Oriented Spanners: Constructions and Structural Properties

Dissertation

zur Erlangung des Grades eines

D o k t o r s d e r N a t u r w i s s e n s c h a f t e n

der Technischen Universität Dortmund
an der Fakultät für Informatik

von
Antonia Kalb

Dortmund
2026

Tag der mündlichen Prüfung:	31.07.2026
Dekan:	Prof. Dr. Jens Teubner
Gutachter:	Prof. Dr. Kevin Buchin
	Prof. Dr. Wolfgang Mulzer

Abstract

In this PhD thesis, we study *oriented geometric spanners*, a directed variant of the classical spanner problem that models one-way networks.

Given a point set P in a metric space and a real number $t \geq 1$, an *oriented t-spanner* is an oriented graph $\overrightarrow{G} = (P, \overrightarrow{E})$, where for each pair $p, q \in P$, the shortest closed walk in $\overrightarrow{G}$ that contains p and q is at most a factor t longer than the perimeter of the smallest triangle in P containing p and q. The *oriented dilation* of $\overrightarrow{G}$ is the minimum t for which $\overrightarrow{G}$ is an oriented t-spanner.

This thesis encompasses all currently known results on oriented spanners. In particular, we address the following problems:

Oriented dilation. We prove All-Pairs-Shortest-Paths-hardness for the problem of computing the oriented dilation of a given metric graph. We provide algorithms to compute the oriented dilation in cubic time and to approximate it in subcubic time. We bound the worst-case dilation t of a minimum dilation spanner by $1.5 \leq t \leq 5/3$ for metric and $2\sqrt{3} - 2 \leq t \leq 5/3$ for Euclidean point sets.

Graph orientation. We prove NP-hardness for deciding whether a given undirected graph can be oriented such that the dilation of the orientation is below a given threshold. We present a greedy algorithm that orients a given set of triangles. The resulting oriented graph, whose edge set consists of these triangles, contains for each input triangle Δ a closed walk of length at most twice the perimeter of Δ. This algorithm is, for example, used to orient a triangulation or a complete graph. Special attention is given to the orientation of complete graphs, i.e., tournaments. Our best algorithm computes a tournament with dilation $5/3$ in $\mathcal{O}(n^3)$ time for n points in a metric space.

Spanner construction. Given a point set, we aim to construct oriented spanners that satisfy certain sparseness criteria. For points in $\mathbb{R}$, a sparse 1-spanner is straightforward. As a one-dimensional equivalent of plane spanners on points in convex position, we study graphs with a one-page book embedding. We greedily construct a one-page plane 5-spanner in $\mathcal{O}(n \log n)$ time and a minimum one-page plane spanner in $\mathcal{O}(n^7)$ time. We show that a $(1+\varepsilon')$-approximation of the minimum oriented triangulation on a given point set implies a $(1 + \varepsilon)$-approximation of the minimum dilation triangulation. Further, given the parameters t and m, it is NP-hard to compute a minimum oriented t-spanner with at most m edges. We present

an orientation of the greedy triangulation that yields an oriented $\mathcal{O}(n)$-spanner for arbitrary and an $\mathcal{O}(1)$-spanner for convex point sets. For higher-dimensional spaces, we present an algorithm that extracts a linear-sized subgraph of a given tournament with dilation t by increasing the dilation only by a small $\varepsilon > 0$. Combining this with our results on tournaments, we construct a $(5/3 + \varepsilon)$-spanner with $\mathcal{O}(n)$ edges in $\mathcal{O}(n^3)$ time and $\mathcal{O}(n^2)$ space for n points in a metric space that admits a well-separated pair decomposition. For points in a Euclidean space of constant dimension, we present a construction of a $(2+\varepsilon)$-spanner with $\mathcal{O}(n)$ edges in $\mathcal{O}(n \log n)$ time and $\mathcal{O}(n)$ space. Further, we consider the average oriented dilation, defined as the sum over the oriented dilation of all pairs of points divided by the number of pairs. We provide an algorithm computing an oriented spanner with $\mathcal{O}(n)$ edges and average dilation $1 + \varepsilon$ for sufficiently large point sets.

Relation to other models. We compare oriented dilation with other models. Specifically, we prove that an oriented spanner is related to an (undirected) 1-*edge fault-tolerant spanner*, i.e., a graph G that guarantees under the failure of an arbitrary edge pq that for each pair of points a shortest path in $G \setminus pq$ is at most a factor t longer than the shortest path in the complete graph without the edge pq. We prove that the greedy triangulation is a 1-edge fault-tolerant $\mathcal{O}(n)$-spanner for arbitrary and an $\mathcal{O}(1)$-spanner for convex point sets.

Acknowledgments

This thesis would not have been possible without the support, guidance, and encouragement of many people who accompanied me along the way. I am deeply grateful to everyone who contributed to the completion of this work.

First and foremost, my sincere thanks go to my supervisor, Kevin Buchin, for insightful discussions, your vision, and for the many opportunities you have given me. I feel truly privileged that you were so generous with your time and expertise.

To my colleagues at TU Dortmund University – Carolin Rehs, for her feedback and guidance; Mart Hagedoorn, for camaraderie and friendship throughout this journey; and Torben Scheele, for co-suffering in managing tutorials for 700+ students.

To the team at the University of Sydney, especially Joachim Gudmundsson and André van Renssen, for hosting me in 2022. This visit sparked my passion for international exchange. I am delighted to be returning as postdoc soon.

To the team at UPC Barcelona, especially Rodrigo Silveira, for hosting me in 2023 and including me in a running project. I am grateful for the opportunity to work with Sergio Cabello during this visit.

To the team at the Computational Geometry Lab in Ottawa, especially Michiel Smid, for hosting me in 2024 and 2025 and for supporting my academic and personal growth. Your interest in oriented spanners shaped the core of my PhD project.

To the teams at BARC and ITU in Copenhagen, especially Sampson Wong, for hosting me in 2025. I appreciated the energetic atmosphere that combined dedication to research with socializing. Thank you, Sampson, for your advice and steady support throughout my PhD that went well beyond what I could have hoped for.

To the team at IIT Bombay, especially Sujoy Bhore, for hosting me in 2025. Thank you for allocating productive research time despite an already packed calendar.

To my co-authors, Sujoy Bhore, Ahmad Biniaz, Prosenjit Bose, Kevin Buchin, Sergio Cabello, Jean-Lou De Carufel, Jacobus Conradi, Delia Garijo, Joachim Gudmundsson, Sariel Har-Peled, Fabian Klute, Abhiruk Lahiri, Guangping Li, Anil Maheshwari, Saeed Odak, Irene Parada, Lukas Plätz, Aleksandr Popov, Carolin Rehs, André van Renssen, André Schulz, Rodrigo Silveira, Michiel Smid and Sampson Wong, for fruitful collaboration and all I learned from your diverse perspectives.

To my family and friends, for your continuous support and encouragement.

To my parents, for laying the foundation for all of this.

Finally, to Christian, for being a constant source of guidance and strength.

Contents

Chapter 1
Introduction

Geometric spanners are a fundamental concept in computational geometry and graph theory, bridging geometric structures with their graph representations. Geometric graphs naturally model numerous real-world infrastructures such as transportation systems, power grids and communication networks. In these applications, ensuring fast connectivity between all points is essential. While the complete graph provides optimal connectivity, its quadratic size is often impractical. Consequently, geometric spanners are sparse subgraphs that approximate pairwise distances.

Formally, given a point set P and a parameter t, a *directed t-spanner* $G = (P, E)$ contains for each pair $p, q \in P$ a path from p to q whose length is at most a factor t longer than their shortest path in the complete bi-directed graph on P. The smallest such factor t is called the *dilation* of G.

Since their introduction in 1989 [78], the research landscape on geometric spanners has expanded from simple sparseness criteria to complex multi-objective constraints, including planarity [26], resilience to node or connection failure [22, 28, 69], embedding in terrains [2, 12], or directionality [5].

Although geometric spanners have been studied for decades (see [74] for a survey), directed versions have only been considered more recently. This is surprising given their broad range of applications from wireless ad-hoc networks [29, 82] to robot motion planning [41] and road networks [9, 46]. Beyond mere directionality, limitations in space or resources, as well as safety requirements often necessitate strictly unidirectional edges. Examples include transportation infrastructures with one-way streets or rail tracks, air traffic with predefined unidirectional airways, and communication systems enforcing asymmetric data transmission to reduce interference or enhance security. In industrial contexts, pipelines and conveyor belts represent physical restrictions to one-way flow, while robot motion planning benefits from directed paths that minimize congestion and simplify collision avoidance.

This motivates our study of *oriented graphs* as spanners, i.e., directed spanners $\overrightarrow{G} = (P, \overrightarrow{E})$ where $(p, q) \in \overrightarrow{E}$ implies $(q, p) \notin \overrightarrow{E}$. Under this restriction, if an edge (p, q) is added, the dilation in the opposite direction is never 1. Even worse, given a set of three points, where p and q are very close to each other and the third point is far away from both, any oriented graph has arbitrarily high dilation for either (p, q) or (q, p) (see Figure 1.1). Thus, considering the dilation of an oriented graph $\overrightarrow{G}$ as the maximum ratio of the shortest path between any pair of points in $\overrightarrow{G}$ to their direct distance would not tell us much about the quality of the spanner. To obtain meaningful results, we introduce the *oriented dilation*, defined as the maximum ratio over all pairs $p, q \in P$ of the shortest closed walk in $\overrightarrow{G}$ through p and q and their shortest cycle in the complete undirected graph on P.

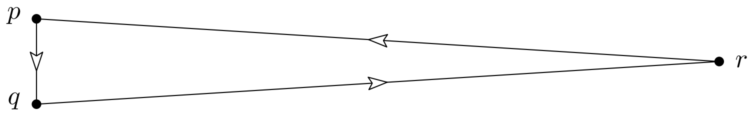


Figure 1.1: If p and q are very close to each other and r is far away from both, any oriented graph has arbitrarily high directed dilation.

This new measure is similar to the notion of dilation in roundtrip spanners [36, 37] that have been considered in the setting of (non-geometric) directed graph spanners; but roundtrip spanners require a starting graph, and using the complete bi-directed geometric graph would not give meaningful results.

By extending geometric spanners to oriented graphs, we uncover a rich landscape of research questions. During four years of doctoral research, we investigated how orientations affect classic spanner constructions and designed new algorithms tailored for this oriented setting. This dissertation presents our results.

1.1 Contributions

We study several fundamental problems within the setting of oriented spanners, addressing questions about their construction and structural properties. Specifically, we consider three main problems:

- **Oriented dilation:** How can we compute or approximate the oriented dilation of a given oriented graph? What are lower and upper bounds on the worst-case dilation?
- **Graph orientation:** How can we orient a given undirected graph to optimize its oriented dilation?
- **Spanner construction:** How can we construct oriented spanners on a given point set that satisfy additional properties like sparseness or planarity?

For each of these problems, we analyze their computational complexity and provide algorithmic solutions.

1.1.1 Computing the Oriented Dilation (Chapter 3)

The hardness of computing the dilation of a given (un)directed graph is a longstanding open question, and only cubic time algorithms are known [55]. It is conjectured that computing the (un)directed dilation is as hard as the all-pairs shortest paths problem (APSP). Using Dijkstra's algorithm, it can be computed in $\mathcal{O}(nm + n^2 \log n)$ time, where n is the number of points and m the number of edges. Fast approximations are known in general and for specific graph classes [74].

We prove APSP-hardness for computing the dilation of a given oriented graph in a metric space. Since the APSP-problem is believed to need truly cubic time [79], it is unlikely that the oriented dilation can be computed in subcubic time. Computing it naively is possible in $\mathcal{O}(n^3)$ time for n points. Note that for graphs with $m = \mathcal{O}(n)$ edges, this is slower than computing the dilation of an undirected graph. We complement this with a subcubic-time approximation. For this, we present a data structure that, given two points $p, q \in P$, returns a point $r \in P \setminus \{p, q\}$ such that the triangle Δ_{pqr} approximates the minimum perimeter triangle of p and q in P.

1.1.2 Orienting Undirected Graphs (Chapter 4)

A natural approach to construct oriented spanners is to first compute a suitable undirected graph and then orient it. However, it is unlikely that this leads to efficient algorithms, as we prove NP-hardness for computing the minimum dilation orientation of a given undirected graph.

We present an algorithm that orients a set of triangles such that the resulting oriented graph contains for each input triangle a closed walk whose length is at most twice its perimeter. Among other applications, we use this algorithm for orienting a triangulation or a complete graph.

