}$, where cycle C_i consists of vertices v_i^1, v_i^2, v_i^3 .
- Each cycle is either a *forward cycle* or a *backward cycle*. If C_i is a forward cycle, then the graph contains directed edges (v_i^1, v_i^2) , (v_i^2, v_i^3) and (v_i^3, v_i^1) . If C_i is a backward cycle then the graph contains directed edges (v_i^1, v_i^3) , (v_i^3, v_i^2) and (v_i^2, v_i^1) . The cycles alternate forward and backward. That is, C_i is forward for odd i and backward for even i .

12:10 Are There Graphs Whose Shortest Path Structure Requires Large Edge Weights?

- There are also edges between cycles, which we call *cross-cycle edges*. In particular, for every i , there is an edge (v_i^1, v_{i+1}^1) , as well as edges (v_i^2, v_{i+1}^2) and (v_i^3, v_{i+1}^3) . Note that edges only go *from* C_i to C_{i+1} ; there are no edges from C_i to any other C_j , and in particular there are no edges going from C_i to C_{i-1} . (That is, the graph is almost a DAG, except that the C_i themselves are cycles.)
- For any edge $e \in C_i$ we set $w(e) = 1/3^i$. We set the weight of all cross-cycle edges to 0^6

2.2 Analysis

The analysis is broken into two parts. First, we will identify a particular set \mathcal{P} of paths and in Lemma 11 we will show that they are unique shortest paths in G under weight function w . Then, in Lemma 12 we will show that any new weight function w' on G for which all paths in \mathcal{P} remain shortest must have aspect ratio $2^{\Omega(n)}$.

Construction of \mathcal{P}

See the blue path in Figure 4 for an example of a path in \mathcal{P} . All paths in \mathcal{P} go from a vertex in C_i to a vertex in C_{i+1} . Consider some cycle C_i . Let $v_i^k, v_i^{k'}$ be any two consecutive vertices in C_i : so if C_i goes forward then we have $(k, k') \in \{(1, 2), (2, 3), (3, 1)\}$, while if C_i goes backward then we have $(k, k') \in \{(1, 3), (3, 2), (2, 1)\}$.

For any such consecutive pair $v_i^k, v_i^{k'}$, we add the following path from v_i^k to $v_{i+1}^{k'}$ to \mathcal{P} : the path takes a cross-cycle edge from v_i^k to v_{i+1}^k , and then it follows C_{i+1} from v_{i+1}^k to $v_{i+1}^{k'}$. For example, if C_i is a forward cycle and the consecutive pair is v_i^2, v_i^3 , then we add to \mathcal{P} the following 3-edge path from v_i^2 to v_{i+1}^3 : $(v_i^2, v_{i+1}^2) \circ (v_{i+1}^2, v_{i+1}^1) \circ (v_{i+1}^1, v_{i+1}^3)$.

► **Lemma 11.** *Each path in \mathcal{P} is the shortest path between its endpoints in G .*

Proof. Fix a path $P \in \mathcal{P}$, where P is from v_i^k to $v_{i+1}^{k'}$ for some consecutive pair $v_i^k, v_i^{k'}$. It is easy to see that $w(P) = 2/3^{i+1}$, which corresponds to two edges on C_{i+1} (recall that the cross-cycle edge has weight 0). Let P' be an alternate path with the same endpoints as P . We will argue that $w(P') > w(P)$. Each edge in C_i is of weight at least $1/3^i$, so if P' uses an edge in C_i then indeed $w(P') \geq 1/3^i > w(P)$. The only edge from v_i^k that is not in C_i is the first edge of P : (v_i^k, v_{i+1}^k) , so P' must begin with this edge. From v_{i+1}^k , if P' uses another cross-cycle edge, it can never return to C_{i+1} since all cross-cycle edges go from some C_j to C_{j+1} . Therefore, the only edges that P' can take from v_{i+1}^k are edges in C_{i+1} . Because C_{i+1} is simply a directed cycle, there is only one path to $v_{i+1}^{k'}$, which is precisely the path that P takes. ◀

► **Lemma 12.** *Any new weight function w_H on G for which all paths in \mathcal{P} remain shortest has aspect ratio $2^{\Omega(n)}$.*

Proof. Let $H = (V, E, w_H)$ be a shortest-paths-preserving graph of G . We will show that the weight of each cycle in H is double that of its successor; that is, for all $i \in \{1, \dots, n/3 - 1\}$, we will show that $w_H(C_i) > 2 \cdot w_H(C_{i+1})$. This implies that the aspect ratio of w_H is $2^{\Omega(n)}$.

For any consecutive pair $v_i^k, v_i^{k'}$, we know that in H the 3-edge path in \mathcal{P} from v_i^k to $v_{i+1}^{k'}$ is shorter than the 2-edge path $(v_i^k, v_{i+1}^k) \circ (v_{i+1}^k, v_{i+1}^{k'})$. (These correspond to the blue and red paths respectively in Figure 4.) Taking the sum of this inequality for all three consecutive

⁶ If we want to keep edge-weights positive, then the same proof goes through if we set $w(e) = \delta$ for all cross-cycle edges; for the exact lower bound (Theorem 4) the value of δ does not matter, but for the approximate lower bound (Theorem 7) we would need to set δ to be very tiny.

pairs in C_i , we get $w_H(C_i) + X > 2w_H(C_{i+1}) + X$, where $X = \sum_{k=1}^3 w_H(v_i^k, v_{i+1}^k)$ is the sum (in H) of the weights of the cross-cycle edges from C_i to C_{i+1} . The X cancels, so we get $w_H(C_i) > 2w_H(C_{i+1})$, as desired. ◀

2.3 Approximate Version

We now show that in directed graphs, some graphs require exponential aspect ratio even if we allow arbitrary stretch. We prove Theorem 7, restated below:

► **Theorem 7** (Exponential lower bound for approximation in directed graphs). *For any α , there are n -node directed graphs G such that any α -stretch shortest-paths preserving graph H has aspect ratio $2^{\Omega(n)}$.*

Construction

The construction of G is exactly the same as for the exact version above, except all edges in C_i have weight $1/(2 \cdot \alpha)^i$ instead of $1/3^i$.