Bounding the worst-case dilation provides fundamental insights into how well distances can be preserved under certain constraints. While in the undirected setting the complete graph trivially is a 1-spanner, this does not hold for oriented spanners. We present example point sets to derive lower bounds and analyze *tournaments*, i.e., orientations of complete graphs, to obtain upper bounds. Our results show that the worst-case dilation t of a minimum dilation spanner satisfies $1.5 \leq t \leq 5/3$ for metric and $2\sqrt{3} - 2 \leq t \leq 5/3$ for Euclidean point sets.

1.1.3 Constructing Oriented Spanners (Chapters 5 to 7)

Our general goal is to obtain spanners with a linear number of edges, which can be achieved by bounding the number of edges explicitly or by restricting to a class of sparse graphs like plane graphs [26]. As a first step towards plane spanners, we study spanners on one-dimensional point sets. Then, we study oriented triangulations. Finally, we study sparse spanners in higher-dimensional spaces.

Table 1.1 offers an overview of all spanner constructions presented in this thesis, including their dilation guarantees and construction times. We say that a spanner is a *minimum (oriented) spanner* if it minimizes the dilation under given restrictions.

One-dimensional point sets (Chapter 5). While obtaining a sparse 1-spanner on a one-dimensional point set is straightforward, already for five points such a spanner has no plane embedding with the leftmost and rightmost point on the outer face. Therefore, we restrict our attention to a graph class that is closer to the plane graph for two-dimensional point sets: one-page book embeddings.

As a *one-page plane spanner* is outerplanar, this particular class of graphs is also motivated by the problem of finding a minimum plane spanner for points in convex position (see Chapter 6). We present a greedy algorithm that returns an oriented 5-spanner with a one-page book embedding in $\mathcal{O}(n \log n)$ time for n points in $\mathbb{R}$.

Further, we provide a dynamic program, that returns a minimum one-page plane spanner in $\mathcal{O}(n^7)$ time. As a natural extension, we present an oriented 2-spanner with a two-page book embedding.

Plane spanners (Chapter 6). The problem of computing the minimum undirected spanner restricted to the class of plane straight-line graphs is known as the *minimum dilation triangulation* problem; its hardness is still open [46, 51], but it is conjectured to be NP-hard [27]. By reducing from this problem, we prove a computational lower bound for computing a *minimum oriented triangulation*. In particular, we show that a $(1+\varepsilon')$-approximation of the minimum oriented triangulation on n' points in $T(n', \varepsilon')$ time implies a $(1+\varepsilon)$-approximation of the minimum dilation triangulation on n points in $T(4n^2, \varepsilon - \delta)$ time for any $0 < \varepsilon < 1$ and any $0 < \delta < \varepsilon^2$. Therefore, it is unlikely that minimum plane oriented spanners can be constructed in polynomial time.

While the computational complexity remains unresolved, much work has focused on bounding the dilation of triangulations. Xia [88] bounded the dilation of the Delaunay triangulation by 1.998. To date, this is the best known upper bound on the minimum dilation in general. For convex point sets, an upper bound of 1.88 is presented in [15]. Bounding the worst-case dilation of the minimum dilation triangulation is an open problem [46]. (See [26] for a survey on plane spanners.)

Das and Joseph [39] showed that all triangulations that fulfill the α-diamond property are undirected $\mathcal{O}(1)$-spanners. One example of such a triangulation is the greedy triangulation, for which we present an orientation that yields an oriented $\mathcal{O}(n)$-spanner. This is tight, since there is a point set for which every orientation of its greedy triangulation has dilation $\Omega(n)$. For points in convex position, we prove that the greedy triangulation results in a plane oriented t-spanner with $t < 27.14 \cdot t_g$, where t_g is an upper bound on the undirected dilation of the greedy triangulation. This raises the question whether all triangulations fulfilling the α-diamond property also provide oriented spanners. We answer this question negatively.

To ensure resilience against edge or vertex failures in geometric networks, prior work has studied *fault-tolerant spanners* [1, 17, 69, 77]. Given the parameters t and k, an undirected graph $G = (P, E)$ is called a *k-edge fault-tolerant t-spanner*, if for every edge set $E' \subset E$ with $|E'| = k$, the shortest path in $G \setminus E'$ between any pair of points is at most a factor t longer than their shortest path in the complete graph on P without the edges in E'.

We identify a close connection between oriented dilation and the dilation of a spanner under the failure of $k = 1$ edges. Building on this, we present the first results on *plane fault-tolerant spanners*: For n points in the Euclidean plane, the greedy triangulation is a 1-edge fault-tolerant $\mathcal{O}(n)$-spanner for arbitrary point sets and an $\mathcal{O}(1)$-spanner for convex point sets. Moreover, the same point sets that yield lower bounds on the oriented dilation of different α-diamond property triangulations also imply corresponding lower bounds on fault-tolerance.

	Point set	Spanner type	Dilation	Time complexity	Reference
Chapter 4	metric space	tournament	minimum	unknown	
			≤ 2	$\mathcal{O}(n^3)$	Corollary 4.3
			$\leq 5/3$	$\mathcal{O}(n^3)$	Theorem 4.5
	$\mathcal{O}(1)$-dim. Eucl.		$\leq 2+\varepsilon$	$\mathcal{O}(n^2)$	Corollary 4.4
Chapter 5	1-dim. Euclidean	sparse	1	$\mathcal{O}(n)$	Corollary 5.2
		1-page-plane	minimum	$\mathcal{O}(n^7)$	Theorem 5.10
		1-page-plane	≤ 5	$\mathcal{O}(n \log n)$	Theorem 5.9
		2-page-plane	≤ 2	$\mathcal{O}(n)$	Theorem 5.12
Chapter 6	2-dim. Euclidean	plane	minimum	Min. Dil. Triang.-hard	Theorem 6.1
			$\mathcal{O}(n)$	$\mathcal{O}(n \log n)$	Theorem 6.3
	2-dim. Eucl. convex		$\mathcal{O}(1)$	$\mathcal{O}(n \log n)$	Theorem 6.5
Chapter 7	$\mathcal{O}(1)$-dim. Euclidean	sparse	minimum	NP-hard	Theorem 7.1
			$\leq 5/3+\varepsilon$	$\mathcal{O}(n^3)$	Corollary 7.3
			$\leq 2+\varepsilon$	$\mathcal{O}(n \log n)$	Theorem 7.4
			avg $\leq 1+\varepsilon+\mathcal{O}(1/n)$	$\mathcal{O}(n \log n)$	Theorem 7.6

Table 1.1: Overview of spanner constructions on point sets of size n.

Sparse spanners (Chapter 7). It is known that computing a minimum undirected spanner with at most $n - 1$ edges, i.e., a *minimum dilation tree*, is NP-hard [34]. More general, given a point set and a parameter $m \geq n - 1$, computing a minimum undirected spanner with at most m edges is NP-hard [51, 58].

The corresponding question for oriented spanners asks for the *minimum dilation cycle*, for which we prove NP-hardness. In particular, it is NP-hard to compute the minimum oriented spanner with at most m edges.

For undirected spanners of linear size and small dilation, the greedy spanner was presented in 1990 [6], giving a t-spanner of size $\mathcal{O}(n)$ for any constant t. The well-separated pair decomposition (WSPD), introduced by Callahan and Kosaraju in 1995 [30], can be used to obtain a $(1 + \varepsilon)$-spanner with $\mathcal{O}(n)$ edges in near-linear construction time. (For a survey on sparse constant dilation spanners see [74].)

We cannot hope to obtain oriented spanners with dilation close to 1: There are point sets in a Euclidean space where no oriented t-spanner exists for $t < 2\sqrt{3} - 2 \approx 1.464$. We present an algorithm that, given a t-tournament and an additional property about short cycles, computes a sparse $(t+\varepsilon)$-spanner in $\mathcal{O}(n^3)$ time. Our algorithm applies to point sets in any metric space that admits a linear-size WSPD, such as the Euclidean space [30], or metric spaces with constant doubling dimension [84]. Plugging in our results for tournaments, an oriented $(5/3 + \varepsilon)$-spanner with $\mathcal{O}(n)$ edges can be computed in $\mathcal{O}(n^3)$ time and $\mathcal{O}(n^2)$ space. Increasing the dilation but significantly improving upon runtime and space requirements, we construct an oriented $(2+\varepsilon)$-spanner for Euclidean point sets with $\mathcal{O}(n)$ edges in $\mathcal{O}(n \log n)$ time and $\mathcal{O}(n)$ space.

Since the maximum oriented dilation is not robust to outliers and in the worst case is bounded below by 1.46, we also consider *average oriented dilation*, i.e., the sum of the oriented dilation of all pairs of points divided by the number of pairs. Average dilation has been considered before in the context of undirected spanners: An undirected spanner with average dilation $1 + \mathcal{O}((\log n/n)^{1/(2d+1)})$ and linear edge number can be computed in $\mathcal{O}(n \log n)$ time [44]. We provide an algorithm computing a sparse oriented spanner with average dilation $1 + \varepsilon$ for point sets in a Euclidean space of constant dimension d: More concretely, our algorithm gives an oriented spanner with average dilation at most $1+\mathcal{O}(1/s + s^d/n)$ with $\mathcal{O}(s^d n)$ edges in $\mathcal{O}(s^d n \log n)$ time and $\mathcal{O}(s^d n)$ space, where s is any sufficiently large number that may depend on n.

1.2 Projects Featured in this Thesis

This thesis is a self-contained presentation of all our papers on oriented spanners existing to date. The results have been rearranged and grouped according to their thematic context rather than by their original publications, allowing a clearer presentation of related findings. In this section, each paper is briefly introduced individually, listed in the chronological order in which these results were obtained.

"Oriented Spanners" [AK4]

This paper introduces the model of oriented spanners. We show that computing a minimum oriented dilation spanner with a given number of edges is NP-hard. Further, we prove that computing the minimum oriented dilation triangulation on a given point set is at least as hard as computing the minimum dilation triangulation. We prove that for convex point sets the greedy triangulation results in a plane oriented $\mathcal{O}(1)$-spanner. For n points in $\mathbb{R}$, we present a dynamic program to compute a graph with a one-page book embedding and minimum oriented dilation that runs in $\mathcal{O}(n^7)$ time, and a greedy algorithm that returns a 5-spanner in $\mathcal{O}(n \log n)$ time. For two-page book embeddings, we present a sparse 2-spanner.

This paper is published in ESA'23 and in Algorithmica. Parts of this paper were presented at EuroCG'23. This research was done in collaboration with Kevin Buchin, Joachim Gudmundsson, Aleksandr Popov, Carolin Rehs, André van Renssen and Sampson Wong.

"Oriented Dilation of Undirected Graphs" [AK7]

We show that given an undirected graph, it is NP-hard to decide whether the edges can be oriented in a way that the oriented dilation of the resulting graph is below a given threshold. Further, for four points in a metric space, we present a tournament with dilation 1.5. Proving tightness, we present a set of four points, where every tournament has dilation 1.5. This improves the known lower bound for the worst-case minimum dilation.

This paper was presented at EuroCG'24. The NP-hardness proof is published in [AK6]. This research was done in collaboration with Kevin Buchin, Carolin Rehs and André Schulz.

"Experimental Analysis of Oriented Spanners on One-dimensional Point Sets" [AK5]

In this paper, we study oriented spanners on one-dimensional point sets that have a one- and two-page book embedding. We model the problem of computing such minimum dilation spanners as satisfiability problem. We utilize this formulation for computational experiments on such spanners, resulting in several interesting insights. We present a lower bound on the worst-case minimum dilation of $1+\Phi$ and $\sqrt{2}$ for oriented spanners with a one- and two-page book embedding, respectively, where Φ is the golden ratio.

This paper is published in CCCG'24. This research was done in collaboration with Kevin Buchin, Guangping Li and Carolin Rehs. As the author did not implement the experiments, this thesis includes only the lower bounds and other structural insights presented in this paper.

"Computing Oriented Spanners and their Dilation" [AK6]

In this paper, we construct a sparse oriented $(2+\varepsilon)$-spanner in $\mathcal{O}(n \log n)$ time for n points in a Euclidean space of constant dimension. Our construction uses the well-separated pair decomposition and an algorithm that efficiently computes a $(1+\varepsilon)$-approximation of the minimum perimeter triangle in the point set containing two given query points. We show that, in general, computing the orientation of an undirected graph that minimizes its oriented dilation is NP-hard, even for point sets in the Euclidean plane. We further prove that even if the oriented graph is already given, computing its oriented dilation is APSP-hard for metric graphs. We complement this result with an approximation of the oriented dilation of a given graph in subcubic time for points in a Euclidean space of constant dimension.

This paper is published in SoCG'25 and is under revision for DCG. Parts of this paper were presented at EuroCG'25. This research was done in collaboration with Kevin Buchin, Anil Maheshwari, Saeed Odak, Carolin Rehs, Michiel Smid and Sampson Wong. This work was supported by a fellowship of the German Academic Exchange Service (DAAD).

"Sparse Oriented Spanners in Metric Spaces" [AK1]

We present an algorithm that computes a tournament with dilation 5/3 on any metric point set. Moreover, for point sets of size n in metric space that admits a linear-size WSPD, we show that given a tournament with dilation t and an additional property about short cycles, a sparse $(t + \varepsilon)$-spanner can be computed in $\mathcal{O}(n^3)$ time. Together with our results on tournaments, we obtain a $(5/3 + \varepsilon)$-spanner of size $\mathcal{O}(n)$ in $\mathcal{O}(n^3)$ time and $\mathcal{O}(n^2)$ space. Finally, for Euclidean spaces, we present a construction algorithm for oriented spanners with average dilation $1+\varepsilon$.

This paper is published in ESA'26. This research was done in collaboration with Sujoy Bhore, Ahmad Biniaz, Kevin Buchin, Jean-Lou De Carufel, Anil Maheshwari, Saeed Odak, Carolin Rehs and Michiel Smid. This work was supported by Tu.hosts, a competitive funding program for PhD students at TU Dortmund University.

"Fault-Tolerance and Oriented Dilation of the Greedy Triangulation" [AK2]

We show that the greedy triangulation is a 1-edge fault-tolerant $\mathcal{O}(n)$-spanner for any point set of size n and an $\mathcal{O}(1)$-spanner for convex point sets. This is tight, due to the existence of a point set whose greedy triangulation has dilation $\Omega(n)$ under the failure of a single edge. For further prominent triangulations, we provide lower bounds on their fault-tolerance. Additionally, we present an orientation of the greedy triangulation that yields a plane oriented $\mathcal{O}(n)$-spanner.

This paper was presented at EuroCG'26. This research was done in collaboration with Prosenjit Bose and Kevin Buchin.

1.3 Other Publications during PhD

This section describes briefly two other publications the author contributed to, which will not be discussed further in this thesis.

"Algorithms for Distance Problems in Continuous Graphs" [AK8]

We studied the problem of computing the diameter and the mean distance of a continuous graph, i.e., a connected graph where all points along the edges, instead of only the vertices, must be taken into account.

In this paper, we use geometric techniques to obtain subquadratic time algorithms to compute the diameter and the mean distance of a continuous graph for two well-established classes of sparse graphs. We show that the diameter and the mean distance of a continuous graph of constant treewidth k can be computed in $\mathcal{O}(n \log^{O(k)} n)$ time, where n is the number of vertices. We also show that computing the diameter and mean distance of a continuous planar graph with n vertices and F faces takes $\mathcal{O}(nF \log n)$ time.

This paper is published in WADS'25 and submitted to TALG. This research was done in collaboration with Sergio Cabello, Delia Garijo, Fabian Klute, Irene Parada and Rodrigo Silveira.

"On Small Pair Decompositions for Point Sets" [AK3]

We studied the problem of computing a well-separated pair decomposition (WSPD) with minimum size, i.e., the minimum number of pairs.

In this paper, we show constant approximation algorithms in low-dimensional Euclidean space and doubling metrics. This problem is computationally hard even in $\mathbb{R}^2$ and hard to approximate in general metric spaces. We also introduce a new pair decomposition, removing the requirement that the diameters of the paired sets should be small. Surprisingly, we show that in a general metric space, one can compute such a decomposition of size $\mathcal{O}(\frac{n}{\varepsilon} \log n)$, which is dramatically smaller than the quadratic bound for WSPDs. In $\mathbb{R}^d$, the bound improves to $\mathcal{O}(d\frac{n}{\varepsilon} \log \frac{1}{\varepsilon})$.

This paper is published in ESA'26 and presented in part at EuroCG'26. This research was done in collaboration with Kevin Buchin, Jacobus Conradi, Sariel Har-Peled, Abhiruk Lahiri, Lukas Plätz, Carolin Rehs and Sampson Wong.

Chapter 2
Preliminaries

Let $(P, |\cdot|)$ be a finite metric space, where the distance between any two points p and q in P is denoted by $|pq|$. By $\mathbb{R}^d$, we denote a Euclidean space of constant dimension d.

For two distinct points p and q, we use pq for an undirected edge adjacent to p and q. By (p, q), we denote a directed edge from p to q. For a graph $G = (P, E)$ and an edge set $E' \subseteq E$, let $G \setminus E'$ be the graph $G' = (P, E \setminus E')$.

A *walk* is defined as a sequence of points and edges of a graph. The *length of a walk* Π is the sum of the lengths of its edges and denoted by $|\Pi|$. A walk is called *closed* if it starts and ends at the same point. A *cycle* is a walk where all edges and points except the start and end point are distinct. A *path* is a walk where all points and edges are distinct.

For two distinct points p and q in P and a graph $G = (P, E)$, we denote by $d_G(p, q)$ the *shortest path* from p to q in G and by $C_G(p, q)$ the *shortest closed walk* containing p and q in G. If such a walk does not exist, then $|C_G(p, q)| = \infty$. Analogously, $|d_G(p, q)| = \infty$ implies that G is not connected.

The perimeter of the triangle spanned by the points p, q, r is written as $|\Delta_{pqr}|$. For two distinct points $p, q \in P$, we use $\Delta^*(p, q)$ to denote a shortest cycle through p and q in the complete undirected graph on P. For point sets in a metric space, this is a *minimum perimeter triangle* in P containing p and q, i.e., the triangle Δ_{pqr} where $r = \arg\min_{r \in P \setminus \{p,q\}} |pr| + |qr|$. For an edge pq of a triangulation T, let $\Delta_T(pq)$ be a *minimum perimeter face of T adjacent to pq*.

An *oriented graph* $\overrightarrow{G} = (P, \overrightarrow{E})$ is a directed graph such that for any two distinct points p and q in P, at most one of the two directed edges (p, q) and (q, p) is in $\overrightarrow{E}$. In other words, if (p, q) is in $\overrightarrow{E}$, then (q, p) is not in $\overrightarrow{E}$.

Definition (oriented dilation)**.** *Let $(P, |\cdot|)$ be a finite metric space and let $\overrightarrow{G}$ be an oriented graph with vertex set P. For two distinct points $p, q \in P$, their* oriented dilation *is defined as*

$$\mathrm{odil}_{\overrightarrow{G}}(p, q) = \frac{|C_{\overrightarrow{G}}(p, q)|}{|\Delta^*(p, q)|}.$$

The oriented dilation of $\overrightarrow{G}$ is defined as $\mathrm{odil}(\overrightarrow{G}) = \max_{p,q \in P} \mathrm{odil}(p, q)$. *We call $\overrightarrow{G}$ an* oriented t-spanner *if* $\mathrm{odil}(\overrightarrow{G}) \leq t$.

Note that, undirecting an oriented spanner with constant oriented dilation does not yield an undirected spanner. Consider an oriented graph on four points, where the edges form a cycle visiting all points (see Figure 2.1). The oriented dilation of this spanner is 2. Placing two opposite points arbitrarily close to each other, the dilation of those two points in the underlying undirected graph is arbitrarily high.

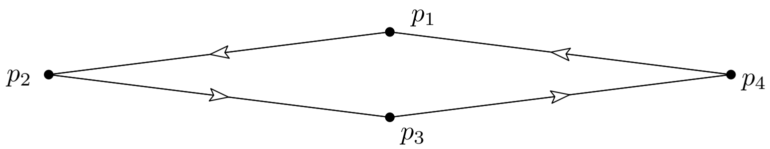


Figure 2.1: An oriented 2-spanner. If p_1 and p_3 are arbitrarily close to each other and p_2 and p_4 equally far away, the underlying undirected graph has arbitrarily high undirected dilation.

Since the dilation of a graph is not robust to outliers, considering the average dilation of a graph may provide more meaningful information on the network.

Definition (oriented average dilation)**.** *Let $(P, |\cdot|)$ be a finite metric space and let $\overrightarrow{G}$ be an oriented graph with vertex set P. The oriented average dilation of $\overrightarrow{G}$ is defined as*

$$\frac{1}{\binom{n}{2}} \sum_{p,q \in P} \mathrm{odil}_{\overrightarrow{G}}(p, q).$$

We call $\overrightarrow{G} = (P, \overrightarrow{E})$ an *orientation* of an undirected graph $G = (P, E)$, if for each undirected edge $pq \in E$ it holds either $(p, q) \in \overrightarrow{E}$ or $(q, p) \in \overrightarrow{E}$.

We often orient triangles *consistently*, meaning that their oriented edges form a cycle. If one edge of a triangle is already oriented, there is exactly one way to complete the consistent orientation. If two edges pq and pr of a triangle Δ_{pqr} are already oriented, consistency requires that they form either the path $p \to q \to r$ or its reverse.

By K_n, we denote the complete undirected graph on a point set of size n. An orientation $\overrightarrow{K}$ of the complete undirected graph $K(P)$ on P is called a *tournament* on P. A tournament with oriented dilation at most t is called a *t-tournament*.

2.1 The Well-Separated Pair Decomposition

Callahan and Kosaraju [30] introduced the *well-separated pair decomposition* (WSPD) as a tool to solve various distance problems on multidimensional point sets. Intuitively, it breaks all $\binom{n}{2}$ pairs of points down to $\mathcal{O}(n)$ subsets that approximate the complete graph.

For a real number $s > 0$, two point sets A and B form an *s-well-separated pair*, if A and B can each be covered by balls of the same radius, say ρ, such that the distance between the balls is at least $s \cdot \rho$. An *s-well-separated pair decomposition* of size m for a point set P is a sequence of s-well-separated pairs $\{A_i, B_i\}$, such that for all $1 \leq i \leq m$ it holds that $A_i \cap B_i = \emptyset$ and for any two points $p, q \in P$, there is a unique index i such that $p \in A_i$ and $q \in B_i$, or $p \in B_i$ and $q \in A_i$.

The following properties are used repeatedly in this paper.

WSPD-Property 1 (Smid [83])**.** *Let $s > 0$ be a real number and let $\{A, B\}$ be an s-well-separated pair. For points $a, a' \in A$ and $b, b' \in B$, it holds that*

(i) $|aa'| \leq 2/s \cdot |ab|$, *and*

(ii) $|a'b'| \leq (1 + 4/s) \cdot |ab|$.

Callahan and Kosaraju [30] showed a fast WSPD-construction based on a split tree.

WSPD-Property 2 (Callahan and Kosaraju [30])**.** *Let P be a set of n points in $\mathbb{R}^d$, where d is a constant, and let $s > 0$ be a real number. An s-WSPD for P consisting of $\mathcal{O}(s^d n)$ pairs can be constructed in $\mathcal{O}(n \log n + s^d n)$ time.*

In Section 7.4, we process the well-separated pairs in a certain order: We sort the pairs $\{A_i, B_i\}$ ascending by the distances of two representatives $a \in A_i$ and $b \in B_i$. Such an ordering is common in inductive proofs for WSPD based algorithms (see [83]). These proofs make use of of the subsequent property, which follows directly from WSPD-Property 1.

WSPD-Property 3. *Let P be a point set in $\mathbb{R}^d$, where d is a constant, and let $s > 4$ be a real number. Let $\{A_i, B_i\}$ for $1 \leq i \leq m$ denote the pairs of an s-WSPD for P, sorted by ascending distance between arbitrary points $a_i \in A_i$ and $b_i \in B_i$. For p, q two distinct points in P, let $\{A_j, B_j\}$ and $\{A_k, B_k\}$ be well-separated pairs such that $p \in A_j$ and $q \in B_j$ and $p, q \in A_k$. By construction, it holds $j < k$.*
(Note that index j is unique, while multiple indices may satisfy the condition for k.)

Those results are for point sets in a Euclidean space $\mathbb{R}^d$, where d is a constant. The WSPD has been generalized to more general metric spaces, such as the unit-disk graph metric [50], or metric spaces with constant doubling dimension [60, 84].