Analysis

We will use the same set \mathcal{P} of paths as for the exact version. We will show in Lemma 13 that each path in \mathcal{P} is not only the shortest path between its endpoints in G , but is also the *only* α -approximate shortest path between its endpoints in G . Since H is an α -stretch shortest-path preserving graph of G , this implies that each path in \mathcal{P} is the *unique* shortest path between its endpoints in H . We have already shown in Lemma 12 that any new weight-function w_H on G for which all paths in \mathcal{P} remain shortest paths has aspect ratio $2^{\Omega(n)}$. Thus, it only remains to prove Lemma 13:

► **Lemma 13.** *Each path in \mathcal{P} is the only α -approximate shortest path between its endpoints in G .*

Proof. Fix a path $P \in \mathcal{P}$. We know that P is a path from v_i^k to $v_{i+1}^{k'}$ for some consecutive pair $v_i^k, v_i^{k'}$. It is easy to see that $w(P) = \frac{2}{(2 \cdot \alpha)^{i+1}}$. Let P' be an alternate path with the same endpoints as P . We will argue that $w(P') > \alpha \cdot w(P)$.

Each edge in C_i is of weight at least:

$$\frac{1}{(2\alpha)^i} > \alpha \cdot \frac{2}{(2 \cdot \alpha)^{i+1}} = \alpha \cdot w(P),$$

so if P' uses an edge in C_i then indeed $w(P') > \alpha \cdot w(P)$. The only edge from v_i^k that is not in C_i is the first edge of P : (v_i^k, v_{i+1}^k) , so P' must begin with this edge. From v_{i+1}^k , if P' uses another cross-cycle edge, it can never return to C_{i+1} since all cross-cycle edges go from some C_j to C_{j+1} . Therefore, the only edges that P' can take from v_{i+1}^k are edges in C_{i+1} . Because C_{i+1} is simply a directed cycle, there is only one path to $v_{i+1}^{k'}$, which is precisely the path that P takes. ◀

3 Exponential Lower Bounds for Undirected Graphs

In this section, we prove our lower bounds for undirected graphs, which are somewhat more complicated. We start with the exact shortest-path preservers (theorem restated below), and turn to approximate ones in the next subsection.

► **Theorem 5** (Exponential lower bound for undirected graphs). *There are n -node undirected graphs G such that any shortest-paths preserving graph H has aspect ratio $2^{\Omega(n)}$.*

High-Level Approach

Our construction has the same basic structure as the lower bound for directed graphs in the previous section. The graph G will again consist of constant-length cycles C_1, \dots, C_k , and we will again construct the graph in such a way as to ensure that the shortest-path-preserver H must satisfy $w_H(C_i) > 2 \cdot w_H(C_{i+1})$; since the number of cycles is $\Omega(n)$, this implies that w_H has aspect ratio $2^{\Omega(n)}$.

At a high level, the argument in directed graphs relied on the fact that every path $P \in \mathcal{P}$ from C_i to C_{i+1} consists of 2 edges in C_{i+1} (plus a cycle-crossing edge), yet is shorter than an alternative path P' between the same endpoints which consists of 1 edge in C_i (plus a cycle-crossing edge) – see Figure 4. This implies that edges in C_i must have at least double the weights of those in C_{i+1} , as desired.

In order to construct an undirected graph which provides a similar guarantee, we need to overcome three new issues with this approach that are specific to undirected graphs.

The first issue is that in directed graphs we ensured that P has fewer edges than P' by alternating the direction of the cycles. But in undirected graphs, there can be no “forward” or “backward” cycles. We overcome this issue by introducing more structure to the cross-cycle edges, which ensures that every $P \in \mathcal{P}$ still uses two edges from C_{i+1} , and that there is still an alternative path P' using a single edge from C_i . In particular, the cross-cycle edges no longer simply go from each vertex in C_i to its copy in C_{i+1} .

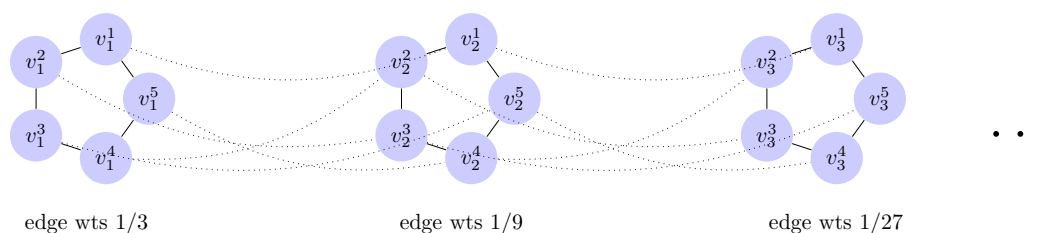
The second issue is that we need our construction to control the direction that the shortest path P will follow along the cycle. This is easily accomplished by increasing the size of each cycle to 5, so that following the 2 edges of C_{i+1} on P is shorter than following the 3 edges of C_{i+1} in the other direction.

Finally, the third issue is that in undirected graphs, there can also be paths from C_i to C_{i+1} that go through multiple levels of cycles (e.g. to C_{i+2}) and then return back to C_{i+1} . (In the directed construction, this could not happen because edges only pointed from lower to higher numbered cycles.) Because of this, we no longer have the freedom to set the cross-cycle edges to have weight 0. In fact, if they were weight 0 then there would be a path of weight 0 from every vertex to every other vertex in the graph, due to the newly defined cross-cycle edges from the first issue. Instead, we need to set the weights of the cross-cycle edges to be higher; for the exact lower bound we can simply set the cross-cycle weights to be very large, but for the approximate lower bound we need to balance them with the weights of the cycle edges. See the high-level description within the approximate version section (Section 3.3) for more details on this issue.