2.2 Approximate Nearest Neighbor Queries

For approximate nearest neighbor queries, we want to preprocess the points in a metric space $(P, |\cdot|)$, such that for any query point q, we can report quickly its *approximate nearest neighbor* in P. That is, if q^* is closest point to q in P, then the query algorithm returns a point q' in P such that $|qq'| \leq (1 + \varepsilon) \cdot |qq^*|$. Such a data structure is offered in [10] for Euclidean point sets:

Lemma 2.1 (Arya, Mount, Netanyahu, Silverman and Wu [10])**.** *Let P be a set of n points in $\mathbb{R}^d$, where d is a constant. In $\mathcal{O}(dn \log n)$ time, a data structure of size $\mathcal{O}(dn)$ can be constructed that supports the following operations:*

- *Given a real number $\varepsilon > 0$ and a query point q in $\mathbb{R}^d$, return a $(1 + \varepsilon)$-approximate nearest neighbor of q in P in $\mathcal{O}(d/\varepsilon^d \log n)$ time.*
- *Inserting a point into P and deleting a point from P in $\mathcal{O}(d \log n)$ time.*

Chapter 3
Computing the Oriented Dilation

Computing the dilation of a given graph is related to the all-pairs shortest paths problem (APSP). By using the well-known cubic-time algorithm for APSP by Floyd and Warshall [48, 86] (and computing the minimum perimeter triangles naively), the dilation of a given oriented graph with n points can be computed in $\mathcal{O}(n^3)$ time.

For points in $\mathbb{R}$, we prove in Chapter 5 that computing the oriented dilation on such graphs is possible in $\mathcal{O}(n^2 \log n)$ time in general (follows from Lemma 5.1) and in $\mathcal{O}(n)$ time for special graph classes on one-dimensional point sets (combine Lemmas 5.4 and 5.5).

In Section 3.1, we prove APSP-hardness for this computation problem on metric graphs. Since APSP is believed to run in truly cubic time [79], it seems unlikely that the oriented dilation can be computed in subcubic time. In Section 3.2, we present a $(1 + \varepsilon)$-approximation of the minimum perimeter triangle of two given query points in $\mathcal{O}(\log n)$ time. This query is used in Section 7.3 to efficiently compute a sparse $(2 + \varepsilon)$-spanner. In Section 3.3, we combine this approximation with the well-separated pair decomposition to approximate the oriented dilation of a given graph in subcubic time.

3.1 Hardness

Our hardness proof is a subcubic reduction from MinimumTriangle, which is the problem of finding a minimum weight cycle in an undirected weighted graph, where the minimum weight cycle is a triangle. For two computational problems A and B, a *subcubic reduction* is a reduction from A to B where any subcubic time algorithm for B implies a subcubic time algorithm for A (for a more thorough definition see [85]). We use the notation $A \leq_3 B$ to denote the existence of such a reduction from A to B.

Vassilevska Williams and Williams [85] proved APSP-hardness for detecting whether a weighted graph has a triangle of negative edge weight. Following their proof, we prove APSP-hardness of the more specific problem MinimumTriangle (for details see Appendix A).

Theorem 3.1 (APSP $\leq_3$ MinimumTriangle)**.** *Let G be an undirected graph with weights in $[7M, 9M]$ with $M \in \mathbb{Z}^+$ whose minimum weight cycle is a triangle. It is APSP-hard to find a minimum weight triangle in G.*

By reduction from MinimumTriangle, we prove APSP-hardness for ODIL, which is the problem of computing the oriented dilation of a given metric graph.

Theorem 3.2 (APSP $\leq_3$ Odil)**.** *Let $\overrightarrow{G}$ be an oriented graph in a metric space. It is APSP-hard to compute the oriented dilation of $\overrightarrow{G}$.*

Proof. Due to Theorem 3.1, it is sufficient to prove MinimumTriangle $\leq_3$ Odil.

Let $G = (P, E)$ be an undirected graph on n points with weights $w : E \to [7M, 9M]$ with $M \in \mathbb{Z}^+$ in which the minimum weight cycle C of G is a triangle. Note that the weights of G fulfill the triangle inequality.

We define a metric space $(P', |\cdot|)$ with $P' = P \cup \{u, v\}$ by the following: For each missing edge of a complete graph on P, we assign its weight to be $14M$. Therefore, we obtain a metric with distances in $[7M, 14M]$. Then, we add the points u and v with distance $15M$ to each point in $p \in P$. To preserve the metric, we increase the distance of every tuple by $15M$. So, the resulting metric space $(P', |\cdot|)$ has the following distances:

- $|pq| = w(p, q) + 15M$ for $pq \in E$,
- $|pq| = 14M + 15M = 29M$ for $pq \notin E$, and
- $|up| = |vp| = |vu| = 15M + 15M = 30M$ for $p \in P$.

This construction is visualized in Figure 3.1.

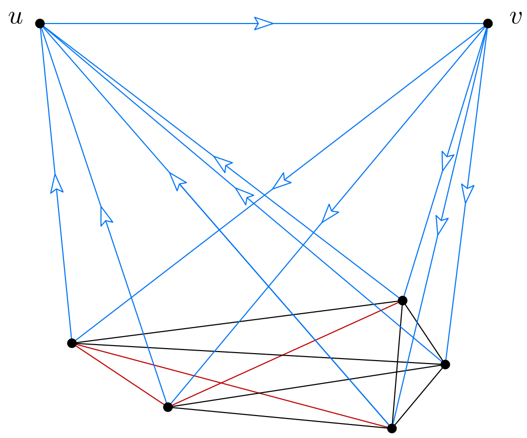


Figure 3.1: Graph construction in the proof of Theorem 3.2: Given an undirected graph on P (black), we add two points u, v with a large distance to all other points and augment it to a complete graph (red, dashed). The oriented graph $\overrightarrow{G}$ (blue) ensures that every shortest closed walk containing p and q in P visits u and v.

Let $\overrightarrow{G} = (P', \overrightarrow{E})$ be an oriented graph with $\overrightarrow{E} = \{(p, u), (v, p) \mid p \in P\} \cup \{(u, v)\}$.

We prove that the weight $w(C)$ of the minimum triangle C in G can be computed by computing $\mathrm{odil}(\overrightarrow{G})$. In particular, we show that $w(C) = \frac{180M}{\mathrm{odil}(\overrightarrow{G})} - 45M$.

We proceed by computing the oriented dilation of $\overrightarrow{G}$. We first note that all edges of $\overrightarrow{G}$ have weight $30M$.

The shortest closed walk of u or v and a point $p \in P$ is a triangle of length

$$|C_{\overrightarrow{G}}(u,p)| = |C_{\overrightarrow{G}}(v,p)| = |C_{\overrightarrow{G}}(u,v)| = 3 \cdot 30M = 90M.$$

Their minimum perimeter triangle contains two edges of length $30M$ (and one edge that is at least $22M$ long). Therefore, their dilation is

$$\operatorname{odil}(u,p) = \frac{|C_{\overrightarrow{G}}(u,p)|}{|\Delta^*(u,p)|} \leq \frac{90M}{2 \cdot 30M} = 1.5.$$

Analogously, it holds that $\operatorname{odil}(v,p) \leq 1.5$ for $p \in P$. For u and v, it holds that $\operatorname{odil}(u,v) = 1$, because every triangle in P' through u and v is a closed walk in $\overrightarrow{G}$.

For two distinct points $p, q \in P$, the shortest closed walk containing p and q visits u and v. It has length $|C_{\overrightarrow{G}}(p,q)| = 6 \cdot 30M = 180M$. For their minimum perimeter triangle holds $|\Delta^*(p,q)| \leq 3 \cdot 29M = 87M$. Since we have

$$\operatorname{odil}(p,q) = \frac{|C_{\overrightarrow{G}}(p,q)|}{|\Delta^*(p,q)|} \geq \frac{180M}{87M} > 2$$

for all $p, q \in P$, a worst-case pair of $\overrightarrow{G}$ must be a pair in P. Therefore, the oriented dilation of the constructed graph $\overrightarrow{G}$ is

$$\operatorname{odil}(\overrightarrow{G}) = \max_{p,q \in P} \frac{|C_{\overrightarrow{G}}(p,q)|}{|\Delta^*(p,q)|} = \frac{180M}{\min\limits_{p,q,r \in P} |pq| + |qr| + |pr|}.$$

Since C is a triangle in P and $w(C) \leq 27M$, it holds that

$$\min_{p,q,r \in P} |pq| + |qr| + |pr| = \min_{p,q,r \in P} w(p,q) + w(q,r) + w(p,r) + 3 \cdot 15M = w(C) + 45M.$$

Therefore, the oriented dilation of $\overrightarrow{G}$ is determined by the minimum weight cycle C. Consequently, given $\operatorname{odil}(\overrightarrow{G})$, the length of C is $w(C) = \frac{180M}{\operatorname{odil}(\overrightarrow{G})} - 45M$.

The most expensive operation in our reduction is defining the metric on $\mathcal{O}(n^2)$ pairwise distances. Following the definitions in [85], all instances have size $\mathcal{O}(n^2 \operatorname{polylog}(M))$ and all operations take subcubic time. Thus, computing the oriented dilation of an oriented metric graph is at least as hard as finding a minimum weight triangle in an undirected graph, which is APSP-hard. □

3.2 The Minimum Perimeter Triangle

Let P be a set of n points in $\mathbb{R}^d$, and let p and q be two distinct points in P. The smallest triangle $\Delta^*(p,q)$ through p and q can be computed naively in $\mathcal{O}(n)$ time.

The following approach answers such a query in $\mathcal{O}(n^{1-c})$ time, for some small constant $c > 0$ that depends on the dimension d. Note that computing $\Delta^*(p,q)$ is equivalent to computing the smallest real number $\rho > 0$, such that the ellipsoid

$$\sqrt{\sum_{i=1}^{d} (p_i - x_i)^2} + \sqrt{\sum_{i=1}^{d} (q_i - x_i)^2} \leq \rho$$

contains at least three points of P.

For a fixed real number ρ, we count the number of points of P in the above ellipsoid: Using *linearization*, see Agarwal and Matoušek [3], we convert P to a point set P' in a Euclidean space whose dimension depends on d. The problem then becomes that of counting the number of points of P' in a query simplex. Chan [32] has shown that such queries can be answered in $\mathcal{O}(n^{1-c})$ time, for some small constant $c > 0$. This result, combined with *parametric search*, see Megiddo [73], allows us to compute $\Delta^*(p, q)$ in $\mathcal{O}(n^{1-c})$ time.

For many applications, including our $(2+\varepsilon)$-spanner construction (see Section 7.3), an approximation of the minimum triangle is sufficient. The following lemma shows that a WSPD with respect to $s = 2/\varepsilon$ provides a $(1 + \varepsilon)$-approximation. So, if the WSPD is already computed, a suitable approximation is given without additional computational costs.

Lemma 3.3. *Let $s > 0$ be a real number and let $\{A, B\}$ be an s-well-separated pair in the point set P. Assume that $|B| \geq 2$. Let b, b' be two distinct points in B.*

1. *For any point $a \in A$, we have $|\Delta_{abb'}| \leq (1 + 2/s)|\Delta^*(a, b)|$.*
2. *Assume that $|A| \geq 2$. For any two distinct points $a, a' \in A$, it holds that $|\Delta^*(a, b)| \leq (1 + 4/s)|\Delta^*(a', b')|$.*

Proof. By the triangle inequality, we have

$$|\Delta^*(a, b)| \geq 2|ab|.$$

Again using the triangle inequality, we have

$$|\Delta_{abb'}| = |ab| + |bb'| + |ab'| \leq |ab| + |bb'| + |ab| + |bb'| = 2|ab| + 2|bb'|.$$

Using WSPD-Property 1, we conclude that

$$|\Delta_{abb'}| \leq (2 + 4/s)|ab| \leq (1 + 2/s)|\Delta^*(a, b)|.$$

Since the minimum perimeter triangle of any two points is always shorter than another triangle through those points, we apply WSPD-Property 1 to conclude that

$$\begin{aligned} |\Delta^*(a, b)| &\leq |\Delta_{abb'}| = |ab| + |bb'| + |ab'| \\ &\leq (1 + 4/s)|a'b'| + 2/s|a'b'| + (1 + 2/s)|a'b'| = (2 + 8/s)|a'b'| \\ &\leq (1 + 4/s)|\Delta^*(a', b')|. \end{aligned}$$ □

Note that Lemma 3.3 applies for every well-separated pair where at least one of the subsets contains at least two points. If both subsets have size one, we approximate the minimum perimeter triangle by the following query (see Algorithm 3.1):

1. We compute an approximate nearest neighbor p' of p. As we search for a point which is not q, we first delete and later reinsert q in the point set. The data structure presented in [10] supports insertion and deletion (see Lemma 2.1).

2. If p' is "far" from p, then $\Delta_{pqp'}$ is a suitable approximation of $\Delta^*(p,q)$.

3. Otherwise, we check a constant number of cells around p. For each cell, we find a suitable point x_c and return the smallest triangle Δ_{pqx_c} over all cells.

Algorithm 3.1 $(1+\varepsilon_1)$-Approximation of the Minimum Perimeter Triangle.

Require: Query points $p, q \in P$, positive reals $\varepsilon_1 < 2$, ε_2, ε_3, $\alpha > 2$, data structure M to approximate nearest neighbors
Ensure: Triangle Δ such that $|\Delta| \leq (1+\varepsilon_1)|\Delta^*(p,q)|$
  $p' = (1+\varepsilon_2)$-approximate nearest neighbor of p (with $p' \neq q$)
  if $|pp'| > \alpha|pq|$, **then return** $\Delta_{pqp'}$
  else
    Let B be the hypercube with sides of length $3\alpha|pq|$ that is centered at p and divided into cells with sides of length of $\varepsilon_3|pq|$
    for each cell in B **do**
      $x_c = (1+\varepsilon_2)$-approximate nearest neighbor of the cell center (which is neither p nor q)
    return the triangle Δ_{pqx_c} with the minimum perimeter across all cells

---

Lemma 3.4. *Let P be a set of n points in $\mathbb{R}^d$ and let $0 < \varepsilon_1 < 2$ be a real number. In $\mathcal{O}(dn\log n)$ time, the set P can be preprocessed into a data structure of size $\mathcal{O}(dn)$, such that, given any two distinct query points p and q in P, a $(1+\varepsilon_1)$-approximation of the minimum perimeter triangle $\Delta^*(p,q)$ can be computed in $\mathcal{O}(d/\varepsilon_1^d \log n)$ time.*

Proof. The data structure is the approximate nearest neighbor structure of Arya, Mount, Netanyahu, Silverman and Wu [10] (see Lemma 2.1). The query algorithm is presented in Algorithm 3.1. Let the constants in this algorithm be equal to $\varepsilon_2 = \varepsilon_1/2$, $\alpha = 4/\varepsilon_1$, and $\varepsilon_3 = \frac{2}{3\sqrt{d}} \cdot \varepsilon_1$.

Let p and q be two distinct points in P. We prove that Algorithm 3.1 returns a $(1+\varepsilon_1)$-approximation of $\Delta^*(p,q)$.

Let r be a point in $P \setminus \{p,q\}$ such that $|\Delta_{pqr}| = |\Delta^*(p,q)|$. Let p' be the $(1+\varepsilon_2)$-approximate nearest neighbor of p in $P \setminus \{p,q\}$ that is computed by the algorithm. By the triangle inequality, it holds

$$|\Delta_{pqp'}| = |pq| + |qp'| + |pp'| \leq 2|pq| + 2|pp'|. \qquad (\star)$$

Case 1: $|pp'| > \alpha|pq|$

Consider Figure 3.2. Let p^* be the exact nearest neighbor of p in the set $P \setminus \{q\}$. Since p' is a $(1+\varepsilon_2)$-approximate nearest neighbor of p in $P \setminus \{q\}$, and since $2|pr| \leq |\Delta^*(p,q)|$, we have

$$|pp'| \leq (1+\varepsilon_2)|pp^*| \leq (1+\varepsilon_2)|pr| \leq (1+\varepsilon_2)|\Delta^*(p,q)|/2.$$

Combining this inequality with Equation $(\star)$ and the assumption that $|pp'| > \alpha|pq|$, we have

$$|\Delta_{pqp'}| \leq (2 + 2/\alpha)|pp'| \leq (1 + 1/\alpha)(1+\varepsilon_2)|\Delta^*(p,q)|.$$

Since $\varepsilon_2 = \varepsilon_1/2$, $\alpha = 4/\varepsilon_1$ and $\varepsilon_1 < 2$, we have

$$\begin{aligned}|\Delta_{pqp'}| &\le (1+\varepsilon_1/4)(1+\varepsilon_1/2)|\Delta^*(p,q)| = (1+3\varepsilon_1/4+\varepsilon_1^2/8)|\Delta^*(p,q)| \\ &\le (1+\varepsilon_1)|\Delta^*(p,q)|.\end{aligned}$$

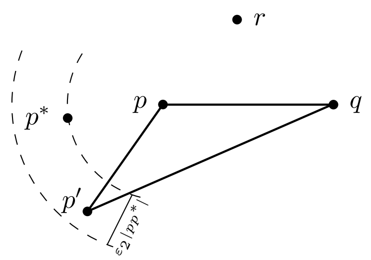


Figure 3.2: Visualization of Case 1 in the proof of Lemma 3.4: Let p' be a $(1+\varepsilon_2)$-approximated nearest neighbor of p. The triangle $\Delta_{pqp'}$ is a $(1+\varepsilon_1)$-approximation of $\Delta^*(p,q) = \Delta_{pqr}$.

Case 2: $|pp'| \le \alpha|pq|$

Consider the hypercube B with sides of length $3\alpha|pq|$ that is centered at p as shown in Figure 3.3.

We first prove that the third point r of $\Delta^*(p,q)$ lies in B. Using Equation ($\star$), the assumption that $|pp'| \le \alpha|pq|$, and the fact that $\alpha \ge 2$, we have

$$|\Delta^*(p,q)| \le |\Delta_{pqp'}| \le (2+2\alpha)|pq| \le 3\alpha|pq|.$$

Since $2|pr| \le |\Delta^*(p,q)|$, it follows that $|pr| \le 3\alpha/2|pq|$ and, therefore, $r \in B$.

Let C be the cell of B that contains r, let c be the center of C, let c^* be the exact nearest neighbor of c in $P \setminus \{p,q\}$, and let x_c be the $(1+\varepsilon_2)$-approximate nearest neighbor of c that is computed by the algorithm. We first prove an upper bound on the distance $|rx_c|$ of

$$|rx_c| \le |rc| + |cx_c| \le |rc| + (1+\varepsilon_2)|cc^*| \le |rc| + (1+\varepsilon_2)|rc| = (2+\varepsilon_2)|rc| \le 3|cr|.$$

Since r and c are in the same $\varepsilon_3|pq|$-sized cube and $\varepsilon_3 = \frac{2}{3\sqrt{d}}\varepsilon_1$, it follows that

$$|rx_c| \le 3 \cdot \sqrt{d}/2 \cdot \varepsilon_3|pq| = \varepsilon_1|pq| \le \varepsilon_1/2 \cdot |\Delta^*(p,q)|.$$

where the last inequality $2|pq| \le |\Delta^*(p,q)|$ follows from the triangle inequality.

Let $\Delta_{pqx_{c'}}$ be the triangle that is returned by the algorithm. It holds that

$$\begin{aligned}|\Delta_{pqx_{c'}}| &\le |\Delta_{pqx_c}| \le |pq| + |qr| + |rx_c| + |x_c r| + |rp| = |\Delta^*(p,q)| + 2|rx_c| \\ &\le (1+\varepsilon_1)|\Delta^*(p,q)|.\end{aligned}$$

The algorithm performs two deletions, two insertions, and $1 + (3\alpha/\varepsilon_3)^d = \mathcal{O}(1)$ approximate nearest neighbor queries. Each such operation takes $\mathcal{O}(d/\varepsilon_3^d \log n)$ time (see Lemma 2.1 [10]). Thus, the entire query takes $\mathcal{O}(d/\varepsilon_1^d \log n)$ time. □

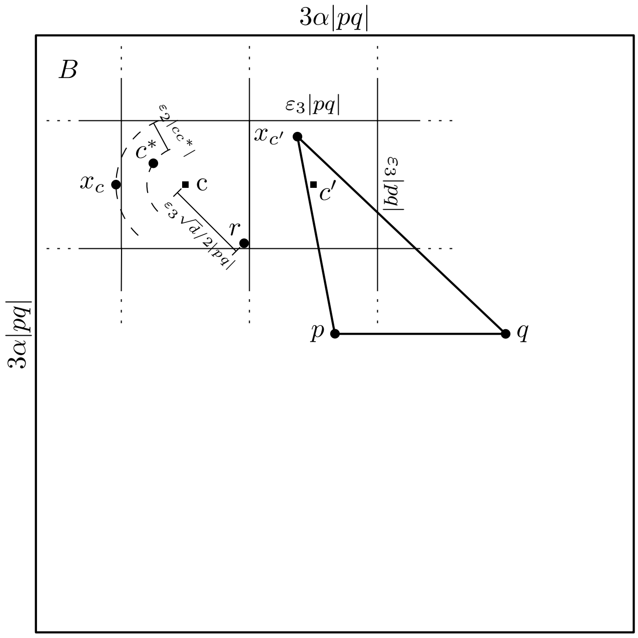


Figure 3.3: Visualization of Case 2 in the proof of Lemma 3.4: The third point r of $\Delta^*(p,q)$ lies in the cell centered at c. Since $\Delta_{pqx_{c'}}$ is returned, we have $|\Delta_{pqx_{c'}}| \leq |\Delta_{pqx_c}|$ where $x_{c'}$ and x_c are $(1+\varepsilon_2)$-approximated nearest neighbors of c' and c.

3.3 Approximating the Oriented Dilation

To compute the dilation, testing all $\binom{n}{2}$ point tuples seems to be unavoidable. However, to approximate the oriented dilation, we show that, as for undirected dilation (see [55, 74]), it is sufficient to compute the dilation of a suitable linear sized selection of pairs of points.

This selection is obtained by the WSPD. For each pair, our algorithm approximates their oriented dilation by approximating their minimum perimeter triangle and shortest closed walk. (Note that a shortest closed walk through p and q is simply the union of a shortest path from p to q and a shortest path from q to p.) The maximum approximated dilation over all pairs is an approximation of the oriented dilation of the given graph. See Algorithm 3.2 for details.

Theorem 3.5. *Let P be a set of n points in $\mathbb{R}^d$, let $\overrightarrow{G}$ be an oriented graph on P, and let $\varepsilon > 0$ be a real number. Assume that, in $T(n)$ time, we can construct a data structure such that, for any two query points p and q in P, we can return, in $f(n)$ time, a k-approximation to the length of a shortest path in $\overrightarrow{G}$ between p and q. We can compute a value $\overline{\text{odil}}$ in $\mathcal{O}(T(n) + 1/\varepsilon^d n(d \log n + f(n)))$ time, such that*

$$(1-\varepsilon) \cdot \text{odil}(\overrightarrow{G}) \leq \overline{\text{odil}} \leq k \cdot \text{odil}(\overrightarrow{G}).$$

Proof. Let the constants in Algorithm 3.2 be equal to $\varepsilon_1 = \varepsilon/2$ and $s = 20/\varepsilon$. Let t^* be the dilation of the given oriented graph $\overrightarrow{G}$ on P. We prove that

$$(1-\varepsilon) \cdot t^* \leq \overline{\text{odil}} \leq k \cdot t^*,$$

where $\overline{\text{odil}}$ is the value returned by Algorithm 3.2.

Algorithm 3.2 Approximation of the Oriented Dilation.

Require: Oriented graph $\overrightarrow{G} = (P, \overrightarrow{E})$, positive reals ε, ε_1, s, parameter k
Ensure: Value $\overline{\text{odil}}$, such that $(1-\varepsilon)\,\text{odil}(\overrightarrow{G}) \leq \overline{\text{odil}} \leq k\,\text{odil}(\overrightarrow{G})$
  Compute s-WSPD with the pairs $\{A_i, B_i\}$ for $1 \leq i \leq m$
  Compute the data structure on P from [10] to approximate nearest neighbors
  $\overline{\text{odil}} = 1$
  for $i = 1$ **to** m **do**
    $\mathcal{A}_i$ = Pick $\min(|A_i|, 2)$ arbitrary points in A_i
    $\mathcal{B}_i$ = Pick $\min(|B_i|, 2)$ arbitrary points in B_i
    for each $ab \in \mathcal{A}_i \times \mathcal{B}_i$ **do**
      $\overline{|d(a,b)|}$ = k-approximated length of a shortest path from a to b in $\overrightarrow{G}$
      $\overline{|d(b,a)|}$ = k-approximated length of a shortest path from b to a in $\overrightarrow{G}$
      $\Delta_{abr'} = (1+\varepsilon_1)$-approximation of $\Delta^*(a,b)$ by Algorithm 3.1
      $\overline{\text{odil}} = \max\left(\frac{\overline{|d(a,b)|}+\overline{|d(b,a)|}}{|\Delta_{abr'}|}, \overline{\text{odil}}\right)$
  return $\overline{\text{odil}}$

Upper bound. For each well-separated pair $\{A_i, B_i\}$, Algorithm 3.2 considers up to four pairs ab in $\mathcal{A}_i \times \mathcal{B}_i$ and approximates the length of a shortest closed walk $C_{\overrightarrow{G}}(a,b)$ through a and b and their minimum perimeter triangle. It holds that

$$\overline{\text{odil}} = \max_{\substack{ab \in \mathcal{A}_i \times \mathcal{B}_i \\ 1 \leq i \leq m}} \frac{k|C_{\overrightarrow{G}}(a,b)|}{(1+\varepsilon_1)|\Delta^*(a,b)|} \leq k \cdot \max_{\substack{ab \in \mathcal{A}_i \times \mathcal{B}_i \\ 1 \leq i \leq m}} \text{odil}(a,b) \leq k \cdot \text{odil}(\overrightarrow{G}) = k \cdot t^*.$$

Lower bound. Let $p, q \in P$ be a worst-case pair in $\overrightarrow{G}$, i.e., $\text{odil}(p,q) = t^*$. Let $\{A_i, B_i\}$ be the well-separated pair with $p \in A_i$ and $q \in B_i$. We distinguish cases based on whether p and/or q are picked by Algorithm 3.2 while processing the pair $\{A_i, B_i\}$ or not.