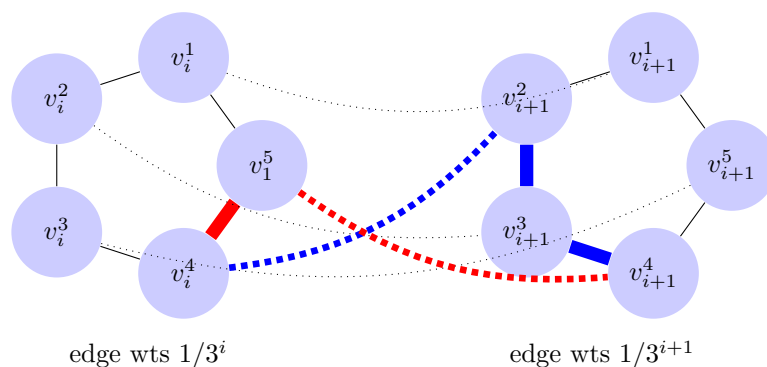
3.1 Construction

We construct our lower-bound graph $G = (V, E, w)$ as follows (see Figure 5):

- We assume that the number of vertices n is divisible by 5. The graph consists of $n/5$ cycles, each with exactly 5 vertices. Label the cycles $C_1, \dots, C_{n/5}$, where cycle C_i consists of vertices $v_i^1, v_i^2, \dots, v_i^5$, and edges $(v_i^1, v_i^2), (v_i^2, v_i^3), (v_i^3, v_i^4), (v_i^4, v_i^5)$, and (v_i^5, v_i^1) .
- For all i , the cross-cycle edges are as follows: $(v_i^1, v_{i+1}^1), (v_i^2, v_{i+1}^3), (v_i^3, v_{i+1}^5), (v_i^4, v_{i+1}^2)$, and (v_i^5, v_{i+1}^4) . The pattern of these edges is that as the superscript of the first vertex increases by 1, the superscript of the second vertex increases by 2 (mod 5).
- All edges in C_i have weight $1/3^i$. Meanwhile, all cross-cycle edges have weight 1.



■ **Figure 5** Our exponential aspect ratio lower bound in undirected graphs (Theorem 5). The graph consists of $n/5$ cycles of length 5. The cross-cycle edges (gray, dotted) go between **differently-numbered** nodes in adjacent cycles; the cross-cycle edges have weight 1. (The cross-cycle edges have weight $1/3^{i-1}$ for the α -stretch lower bound.)



■ **Figure 6** Just like in our directed lower bound, our analysis uses the fact that the blue shortest path uses 0 edges from C_i and 2 edges from C_{i+1} , while the alternate non-shortest red path uses 1 edge from C_i and 0 edges from C_{i+1} . For example, in this diagram, the shortest $v_i^4 \rightsquigarrow v_{i+1}^4$ path is $(v_i^4, v_{i+1}^2, v_{i+1}^3, v_{i+1}^4)$ (thick, blue) and not $(v_i^4, v_i^5, v_{i+1}^4)$ (thick, red).

3.2 Analysis

Similar to the directed case, the analysis is broken into two parts. First, we will identify a particular set \mathcal{P} of paths and in Lemma 14 we will show that they are unique shortest paths in G . Then, in Lemma 15 we will show that any reassignment of edge weights of G for which all paths in \mathcal{P} remain shortest has aspect ratio $2^{\Omega(n)}$.

Construction of \mathcal{P}

See the blue path in Figure 6 for an example of a path in \mathcal{P} . All paths in \mathcal{P} go from a vertex in C_i to a vertex in C_{i+1} . For every vertex v_i^k add to \mathcal{P} the following 3-edge path from v_i^k . Take the cross-cycle edge from v_i^k to the cycle C_{i+1} and let v_{i+1}^j be the other endpoint of the edge. Then take two cycle edges in C_{i+1} : first the edge to $v_{i+1}^{j+1 \pmod{5}}$ and then the edge to $v_{i+1}^{j+2 \pmod{5}}$.

► **Lemma 14.** *Each path in \mathcal{P} is the unique shortest path between its endpoints.*

Proof. Fix a path $P \in \mathcal{P}$. P is from v_i^k to $v_{i+1}^{j+2 \pmod{5}}$. We calculate $w(P)$ as follows: the first term is for the weight-1 cross-cycle edge, while the second term is the sum of the two edge weights in C_{i+1} .

$$w(P) = 1 + 2/3^{i+1}.$$

Let P' be an alternate path with the same endpoints as P . We will argue that $w(P') > w(P)$.

12:14 Are There Graphs Whose Shortest Path Structure Requires Large Edge Weights?

Any path from C_i to C_{i+1} must use a cross-cycle edge, which has weight 1. If P' uses another cross-cycle edge, P' gains an additional weight of 1. If P' uses an edge in C_i , P' gains weight $1/3^i$. In either of these cases, we have

$$w(P') \geq 1 + 1/3^i > 1 + 2/3^{i+1} = w(P).$$

Thus, the only case left to consider is when P' uses exactly one cross-cycle edge, and does not use any edges from C_i .

The only edge from v_i^k that is not in C_i is the first edge of P : (v_i^k, v_{i+1}^j) , so P' must begin with this edge. From v_{i+1}^j , the remainder of P' must be within C_{i+1} , since P' does not use another cross-cycle edge. Because C_{i+1} is a cycle, there are exactly two simple paths from v_{i+1}^j to $v_{i+1}^{j+2 \pmod{5}}$: P takes the one on 2 edges, while the other path goes in the other direction around the 5-cycle, so it has 3 edges and is longer than P . This completes the proof. \blacktriangleleft

► **Lemma 15.** *Any reassignment of edge weights to G such that all paths in \mathcal{P} remain shortest has aspect ratio $2^{\Omega(n)}$.*

Proof. Let $H = (V, E, w_H)$ be a shortest-paths-preserving graph of G . We will show that for all $i \in \{1, \dots, n/5 - 1\}$, we have $w_H(C_i) > 2 \cdot w_H(C_{i+1})$. Since there are $n/5$ cycles, this implies that the aspect ratio of w_H is $2^{\Omega(n)}$.