Case 1: Neither p nor q is picked by Algorithm 3.2
Let $a \in A_i \setminus \{p\}$ and $b \in B_i \setminus \{q\}$ be the points picked by Algorithm 3.2.
First, we bound the length of a shortest closed walk containing p and q by

$$\begin{aligned} |C_{\overrightarrow{G}}(p,q)| &\leq |C_{\overrightarrow{G}}(p,a)| + |C_{\overrightarrow{G}}(a,b)| + |C_{\overrightarrow{G}}(b,q)| \\ &\leq t^*|\Delta^*(p,a)| + |C_{\overrightarrow{G}}(a,b)| + t^*|\Delta^*(b,q)|. \end{aligned}$$

Since p and q are both not picked, this implies $|A_i| \geq 3$ and $|B_i| \geq 3$. Therefore, $|\Delta^*(p,a)|$ is at most the perimeter of any triangle with vertices p, a, and one other point in A_i. Each of the three edges of such a triangle can be bounded using WSPD-Property 1. It holds that

$$|\Delta^*(p,a)| \leq 3 \cdot 2/s \cdot |pq| \leq 3/s|\Delta^*(p,q)|,$$

where the last inequality $|\Delta^*(p,q)| \geq 2|pq|$ follows from the triangle inequality. Analogously, we bound $|\Delta^*(b,q)|$ by $3/s|\Delta^*(p,q)|$. It follows that

$$|C_{\overrightarrow{G}}(p,q)| \leq 2 \cdot 3/s \cdot t^*|\Delta^*(p,q)| + |C_{\overrightarrow{G}}(a,b)| = 6/s|C_{\overrightarrow{G}}(p,q)| + |C_{\overrightarrow{G}}(a,b)|.$$

We use $|\Delta^*(a,b)| \le (1+4/s)|\Delta^*(p,q)|$ (due to Lemma 3.3) to bound

$$\begin{aligned} |C_{\overrightarrow{G}}(p,q)| &\le \frac{s}{s-6} \cdot |C_{\overrightarrow{G}}(a,b)| = \frac{s}{s-6}\, \mathrm{odil}(a,b)|\Delta^*(a,b)| \\ &\le \frac{s+4}{s-6} \cdot \mathrm{odil}(a,b)|\Delta^*(p,q)|. \end{aligned}$$

Since $t^* = \frac{|C_{\overrightarrow{G}}(p,q)|}{|\Delta^*(p,q)|}$, this leads us to a lower bound of the dilation of a and b, which is

$$\mathrm{odil}(a,b) \ge \frac{s-6}{s+4} \cdot t^* \ge (1-10/s)\, t^*.$$

For the picked points a and b, Algorithm 3.2 computes an approximation of $C_{\overrightarrow{G}}(a,b)$ and a $(1+\varepsilon_1)$-approximation of $\Delta^*(a,b)$. It holds that

$$\overline{\mathrm{odil}} \ge \frac{|C_{\overrightarrow{G}}(a,b)|}{(1+\varepsilon_1)|\Delta^*(a,b)|} \ge (1-\varepsilon_1)\, \mathrm{odil}(a,b) \ge (1-\varepsilon_1)\,(1-10/s)\, t^*.$$

Since $\varepsilon_1 = \varepsilon/2$ and $s = 20/\varepsilon$, we conclude that

$$\overline{\mathrm{odil}} \ge (1-\varepsilon/2)^2 \cdot t^* \ge (1-\varepsilon)t^*.$$

Case 2: Either p or q is picked by Algorithm 3.2
W.l.o.g., let p be picked for A_i and $b \in B_i \setminus \{q\}$ be picked for B_i.
Since q is not picked for B_i, it holds $|B_i| \ge 3$. Analogously to Case 1, we show

$$\begin{aligned} |C_{\overrightarrow{G}}(p,q)| &\le |C_{\overrightarrow{G}}(p,b)| + |C_{\overrightarrow{G}}(b,q)| \\ &\le |C_{\overrightarrow{G}}(p,b)| + 3/s \cdot t^*|\Delta^*(p,q)| \\ &\le \frac{s+4}{s-6} \cdot \mathrm{odil}(p,b)|\Delta^*(p,q)|, \end{aligned}$$

which leads to

$$\overline{\mathrm{odil}} \ge (1-\varepsilon_1)\, \mathrm{odil}(p,b) \ge (1-\varepsilon)t^*.$$

Case 3: Both p and q are picked by Algorithm 3.2
Since Algorithm 3.2 computes an approximation of $C_{\overrightarrow{G}}(p,q)$ and $(1+\varepsilon_1)$-approximation of $\Delta^*(p,q)$, it holds that

$$\overline{\mathrm{odil}} \ge \frac{|C_{\overrightarrow{G}}(p,q)|}{(1+\varepsilon_1)|\Delta^*(p,q)|} \ge (1-\varepsilon_1)\, \mathrm{odil}(p,q) \ge (1-\varepsilon)t^*,$$

where the last inequality follows from $\varepsilon_1 = \varepsilon/2$.

The preprocessing is dominated by computing the data structure for answering approximate shortest path queries and a well-separated pair decomposition in $\mathcal{O}(T(n) + n\log n + s^d n)$ time (WSPD-Property 2). For each of the $\mathcal{O}(s^d n)$ pairs, we consider up to four pairs ab and approximate their minimum perimeter triangle in $\mathcal{O}(s^d d \log n)$ time (compare to Lemma 3.4). The runtime to approximate the length of a shortest closed walk through a and b depends on the runtime $f(n)$ of the used path approximation. Therefore, the oriented dilation of a given graph can be approximated in $\mathcal{O}(T(n) + s^d nd\log n + s^d n f(n))$ time. □

The precision and runtime of our dilation approximation depend on approximating the length of a shortest path between two query points. The point-to-point shortest path problem in directed graphs is an extensively studied problem (see [53] for a survey). Alternatively, shortest path queries can be preprocessed via approximate APSP (see [42, 81, 90]). For planar directed graphs, the authors in [8] present a data structure built in $\mathcal{O}(n^2/r)$ time, such that the exact length of a shortest path between two query points can be returned in $\mathcal{O}(\sqrt{r})$ time. Setting $r = n^{2/3}$ minimizes the runtime of our algorithm to $\mathcal{O}(n^{4/3})$. It should be noted that even with exact shortest path queries ($k = 1$) the computed value remains an approximation due to the approximation of the minimum perimeter triangle.

Chapter 4
Orienting Undirected Graphs

A natural approach to construct an oriented spanner is to first compute a suitable undirected graph and then orient it. For instance, we obtain a plane oriented spanner by orienting a greedy triangulation (see Section 6.2). Also in Sections 7.3 and 7.4, we first construct a sparse undirected graph and then orient it greedily.

In Section 4.1, we show that, in general, it is NP-hard to decide whether the edges of an arbitrary graph can be oriented in such a way that the oriented dilation of the resulting graph is below a given threshold, even for Euclidean graphs.

In Section 4.2, we present the above mentioned greedy orientation algorithm used in Sections 6.2, 7.3 and 7.4.

As our NP-hardness construction does not hold for complete graphs, in Section 4.3, we look into tournaments. Our best construction for arbitrary metric point sets of size n is a 5/3-tournament in $\mathcal{O}(n^3)$ time. In Section 7.2, we present an algorithm that sparsifies a dense graph, combining this algorithm with our 5/3-tournament leads to a sparse $(5/3+\varepsilon)$-spanner (see Section 7.3).

While in the undirected setting, the complete graph is a 1-spanner, this is not always true for oriented graphs. Since adding edges can only decrease or preserve the dilation, analyzing the oriented dilation of tournaments bounds the minimum oriented dilation. In Section 4.4, we show that the worst-case dilation t of the minimum dilation tournament is bounded by $1.5 \leq t \leq 5/3$ for metric point sets and $2\sqrt{3}-2 \leq t \leq 5/3$ for points in a Euclidean space.

4.1 Hardness

In this section, we prove that, given an undirected graph, it is NP-hard to find its minimum dilation orientation.

Theorem 4.1. *Let G be an undirected graph whose vertex set is a finite set of points in a metric space. Given a real number t', it is NP-hard to decide if there exists an orientation $\overrightarrow{G}$ of G with oriented dilation $\mathrm{odil}(\overrightarrow{G}) \leq t'$. This is even true for point sets in the Euclidean plane.*

Proof. We reduce from the NP-complete problem planar 3-SAT [71]. We start with a planar Boolean formula φ in conjunctive normal form with an incidence graph G_φ as illustrated in Figure 4.1. This graph can be embedded on a polynomial-size grid whose cells have unit length [52, 71]. In this proof, we construct a graph G based on G_φ such that there is an orientation $\overrightarrow{G}$ of G with $\mathrm{odil}(\overrightarrow{G}) \le t'$ for $t' := 1.043$ if and only if φ is satisfiable.

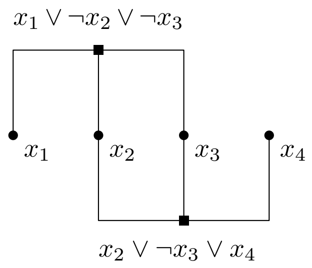


Figure 4.1: Example: Incidence graph of a planar 3-SAT formula $\varphi = (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_2 \vee \neg x_3 \vee x_4)$ embedded on a square grid.

First, we add (a polynomial number of) grid points on the edges such that all edges have length 1. Then, every point p with grid coordinates (x, y) is replaced by a so-called *oriented point*, which is a pair of points $P = \{t(p), b(p)\}$ with *top* $t(p) = (x, y + \frac{\varepsilon}{2})$ and *bottom* $b(p) = (x, y - \frac{\varepsilon}{2})$ with $\varepsilon := 10^{-6}$. Note that p .denotes the original point respectively grid coordinates, while P refers to the associated oriented point as a single conceptual unit.

We extend the definition of the dilation to oriented points as follows. We define $\mathrm{odil}(P, P')$ as the maximum dilation between one point of P and one of P', i. e.,

$$\mathrm{odil}(P, P') = \max\{\mathrm{odil}(t(p), t(p')), \mathrm{odil}(b(p), t(p')), \mathrm{odil}(t(p), b(p')), \mathrm{odil}(b(p), b(p'))\}.$$

Analogously, by $\Delta^*(P, P')$, we denote the minimum triangle containing one of $\{t(p), b(p)\}$ and one of $\{t(p'), b(p')\}$. To bound $\mathrm{odil}(P, P')$, we repeatedly use

$$2|pp'| \le |\Delta^*(P, P')| \le 2|pp'| + 2\varepsilon.$$

We add an edge between $t(p)$ and $b(p)$, its orientation encodes whether the point represents “true” or “false”. Within this proof, we assume without loss of generality that an edge directed from $b(p)$ to $t(p)$, i.e., an *upward* edge, represents “true”, while a *downward* edge represents “false”. If this is not the case, we can simply flip the orientation of all edges.

Our construction consists of the following ingredients:

- *Wire*: Connects two oriented points and propagates their orientation: in any orientation of $\overrightarrow{G}$ with dilation at most t', both points must have the same orientation.

- *Tree of knowledge*: A tree of wires that fixes whether *upwards* or *downwards* represents “true” across the entire construction.

- *$K_{2,2}$ gadget*: Connects all pairs of oriented points P, P' that are neither wire-neighbours nor part of a clause gadget, guaranteeing $\mathrm{odil}(P, P') \le t'$ irrespective of their orientation. Thus, they never dominate $\mathrm{odil}(\overrightarrow{G})$.

- *Clause gadget*: Connects the three wires corresponding to the literals of a clause and enforces that in any orientation with dilation at most t', at least one wire must match the orientation of the tree of knowledge (i.e., at least one literal is "true").

Wire gadget. Since grid points are replaced by oriented points, also the edges in the plane embedding of our formula graph G_φ are replaced. We add edges between top-bottom and bottom-top of each pair of oriented points whose original points are adjacent (see Figure 4.2). Note that wires propagate the orientation of oriented points: if two *neighbors along a wire* have different orientations, we have $\mathrm{odil}(P, P') \geq 2 - \varepsilon > 1.043$, since their shortest closed walk visits another point. If they have the same orientation, it holds $\mathrm{odil}(P, P') \leq 1 + 2\varepsilon \leq t'$.

To switch a signal (for a negated variable in a clause), we start a wire by top-top and bottom-bottom edges as shown in Figure 4.3 and then continue with the standard wire.

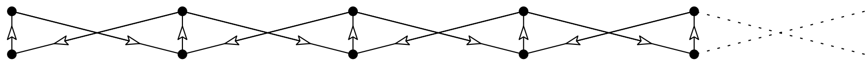


Figure 4.2: A wire where oriented points are oriented upwards.

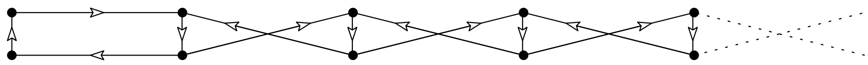


Figure 4.3: A wire where the orientation is switched to negate the signal.

Tree of knowledge. To ensure that all clause gadgets encode the same orientation of oriented points as "true", we add a *tree of knowledge*. This is a tree with vertices on a unit grid shifted by $(0.5, 0.5)$ relative to the grid of G_φ. Again, we use wires as edges. The tree has two leaves per clause (see Figure 4.4). W.l.o.g., we assume that all its oriented points of the tree of knowledge are oriented upwards (thus "true").

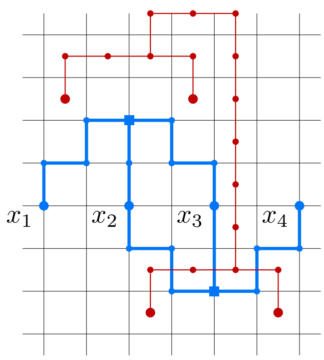


Figure 4.4: G_φ (blue) and its underlying grid together with a tree of knowledge (red).

$\mathbf{K_{2,2}}$ gadget. All oriented points, which are not neighbors along a wire, are linked by the four possible edges between top-top, bottom-bottom, top-bottom and bottom-top (compare to Figure 4.5). This ensures $\mathrm{odil}(P, P') \leq 1 + \varepsilon \leq t'$ for non-neighboring points.

For an oriented point P, the shortest closed walk containing $t(p)$ and $b(p)$ visits the closest point p' of p. Therefore, it holds that $\mathrm{odil}(t(p), b(p)) \leq 1 + 2\varepsilon$, both if p and p' are neighbors (shortest closed walk is through a wire) and not (through $K_{2,2}$).

Figure 4.5: $K_{2,2}$ for oriented points with same orientation (left) and different orientation (right).

Adding these $K_{2,2}$ gadgets does not affect the functionality of wires: Let p and p' be neighbors and p^* a third point. Since p^* has at least distance $(0.5, 0.5)$ to p and p', a walk from p to p' visiting p^* has at least length $1 + \sqrt{2 - 2\varepsilon} > 1.043$. For $\varepsilon = 10^{-6}$, we have $\mathrm{odil}(P, P') > t'$ if P and P' have a different orientation. Thus, a wire cannot be shortcut by $K_{2,2}$ gadgets (compare to Figure 4.6).

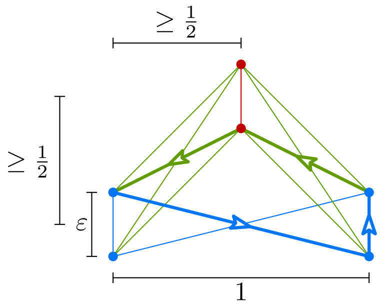


Figure 4.6: $K_{2,2}$ gadgets (green) do not affect the functionality of a wire (blue): The shortest closed walk through the $K_{2,2}$ gadgets is longer than $1 + \sqrt{2 - 2\varepsilon} > t'$ irrespective of its orientation.

Clause gadget. For each clause, the tree of knowledge has two leaves l_1 and l_2, which are connected by a *clause gadget* instead of a $K_{2,2}$ gadget. The clause gadget connects the leaves through the three incoming wires corresponding to the literals of the clause. Its geometry ensures that $\mathrm{odil}(L_1, L_2) > t'$, if all three wires are oriented downwards (all literals are “false”) and $\mathrm{odil}(L_1, L_2) \leq t'$, if at least one wire is oriented upwards (the clause is “true”).

Figure 4.7 shows the two leaves and G_φ at a clause. We may assume that G_φ is embedded as shown, in particular leaving the area directly above the clause empty.

The points l (left), b (bottom) and r (right) are the endpoints of the paths in G_φ from the variables to the clause. As shown in Figures 4.7 and 4.8, they are placed such that they lie just inside an ellipse with l_1 and l_2 as foci, and without any other points in the ellipse.

As shown in Figure 4.9, the five oriented points inside the ellipse are connected by the edges $t(l_1)b(p)$ and $t(p)b(l_2)$ for $p \in \{l, b, r\}$ and the edge $t(l_2)b(l_1)$. Note that there is no $K_{2,2}$ between the leaves and the wire endpoints, but there is one between all other non-neighboring points (except the leaves), including between L, B and R (see Figure 4.9).

This creates three possible closed walks containing L_1 and L_2, each passing through one of $\{L, B, R\}$. If one of $\{L, B, R\}$ is oriented upwards, the edges of the clause gadget can be oriented so that the corresponding walk yields $\mathrm{odil}(L_1, L_2) \leq t'$.

As shown in Figures 4.8 and 4.9, for each of $\{l, b, r\}$ there is a *satellite* point s_l, s_b, s_r located on the corresponding path between the variable and the clause. Each satellite lies close to l, b, r, but outside the ellipse. Their purpose is to ensure that the oriented dilation of L_1 (and L_2) and each of $\{L, B, R\}$ is below t', even if these are oriented downwards. Consequently, we enforce $\mathrm{odil}(\overrightarrow{G}) = \mathrm{odil}(L_1, L_2)$.

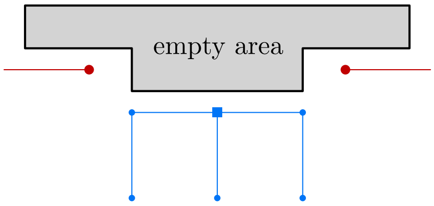


Figure 4.7: Embedded clause.

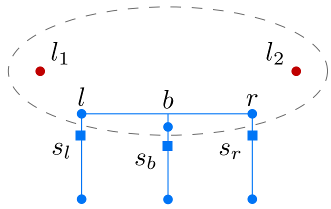


Figure 4.8: Clause gadget.

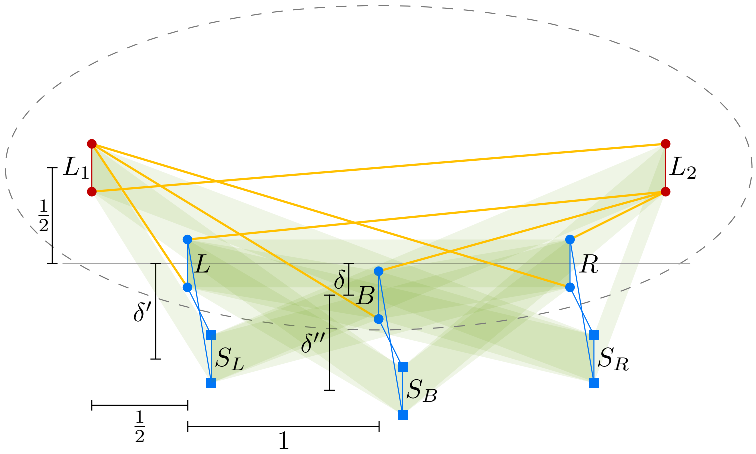


Figure 4.9: Detailed clause gadget. $K_{2,2}$ gadgets are indicated by green parallelograms. For visibility, the satellite points are drawn slightly offset instead of exactly below their corresponding literals.

To impose $\mathrm{odil}(\overrightarrow{G}) = \mathrm{odil}(L_1, L_2)$, the points l, b, r and their satellites must be placed carefully. Consider Figure 4.9. The clause is embedded such that the horizontal distances between l and b, between and b and r, are 1, while the horizontal distances between l_1 and l, and between r and l_2, are $1/2$. The vertical distance from l_1 (and l_2) to l and r is $1/2$, while their vertical distance to B is $1/2 + \delta$. Let δ' denote the distance $|ls_l| = |rs_r|$ and let $\delta'' = |bs_b|$. In the following, we set values for δ, δ' and δ'', which determines the exact placement of l, b, r and their satellites.

First, δ' has to be set such that the satellite is not in the ellipse, but the dilation of a leaf and L or R is bounded by t', even if the shortest closed walk is through the $K_{2,2}$ between the leaf and the satellite. Thus, we set $\delta' := 0.0152$ and achieve

$$\begin{aligned}\operatorname{odil}(L_1, L) &\leq \frac{2|l_1 s_l| + 2|s_l l| + 5\varepsilon}{2|l_1 l|} \leq \frac{|l_1 l| + |l s_l| + \delta' + 4\varepsilon}{|l_1 l|} \\ &\leq 1 + \frac{2\delta' + 3\varepsilon}{\sqrt{\left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2}} \leq 1 + 2\sqrt{2}\delta' + 5\varepsilon \leq 1.043.\end{aligned}$$

Analogously, we can upper bound $\operatorname{odil}(L_2, L)$, $\operatorname{odil}(L_1, R)$ and $\operatorname{odil}(L_2, R)$ by 1.043.

As can be seen in Figures 4.8 and 4.9, the point b has to be shifted by $(0, -\delta)$ such that it is only just contained in the ellipse. Thus, δ has to be set such that $|l_1 b| + |b l_2| \leq |l_1 l| + |l l_2|$. This is true for $\delta := 0.1335$:

$$\begin{aligned}|l_1 b| + |b l_2| &= 2\sqrt{\left(\frac{1}{2} + \delta\right)^2 + \left(\frac{3}{2}\right)^2} \\ &\leq |l_1 l| + |l l_2| = \sqrt{\left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2} + \sqrt{\left(\frac{1}{2}\right)^2 + \left(\frac{5}{2}\right)^2}.\end{aligned}$$

The wire from below to B is disrupted by a δ'' close satellite which is not contained in the ellipse, to ensure that the dilation of the leaves and B is bounded by t', even if the shortest closed walk is through the $K_{2,2}$ to the satellite. Thus, we set $\delta'' := 0.0304$ and achieve

$$\begin{aligned}\operatorname{odil}(L_1, B) &\leq \frac{2|l_1 s_b| + 2|s_b b| + 5\varepsilon}{2|l_1 b|} \leq \frac{|l_1 b| + 2\delta'' + 3\varepsilon}{|l_1 b|} = 1 + \frac{2\delta'' + 3\varepsilon}{\sqrt{\left(\frac{1}{2} + \delta\right)^2 + \left(\frac{3}{2}\right)^2}} \\ &\leq 1 + \frac{2\delta'' + 3\varepsilon}{\sqrt{2 + \delta}} \leq 1 + \frac{2\delta'' + 3\varepsilon}{\sqrt{2}} \leq 1 + \sqrt{2}\delta'' + 3\varepsilon \leq 1.043\end{aligned}$$

Since $\operatorname{odil}(L_1, B) = \operatorname{odil}(L_2, B)$, it also holds that $\operatorname{odil}(L_2, B) \leq 1.043$.

If L has the same orientation as the tree of knowledge, the dilation is

$$\begin{aligned}\operatorname{odil}(L_1, L_2) &\leq \frac{|l_1 l| + |l l_2| + |l_1 l_2| + 6\varepsilon}{2|l_1 l_2|} \\ &= \frac{\sqrt{\left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2} + \sqrt{\left(\frac{1}{2}\right)^2 + \left(\frac{5}{2}\right)^2} + 3}{6} + \varepsilon \leq 1.043.\end{aligned}$$

The same holds if R has the same orientation as the tree of knowledge.