First, we claim that for every path $P \in \mathcal{P}$ from C_i to C_{i+1} , there is a not-shortest 2-edge path P' with the same endpoints as P , such that P' includes one edge in C_i and one cross-cycle edge. (P and P' correspond to the blue and red paths respectively in Figure 6.) The reason for this is by the construction of the cross-cycle edges. Recall that the cross-cycle edges are (v_i^1, v_{i+1}^1) , (v_i^2, v_{i+1}^3) , (v_i^3, v_{i+1}^5) , (v_i^4, v_{i+1}^2) , (v_i^5, v_{i+1}^4) , where the pattern is that as the superscript of the first vertex increases by 1, the superscript of the second vertex increases by 2 (mod 5). Specifically, P has vertices $v_i^k, v_{i+1}^j, v_{i+1}^{j+1 \pmod{5}}$, and $v_{i+1}^{j+2 \pmod{5}}$, while P' has vertices $v_i^k, v_{i+1}^{k+1 \pmod{5}}, v_{i+1}^{j+2 \pmod{5}}$.

Since P is the unique shortest path between its endpoints in H , we know that $w_H(P) < w_H(P')$. Taking the sum of this inequality for all 5 paths in \mathcal{P} that go from C_i to C_{i+1} , we get $w_H(C_i) + X > 2w_H(C_{i+1}) + X$, where X is the sum of the weights of the cross-cycle edges from C_i to C_{i+1} . The X cancels, so we get $w_H(C_i) > 2w_H(C_{i+1})$, as desired. \blacktriangleleft

3.3 Approximate Version

We now show that exponential aspect ratio is required even if we allow stretch. But unlike in the directed case, we do not have a lower bound against arbitrary stretch, only against a small constant stretch.

► **Theorem 8** (Exponential lower bound for approximation in undirected graphs). *For any $\varepsilon \leq 13/12$, there are n -node undirected graphs G such that any $(1 + \varepsilon)$ -stretch shortest-paths preserving graph H has aspect ratio $2^{\Omega(n)}$.*

High-Level Description

For the approximate version, the third issue for undirected graphs in the high-level description above becomes more troublesome: namely, that alternate paths can go through multiple levels of cycles and then return back to the original level. For the exact version for undirected graphs, we handled this by setting cross-cycle edges to have very high weight relative to the cycle edges; in particular, we set all of their weights to 1. This does not work for the approximate version for the following reason.

Just as in the directed case (Lemma 13), we want to ensure that each path in \mathcal{P} is the *only* α -approximate shortest path between its endpoints in G . This is no longer true for undirected graphs if we set the cross-cycle edges to have very large weight. In particular, recall that each path $P \in \mathcal{P}$ consist of one cross-cycle edge and two cycle edges from C_{i+1} , while there also exists a corresponding not-shortest path P' between the same endpoints that consist of one cross-cycle edge and one cycle edge from C_i . The two paths both use exactly one cross-cycle edge, so if that weight is very large then it will dominate the edge-weights on the cycle, and P' will be an α -approximation to P .

To address this issue, we need to set the weight of the cross-cycle edges carefully; setting these weights either too large or too small does not work. This balancing act is the reason that our lower-bound construction only works for a small constant α .

Construction

The construction for our lower-bound graph $G = (V, E, w)$ is identical to the exact version for undirected graphs, except that each cross-cycle edge e from C_i to C_{i+1} is given weight $w(e) = 1/3^{i-1}$.

Analysis

We will use the same set \mathcal{P} of paths as the exact undirected version (see Section 3.2). We will show in Lemma 16 that each path in \mathcal{P} is the *only* $13/12$ -approximate shortest path between its endpoints in G . Since $\alpha \leq 13/12$, this means that for any α -approximate shortest-path-preserving graph $H = (V, E, w_H)$ of G , each path in \mathcal{P} is the unique shortest path between its endpoints in H . Combined with Lemma 15, this implies that H has aspect ratio $2^{\Omega(n)}$, as desired. Thus, it remains only to prove Lemma 16:

► **Lemma 16.** *Each path in \mathcal{P} is the only $13/12$ -approximate shortest path between its endpoints in G .*

Proof. Fix a path $P \in \mathcal{P}$. P is from v_i^k to $v_{i+1}^{j+2 \pmod{5}}$ (where j is such that (v_i^k, v_{i+1}^j) is a cross-cycle edge). We calculate $w(P)$ as follows: the first term is for the cross-cycle edge, while the second term is the sum of the two edge weights in C_{i+1} :

$$w(P) = 1/3^{i-1} + 2/3^{i+1}.$$

Let P' be an alternate path with the same endpoints as P . We will argue that $w(P') > 13/12 \cdot w(P)$.

Any path from C_i to C_{i+1} must use a cross-cycle edge between C_i and C_{i+1} , which has weight $1/3^{i-1}$. If P' uses another cross-cycle edge, its next cross-cycle edge is from C_{i+1} to either C_{i+2} to C_i , so has weight at least $1/3^i$. This means that we would have:

$$\begin{aligned} w(P') &\geq 1/3^{i-1} + 1/3^i \\ &> 13/12 \cdot (1/3^{i-1} + 2/3^{i+1}) \\ &= 13/12 \cdot w(P). \end{aligned}$$

Thus, suppose P' uses exactly one cross-cycle edge.

Furthermore, each edge in C_i is of weight at least $1/3^i$, so due to exactly the same string of inequalities, if P' used any edge in C_i then we would again have $w(P') > 13/12 \cdot w(P)$. Thus, suppose P' does not use any edges from C_i .

12:16 Are There Graphs Whose Shortest Path Structure Requires Large Edge Weights?

The only edge from v_i^k that is not in C_i is the first edge of P : (v_i^k, v_{i+1}^j) , so P' must begin with this edge. From v_{i+1}^j , recall that P' cannot use another cross-cycle edge so the remainder of P' must be within C_{i+1} . Because C_{i+1} is a cycle, there are exactly two simple paths from v_{i+1}^j to $v_{i+1}^{j+2 \pmod{5}}$: P takes the one on 2 edges and the other one goes the other way around the 5-cycle and has 3 edges. These edges have weight $1/3^{i+1}$, so the total weight of 3 of these edges is $1/3^i$. Thus, if P' takes the 3-edge path, exactly the same string of inequalities as above implies $w(P') > 13/12 \cdot w(P)$. ◀

4 Upper Bound for DAGs

In this section, we show that in contrast to general directed graphs, any DAG G admits a shortest-path-preserving graph with aspect ratio $O(n)$.

► **Theorem 3** (Linear upper bound for DAGs). *Every n -node DAG has a shortest-paths preserving graph of aspect ratio $O(n)$.*

Since G is a DAG, we can label the vertices v_1, \dots, v_n according to their topological order; that is, for every edge $(v_i, v_j) \in E$ we have $j > i$.

Let W be the heaviest edge weight in the input graph $G = (V, E, w)$. We now define a new weight function w_H : given any edge (v_i, v_j) in G , set

$$w_H(v_i, v_j) = w(v_i, v_j) + W \cdot (j - i)$$

We now argue that $H = (V, E, w_H)$ is a shortest-path-preserver of G .

► **Lemma 17.** *Every shortest path in H is a shortest path in G and vice versa.*

Proof. This follows directly from the well-known fact that price functions do not change shortest paths; note that $w_H(v_i, v_j)$ are precisely the reduced weights obtained from price function $\phi(v_i) = W \cdot i$. See e.g. [20] for more details on price functions.

For the sake of completeness, we also prove the Lemma from scratch in Section A of the appendix. ◀

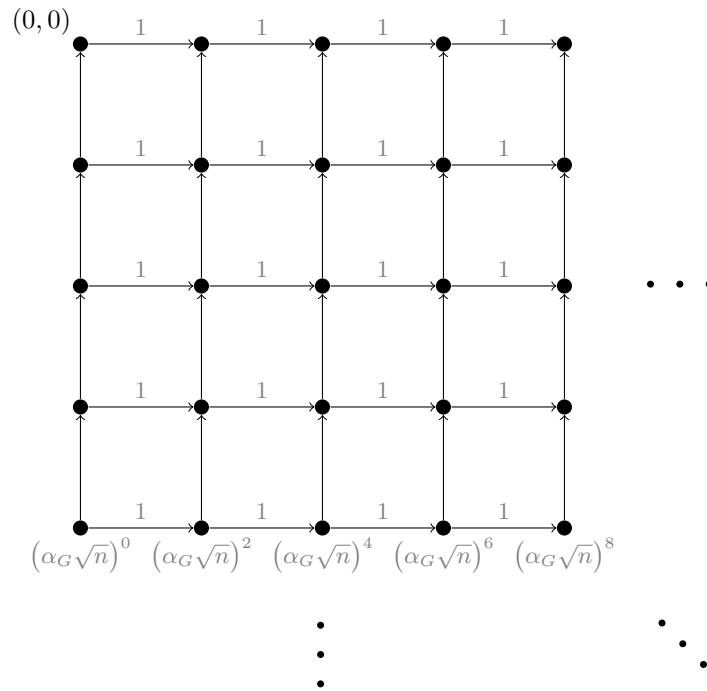
► **Lemma 18.** *The weight function w_H has aspect ratio at most $n + 1$.*

Proof. Recall that we defined W to be the heaviest edge weight in G . For any edge (v_i, v_j) we have $1 \leq j - i < n$. We can thus conclude that the maximum edge weight in H is at most $W + nW = (n + 1)W$, while the minimum edge weight in H is at least W . This implies that w_H has aspect ratio at most $n + 1$. ◀

5 Lower Bound for DAGs

In this section we will prove an exponential lower bound for two-sided approximation on DAGs:

► **Theorem 10** (Exponential lower bound for two-sided approximation in DAGs). *For any $\alpha_H, \alpha_G > 1$, there is an n -node DAG G such that any $(\alpha_H \rightarrow \alpha_G)$ -stretch shortest-paths preserving graph H has aspect ratio $(\alpha_H)^{\Omega(\sqrt{n})}$.*



■ **Figure 7** The graph G . The weights under each column indicate that every edge in that column has that weight.

5.1 Construction

The lower-bound graph $G = (V, E, w)$ is the $\sqrt{n} \times \sqrt{n}$ grid graph where all horizontal edges are directed to the right and all vertical edges are directed upwards (see Figure 7). We will refer to the vertices by their (row, column) coordinates, where the vertex in the top left corner is $(0,0)$.

The edge weights are defined as follows. All horizontal edges have weight 1. The weight of each vertical edge depends on its column. All vertical edges in column j have weight $(\alpha_G \cdot \sqrt{n})^{2j}$.

This completes the construction of G . Note that G is a DAG.

5.2 Analysis

The analysis is broken into two parts. First, in Lemmas 19 and 20 we will show that each path from a particular collection of paths is the *only* α_G -approximate shortest path between its endpoints. This means that in any $(\alpha_H \rightarrow \alpha_G)$ -stretch shortest-paths preserving graph $H = (V, E, w_H)$, each of these paths must be the *only* α_H -approximate shortest path between its endpoints. Then, in Lemma 21 we will prove by induction on the size of the grid, that the sum of edge weights in the last row and the last column of H must be exponentially large.

► **Lemma 19.** *Let $s = (i, j)$ be a vertex in G , and let $t = (i - 1, k)$ where $k > j$ (that is, t is exactly one row above and at least one column to the right of s). Then, the shortest st -path P uses one vertical edge followed by a series of horizontal edges. Furthermore, P is the only α_G -approximate st -shortest path.*

12:18 Are There Graphs Whose Shortest Path Structure Requires Large Edge Weights?

Proof. Since edges are only directed up and to the right, and t is exactly one row above s , the only st -paths use a (possibly empty) series of horizontal edges, followed by a single vertical edge, followed by a (possibly empty) series of horizontal edges. Since all horizontal edges have weight 1, the contribution of these edges to $w(P)$ is \sqrt{n} . Let ℓ be the column of the single vertical edge e in P (so $j \leq \ell \leq k$). Then, $w(e) = (\alpha_G \cdot \sqrt{n})^{2\ell}$. This quantity is minimized when ℓ is minimized, so $\ell = j$. This means that the shortest st -path P has its vertical edge in column j (the column containing s). This proves the first part of the claim.

To prove that P is the only α_G -approximate st -shortest path, we consider any other st -path P' . Let $\ell' > j$ be the column of the single vertical edge in P' . Then, $w(P') = \sqrt{n} + (\alpha_G \cdot \sqrt{n})^{2\ell'} \geq (\alpha_G \cdot \sqrt{n})^{2(j+1)}$. Then,

$$\begin{aligned} \frac{w(P')}{w(P)} &\geq \frac{(\alpha_G \cdot \sqrt{n})^{2(j+1)}}{\sqrt{n} + (\alpha_G \cdot \sqrt{n})^{2j}} \\ &> \frac{(\alpha_G \cdot \sqrt{n})^{2(j+1)}}{(\alpha_G \cdot \sqrt{n})^{2j+1}} \\ &\geq \alpha_G. \end{aligned} \quad \blacktriangleleft$$

We also make a symmetric claim:

► **Lemma 20.** *Let $s = (i, j)$ be a vertex in G , and let $t = (k, j + 1)$ where $k < i$ (that is, t is at least one row above and exactly one column to the right of s). Then, the shortest st -path P uses a series of vertical edges followed by exactly one horizontal edge. Furthermore, P is the only α_G -approximate st -shortest path.*

Proof. Since edges are only directed up and to the right, and t is exactly one column to the right of s , the only st -paths use a (possibly empty) series of vertical edges, followed by a single horizontal edge, followed by a (possibly empty) series of vertical edges. The contribution of the single horizontal edge to $w(P)$ is 1. The contribution of the vertical edges is as follows. Let b be the number of vertical edges on P before the horizontal edge. Then the sum of the weights of the vertical edges is $b(\alpha_G \cdot \sqrt{n})^{2j} + (k - i - b)(\alpha_G \cdot \sqrt{n})^{2(j+1)}$. This quantity is minimized when b is maximized, so $b = k - i$. This proves the first part of the claim.

To prove that P is the only α_G -approximate st -shortest path, we consider any other st -path P' . Let $b' < k - i$ be the number of vertical edges on P' before the horizontal edge. Then,

$$\begin{aligned} w(P') &= 1 + b'(\alpha_G \cdot \sqrt{n})^{2j} + (k - i - b')(\alpha_G \cdot \sqrt{n})^{2(j+1)} \\ &\geq (k - i - 1)(\alpha_G \cdot \sqrt{n})^{2j} + (\alpha_G \cdot \sqrt{n})^{2(j+1)} \quad (\text{setting } b' = k - i - 1) \\ &\geq (\alpha_G \cdot \sqrt{n})^{2(j+1)}. \end{aligned}$$

We know that $w(P) = 1 + (k - i)(\alpha_G \cdot \sqrt{n})^{2j}$. Thus,

$$\begin{aligned} \frac{w(P')}{w(P)} &\geq \frac{(\alpha_G \cdot \sqrt{n})^{2(j+1)}}{1 + (k - i)(\alpha_G \cdot \sqrt{n})^{2j}} \\ &\geq \frac{(\alpha_G \cdot \sqrt{n})^{2(j+1)}}{1 + \sqrt{n}(\alpha_G \cdot \sqrt{n})^{2j}} \\ &> \frac{(\alpha_G \cdot \sqrt{n})^{2(j+1)}}{(\alpha_G \cdot \sqrt{n})^{2j+1}} \\ &\geq \alpha_G. \end{aligned} \quad \blacktriangleleft$$

Now we will prove that in H , the sum of last column and the last row of the grid must have exponentially large weight. We can assume without loss of generality that the minimum edge weight in H is 1.

► **Lemma 21.** *Suppose that the minimum edge weight in H is 1. Then, the sum of edge weights (in H) of the last row and last column is at least $(\alpha_H)^{\sqrt{n}-1}$.*

Proof. The proof is by induction on the dimension of the grid. We will show that for any L , when H is an $L \times L$ grid, the sum of weights in the last row and the last column is at least $(\alpha_H)^{L-1}$.

Base Case. In the base case H is a 2×2 grid. Let P' be the path $(1, 0) \rightarrow (1, 1) \rightarrow (0, 1)$; that is $w_H(P')$ is the sum of weights in the last row and column. Our goal is to show that $w_H(P') \geq \alpha_H$. Let P be the shortest path from $(1, 0)$ to $(0, 1)$. By Lemma 19, P uses one vertical edge followed by one horizontal edge; that is, $P = (1, 0) \rightarrow (0, 0) \rightarrow (0, 1)$. By Lemma 19, we also know that P' , which has the same endpoints as P , is not an α_G -approximate shortest path in G , and thus cannot be an α_H approximate shortest path in H . Since the minimum edge weight in H is 1, $w_H(P) \geq 2$, so $w_H(P') \geq 2 \cdot \alpha_H$, as desired.

Inductive Hypothesis. Suppose that when H is an $L \times L$ grid, the sum of weights in the last row and the last column is at least $(\alpha_H)^{L-1}$.

Inductive Step. We will show that when H is an $(L+1) \times (L+1)$ grid, the sum of edge weights in the last row and the last column is at least $(\alpha_H)^L$.

Let $H_L \subseteq H$ be the $L \times L$ grid that excludes the last row and last column of H . We will apply the inductive hypothesis on H_L .

We define the paths X , X' , Y , and Y' as shown in Figure 8. X is the last row of H_L . X' is X shifted down by one row. Y is the last column of H_L . Y' is Y shifted to the right by one column. That is,

- X is the horizontal path $(L-1, 0) \rightarrow (L-1, L-1)$,
- X' is the horizontal path $(L, 0) \rightarrow (L, L-1)$,
- Y is the vertical path $(L-1, L-1) \rightarrow (0, L-1)$, and
- Y' is the vertical path $(L-1, L) \rightarrow (0, L)$.

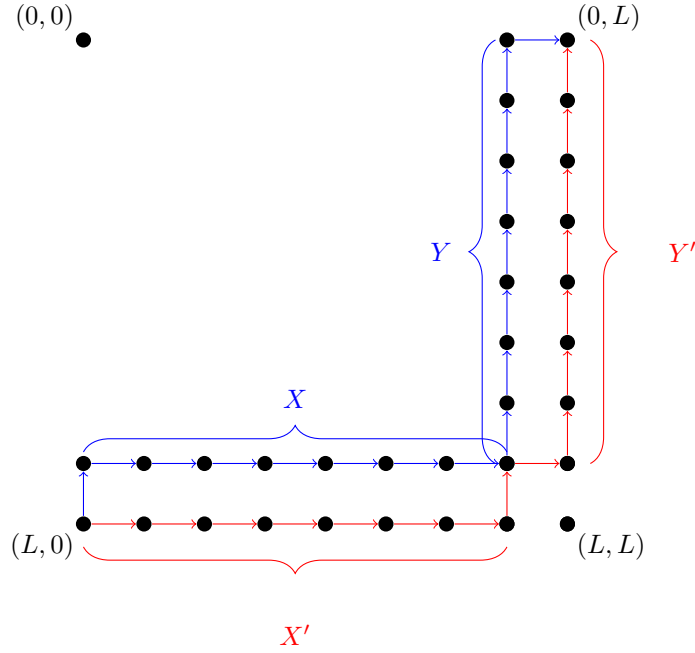
By the inductive hypothesis we know that $w_H(X) + w_H(Y) > (\alpha_H)^{L-1}$.

Let P_X be the shortest path in H from $(L, 0)$ to $(L-1, L-1)$ (see Figure 8). By Lemma 19, P_X takes one vertical edge followed by all of the edges in X . Consider an alternate path $P_{X'}$ with the same endpoints, which takes all of the edges in X' followed by the vertical edge $(L, L-1) \rightarrow (L-1, L-1)$. By Lemma 19, $P_{X'}$ is not an α_G -approximate shortest path in G , and thus cannot be an α_H approximate shortest path in H . Thus,

$$w_H(X') + w_H((L, L-1), (L-1, L-1)) > \alpha_H \cdot w_H(X). \quad (1)$$

Now, we will derive a symmetric inequality for Y instead of X . Let P_Y be the shortest path in H from $(L-1, L-1)$ to $(0, L)$. By Lemma 20, P_Y takes all of the edges in Y followed by one horizontal edge. Consider an alternate path $P_{Y'}$ with the same endpoints, which takes the horizontal edge $(L-1, L-1) \rightarrow (L-1, L)$ followed by all of the edges in Y' . By Lemma 20, $P_{Y'}$ is not an α_G -approximate shortest path in G , and thus cannot be an α_H approximate shortest path in H . Thus,

$$w_H(Y') + w_H((L-1, L-1), (L-1, L)) > \alpha_H \cdot w_H(Y). \quad (2)$$



■ **Figure 8** The paths X , X' , Y , and Y' . The shortest path P_X is the blue path from $(L,0)$ to $(L-1,L-1)$, while the not shortest path $P_{X'}$ is the red path with the same endpoints. The shortest path P_Y is the blue path from $(L-1,L-1)$ to $(0,L)$, while the not-shortest path $P_{Y'}$ is the red path with the same endpoints. The remaining grid edges are omitted from the figure for simplicity.

Taking the sum of Equations (1) and (2), we have

$$w_H(X') + w_H(Y') + w_H((L,L-1), (L-1,L-1)) + w_H((L-1,L-1), (L-1,L)) > \alpha_H \cdot (w_H(X) + w_H(Y)) \tag{3}$$

By Lemma 19, we know that $(L,L-1) \rightarrow (L-1,L-1) \rightarrow (L-1,L)$ is a shortest path, while $(L,L-1) \rightarrow (L,L) \rightarrow (L-1,L)$ is not. Thus, the left hand side of Equation (3) is strictly less than $w_H(X') + w_H(Y') + w_H((L,L-1), (L,L)) + w_H((L,L), (L-1,L))$, which is precisely the sum of the last row and last column of H . Thus, our goal is to show that the right hand side of Equation (3) is at least $(\alpha_H)^L$.

By the inductive hypothesis, $w_H(X) + w_H(Y) > (\alpha_H)^{L-1}$, so we have

$$\alpha_H \cdot (w_H(X) + w_H(Y)) > \alpha_H \cdot (\alpha_H)^{L-1} = (\alpha_H)^L.$$

This completes the proof. ◀

We have just proven in Lemma 21 that in H the sum of edge weights in the last row and last column is at least $(\alpha_H)^{\sqrt{n}-1}$. This means that at least one of these $2\sqrt{n}$ edges has weight at least $(\alpha_H)^{\sqrt{n}-1} / (2\sqrt{n}) = (\alpha_H)^{\Omega(\sqrt{n})}$. This completes the proof of Theorem 10.

References

- 1 Amir Abboud and Greg Bodwin. Reachability preservers: New extremal bounds and approximation algorithms. In *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1865–1883. Society for Industrial and Applied Mathematics, 2018.
- 2 Shyan Akmal and Nicole Wein. A local-to-global theorem for congested shortest paths. *31st Annual European Symposium on Algorithms (ESA 2023)*, 2023.

- 3 Saeed Akhoondian Amiri and Julian Wargalla. Disjoint shortest paths with congestion on dags. *arXiv preprint*, 2020. [arXiv:2008.08368](https://arxiv.org/abs/2008.08368).
- 4 Lorenzo Balzotti. Non-crossing shortest paths are covered with exactly four forests. *arXiv preprint*, 2022. [arXiv:2210.13036](https://arxiv.org/abs/2210.13036).
- 5 András A Benczúr and David R Karger. Approximating st minimum cuts in $\tilde{O}(n^2)$ time. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 47–55, 1996.
- 6 Aaron Bernstein, Maximilian Probst Gutenberg, and Christian Wulff-Nilsen. Near-optimal decremental SSSP in dense weighted digraphs. In *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020, Durham, NC, USA, November 16-19, 2020*, pages 1112–1122, 2020.
- 7 Aaron Bernstein and Nicole Wein. Closing the gap between directed hopsets and shortcut sets. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 163–182. SIAM, 2023.
- 8 Greg Bodwin. On the structure of unique shortest paths in graphs. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2071–2089. SIAM, 2019.
- 9 Nairen Cao, Jeremy T. Fineman, and Katina Russell. Brief announcement: An improved distributed approximate single source shortest paths algorithm. In *PODC '21: ACM Symposium on Principles of Distributed Computing, Virtual Event, Italy, July 26-30, 2021*, pages 493–496, 2021.
- 10 Ruoxu Cen, Ran Duan, and Yong Gu. Roundtrip Spanners with $(2k-1)$ Stretch. In *47th International Colloquium on Automata, Languages, and Programming (ICALP 2020)*, pages 24:1–24:11, 2020. [doi:10.4230/LIPIcs.ICALP.2020.24](https://doi.org/10.4230/LIPIcs.ICALP.2020.24).
- 11 Parinya Chalermsook, Syamantak Das, Yunbum Kook, Bundit Laekhanukit, Yang P Liu, Richard Peng, Mark Sellke, and Daniel Vaz. Vertex sparsification for edge connectivity. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1206–1225. SIAM, 2021.
- 12 Julia Chuzhoy and Ruimin Zhang. A new deterministic algorithm for fully dynamic all-pairs shortest paths. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing, STOC 2023, Orlando, FL, USA, June 20-23, 2023*, pages 1159–1172, 2023.
- 13 Daniel Cizma and Nati Linial. Geodesic geometry on graphs. *Discrete & Computational Geometry*, 68(1):298–347, 2022.
- 14 Daniel Cizma and Nati Linial. Irreducible nonmetrizable path systems in graphs. *Journal of Graph Theory*, 102(1):5–14, 2023.
- 15 Don Coppersmith and Michael Elkin. Sparse sourcewise and pairwise distance preservers. *SIAM Journal on Discrete Mathematics*, 20(2):463–501, 2006.
- 16 Sabine Cornelsen, Maximilian Pfister, Henry Förster, Martin Gronemann, Michael Hoffmann, Stephen Kobourov, and Thomas Schneck. Drawing shortest paths in geodetic graphs. In *International Symposium on Graph Drawing and Network Visualization*, pages 333–340. Springer, 2020.
- 17 Sebastian Forster and Danupon Nanongkai. A faster distributed single-source shortest paths algorithm. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, pages 686–697, 2018.
- 18 Torben Hagerup, Jyrki Katajainen, Naomi Nishimura, and Prabhakar Ragde. Characterizing multiterminal flow networks and computing flows in networks of small treewidth. *Journal of Computer and System Sciences*, 57(3):366–375, 1998.
- 19 Monika Henzinger, Sebastian Krinninger, and Danupon Nanongkai. Decremental single-source shortest paths on undirected graphs in near-linear total update time. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 146–155, 2014.

- 20 Donald B. Johnson. Efficient algorithms for shortest paths in sparse networks. *J. ACM*, 24(1):1–13, 1977.
- 21 Ravi Kannan, Santosh Vempala, and Adrian Vetta. On clusterings: Good, bad and spectral. *Journal of the ACM (JACM)*, 51(3):497–515, 2004.
- 22 Shimon Kogan and Merav Parter. New diameter-reducing shortcuts and directed hopsets: Breaking the barrier. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1326–1341. SIAM, 2022.
- 23 Robert Krauthgamer, Huy L Nguyễn, and Tamar Zondiner. Preserving terminal distances using minors. *SIAM Journal on Discrete Mathematics*, 28(1):127–141, 2014.
- 24 David Peleg and Jeffrey Ullman. An optimal synchronizer for the hypercube. *SIAM Journal on Computing (SICOMP)*, 18(4):740–747, 1989.
- 25 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.
- 26 Iam Roditty, Mikkel Thorup, and Uri Zwick. Roundtrip spanners and roundtrip routing in directed graphs. *ACM Transactions on Algorithms (TALG)*, 4(3):1–17, 2008.
- 27 Václav Rozhon, Bernhard Haeupler, Anders Martinsson, Christoph Grunau, and Goran Zuzic. Parallel breadth-first search and exact shortest paths and stronger notions for approximate distances. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing, STOC 2023, Orlando, FL, USA, June 20-23, 2023*, pages 321–334, 2023.
- 28 Avi Shoshan and Uri Zwick. All pairs shortest paths in undirected graphs with integer weights. In *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)*, pages 605–614. IEEE, 1999.
- 29 Daniel A Spielman and Shang-Hua Teng. Spectral sparsification of graphs. *SIAM Journal on Computing*, 40(4):981–1025, 2011.
- 30 Mikkel Thorup. On shortcutting digraphs. In *International Workshop on Graph-Theoretic Concepts in Computer Science*, pages 205–211. Springer, 1992.
- 31 Jeffrey D Ullman and Mihalis Yannakakis. High-probability parallel transitive-closure algorithms. *SIAM Journal on Computing*, 20(1):100–125, 1991.
- 32 Chun Jiang Zhu and Kam-Yiu Lam. Deterministic improved round-trip spanners. *Information Processing Letters*, 129:57–60, 2018.
- 33 Uri Zwick. All pairs shortest paths using bridging sets and rectangular matrix multiplication. *Journal of the ACM (JACM)*, 49(3):289–317, 2002.

A Proof of Lemma 17

Proof. Consider any paths P, P' between the same set of endpoints. We will show that $w(P) - w(P') = w_H(P) - w_H(P')$, which clearly implies the lemma.

Let $P = v_{i_1}, v_{i_2}, \dots, v_{i_k}$. We have

$$w(P) = \sum_{j=1}^{k-1} w(v_{i_j}, v_{i_{j+1}})$$

We also have

$$w_H(P) = \sum_{j=1}^{k-1} [w(v_{i_j}, v_{i_{j+1}}) + W(i_{j+1} - i_j)] = \sum_{j=1}^{k-1} w(v_{i_j}, v_{i_{j+1}}) + W(i_k - i_1),$$

where the second inequality holds because the sum $\sum_{j=1}^{k-1} W(i_{j+1} - i_j)$ telescopes.

Combining the inequalities above yields $w_H(P) = w(P) + W(i_k - i_1)$. By an identical argument we $w_H(P') = w(P') + W(i_k - i_1)$; here we use the fact that P and P' have the same endpoints, namely v_{i_1} and v_{i_k} . Subtracting these two equalities, we conclude that

$$w(P) - w_H(P) = w(P') - w_H(P') \rightarrow w(P) - w(P') = w_H(P) - w_H(P'). \quad \blacktriangleleft$$