If B has the same orientations as the tree of knowledge, the dilation is

$$\operatorname{odil}(L_1, L_2) \leq \frac{|l_1 b| + |b l_2| + |l_1 l_2| + 6\varepsilon}{2|l_1 l_2|} = \frac{2\sqrt{\left(\frac{1}{2} + \delta\right)^2 + \left(\frac{3}{2}\right)^2} + 3}{6} + \varepsilon \leq 1.043.$$

If L, B and R have a different orientation than the tree of knowledge (i.e., every literal is "false"), by triangle inequality there are the following possible shortest closed walks through L_1 and L_2 which leads to three lower bounds for $\operatorname{odil}(L_1, L_2)$:

- A closed walk is from L_1 to the satellite of L (or R symmetric) to L_2 and back to L_1:

$$\begin{aligned}\mathrm{odil}(L_1, L_2) &\geq \frac{|l_1 s_l| + |s_l l_2| + |l_1 l_2|}{2|l_1 l_2| + 2\varepsilon} \\ &\geq \frac{\sqrt{(\frac{1}{2}+\delta')^2 + \left(\frac{1}{2}\right)^2} + \sqrt{\left(\frac{1}{2}+\delta'\right)^2 + \left(\frac{5}{2}\right)^2} + 3}{6} \cdot (1-\varepsilon) > 1.043.\end{aligned}$$

- A closed walk is from L_1 to the satellite of B to L_2 and back to L_1:

$$\begin{aligned}\mathrm{odil}(L_1, L_2) &\geq \frac{|l_1 s_b| + |s_b l_2| + |l_1 l_2|}{2|l_1 l_2| + 2\varepsilon} \\ &\geq \frac{2\sqrt{\left(\frac{1}{2}+\delta+\delta''\right)^2 + \left(\frac{3}{2}\right)^2} + 3}{6} \cdot (1-\varepsilon) > 1.043.\end{aligned}$$

- A closed walk is from L_1 to L to B to L_2 (or symmetric through B to R) and back to L_1:

$$\begin{aligned}\mathrm{odil}(L_1, L_2) &\geq \frac{|l_1 l| + |lb| + |bl_2| + |l_1 l_2|}{2|l_1 l_2| + 2\varepsilon} \\ &\geq \frac{\sqrt{\left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^2} + \sqrt{\delta^2+1} + \sqrt{\left(\frac{1}{2}+\delta\right)^2 + \left(\frac{3}{2}\right)^2} + 3}{6} \cdot (1-\varepsilon) \\ &> 1.043.\end{aligned}$$

Thus, we obtain the following conditions:

- If one of the oriented points L, B, R has the same orientation as the tree of knowledge (i.e., the literal is "true"), the dilation between L_1 and L_2 is smaller than $t' := 1.043$.
- If all of the oriented points L, B, R have the opposite orientation of the tree of knowledge (i.e., all literals are "false"), the shortest closed walk containing L_1 and L_2 either leaves the ellipse or takes at least two points from $\{L, B, R\}$ and thus their dilation is greater than t'.

Therefore, formula φ is satisfiable if and only if there exists an orientation of our constructed graph with dilation at most 1.043.

Size of the construction. The reduction runs in polynomial time in $|\varphi|$. The incidence graph G_φ can be embedded on a grid of polynomial size with unit-length cells [52, 71], and subdividing its edges to unit length introduces only polynomially many additional grid points. The tree of knowledge is likewise a subdivided grid graph and therefore contributes only polynomially many further points. Each clause gadget adds a constant number of points per clause. Replacing every grid point by an oriented point doubles the number of points. So, it remains polynomial in $|\varphi|$. Each of the polynomial number of original unit-length edge is replaced by a wire of two edges and each clause adds only a constant number of additional edges. Each point of the subdivided grid has a polynomial number of non-adjacent points, and a $K_{2,2}$ gadget adds four edges for every such non-adjacent pair. Thus, the total number of edges remains polynomial as well. Consequently, both the number of points and edges of the constructed graph G are polynomial in $|\varphi|$, and G can be constructed in polynomial time. □

Remark. *In the algebraic model the oriented dilation is computable in cubic time (for details see Chapter 3). However, establishing containment in NP for the problem of deciding whether there is an orientation of a given graph with dilation at most t' requires the Turing model. Here, computing the oriented dilation reduces to the square-root sum problem [75], for which it is open whether it lies in NP.*

Following the construction in the proof of Theorem 4.1, Figure 4.10 illustrates the graph for the formula $\varphi = (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_2 \vee \neg x_3 \vee x_4)$.

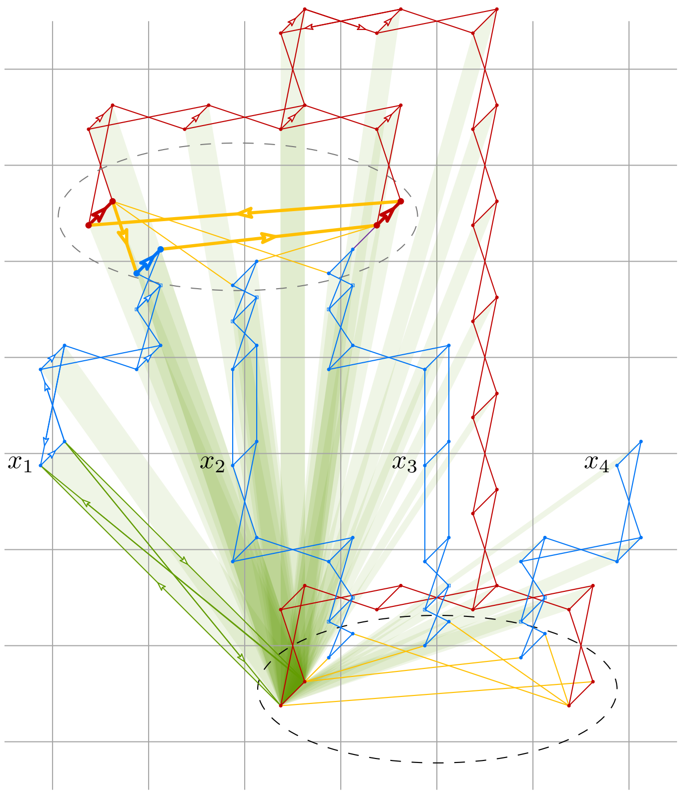


Figure 4.10: Graph constructed for $\varphi = (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_2 \vee \neg x_3 \vee x_4)$. For visibility, oriented points are placed diagonally instead of vertically. Only for one point, all $K_{2,2}$ gadgets are indicated by green parallelograms. If the oriented point at the end of the wire from x_1 (blue) is, as indicated, oriented the same way as the tree of knowledge (red), this corresponds to setting x_1 to "true", resulting in an oriented closed walk in the clause gadget (yellow) that gives a dilation smaller than 1.043.

4.2 Greedy Orientation

Given a set of triangles, we orient these greedily in a consistent manner (compare to Algorithm 4.1):

1. Sort the input triangles in ascending order by their perimeter.
2. In this order, orient each triangle clockwise or anti-clockwise if possible; otherwise, we skip this triangle.
3. At the end of the algorithm, orient all remaining edges arbitrarily.

By this, we orient the undirected graph whose edge set consists of these triangles. Shortest closed walks in the resulting oriented graph are bounded relatively to the input triangles.

Algorithm 4.1 Greedy 2-Orientation.

Require: Set P of n points, list L of m triangles Δ_{pqr} in P
Ensure: Orientation $\overrightarrow{G} = (P, \overrightarrow{E})$ of the undirected graph whose edge set consists of all edges of the triangles in L
  Sort the triangles Δ_{pqr} in L ascending by perimeter
  for each Δ_{pqr} **do**
    if no edge of Δ_{pqr} is oriented **then**
      Orient (p, q) arbitrarily and orient Δ_{pqr} consistently according to (p, q)
    else
      Orient Δ_{pqr} consistently, if this is still possible; otherwise, skip Δ_{pqr}
  Orient the remaining edges arbitrarily

Lemma 4.2. *Let $(P, |\cdot|)$ be a finite metric space with n points. Let L be a list of m triangles Δ_{pqr} in P. Let $G = (P, E)$ be the undirected graph whose edge set consists of all edges of the triangles in L. In $\mathcal{O}(m \log m)$ time, we can compute an orientation $\overrightarrow{G}$ of G, such that for any two distinct points $p, q \in P$ it holds $|C_{\overrightarrow{G}}(p, q)| \leq \min_{\Delta_{pqx} \in L} 2|\Delta_{pqx}|$.*

In particular, there exists a cycle C in $\overrightarrow{G}$ through p and q which consists of at most four edges and whose length is $|C| \leq \min_{\Delta_{pqx} \in L} 2|\Delta_{pqx}|$.[1]

Proof. We prove that Algorithm 4.1 computes such an orientation $\overrightarrow{G} = (P, \overrightarrow{E})$.

Let p and q be two distinct points in P. Let r be a point in $P \setminus \{p, q\}$ such that $|\Delta_{pqr}| = |\Delta^*(p, q)|$. If there is no such triangle in L, then $|C_{\overrightarrow{G}}(p, q)| = \infty$.

If Δ_{pqr} is oriented consistently then

$$|C_{\overrightarrow{G}}(p, q)| = |\Delta_{pqr}| \leq \min_{\Delta_{pqx} \in L} 2|\Delta_{pqx}|.$$

Otherwise, pr and qr were already oriented, when Algorithm 4.1 processed Δ_{pqr}. Their orientation is either (p, r) and (q, r) or (r, p) and (r, q). We show the first case, the other case is analogous.

[1] The number of edges is relevant for the proof of Lemma 7.5.

Since pr and qr are oriented before Algorithm 4.1 processed Δ_{pqr}, $\overrightarrow{G}$ contains a consistently oriented triangle Δ_{prx} and a consistently oriented triangle Δ_{qry} with $x, y \in P \setminus \{p, q, r\}$. Due to the greedy order, the perimeter of both is at most $|\Delta_{pqr}|$.

Case 1: pq is oriented as (p, q)
Then $\overrightarrow{G}$ contains the cycle

$$C = p \to q \to r \to x \to p$$

of length

$$|C| = |pq| + |qr| + |rx| + |xp| \leq |\Delta_{pqr}| + |\Delta_{prx}| \leq 2|\Delta_{pqr}|.$$

Thus, $|C_{\overrightarrow{G}}(p, q)| \leq \min\limits_{\Delta_{pqx} \in L} 2|\Delta_{pqx}|$.

Case 2: pq is oriented as (q, p)
Then $\overrightarrow{G}$ contains the cycle

$$C = q \to p \to r \to y \to q$$

of length

$$|C| = |pq| + |qr| + |ry| + |yq| \leq |\Delta_{pqr}| + |\Delta_{qry}| \leq 2|\Delta_{pqr}|.$$

Thus, $|C_{\overrightarrow{G}}(p, q)| \leq \min\limits_{\Delta_{pqx} \in L} 2|\Delta_{pqx}|$.

Sorting the triangles costs $\mathcal{O}(m \log m)$ time. Each triangle has three pointers to its edges, so accessing and orienting an edge takes $\mathcal{O}(1)$ time. Thus, the total runtime is $\mathcal{O}(m \log m)$. □

4.3 Tournaments

The complete graph is an interesting graph class due to its rich structure and properties.

Observation. *Let $(P, |\cdot|)$ be a finite metric space and $\overrightarrow{K}$ a tournament on P. For two distinct points $p, q \in P$, the shortest closed walk $C_{\overrightarrow{K}}(p, q)$ is a cycle.*

Inputting all $\binom{n}{3}$ triangles in a given point set P of size n, Algorithm 4.1 orients the complete graph on P in $\mathcal{O}(n^3 \log n)$ time (compare to Lemma 4.2). To improve the runtime, we precompute the minimum perimeter triangle for any pair of points in P in $\mathcal{O}(n^3)$ time. With these $\mathcal{O}(n^2)$ triangles as input Algorithm 4.1 computes a 2-tournament in $\mathcal{O}(n^2 \log n)$ time.

Corollary 4.3. *For every metric space $(P, |\cdot|)$ with n points, a 2-tournament on P can be computed in $\mathcal{O}(n^3)$ time.*

In Section 7.3 we construct a sparse $(2 + \varepsilon)$-spanner in $\mathcal{O}(n \log n)$ time for point sets in a Euclidean space $\mathbb{R}^d$. By orienting the remaining edges randomly, we can compute a $(2 + \varepsilon)$-tournament in $\mathcal{O}(n^2)$ time.

Corollary 4.4. *Given a set P of n points in $\mathbb{R}^d$, where d is a constant and a real number $0 < \varepsilon < 2$, a $(2+\varepsilon)$-tournament on P can be constructed in $\mathcal{O}(1/\varepsilon^d n^2)$ time.*

Using a more complex greedy procedure, we improve upon the dilation of 2. We offer an algorithm that computes a 5/3-tournament on any metric point set of size n also in $\mathcal{O}(n^3)$ time. At the end of this section, we show the tightness of the approximation factor of this algorithm.

The 2-tournament is achieved by greedily orienting the minimum perimeter triangles of all pairs of points in ascending order of their perimeter (compare to Algorithm 4.1). The following algorithm has a different greedy strategy: We process all pairs of points sorted by distance and orient adjacent triangles consistently. This seemingly minor modification leads to a substantial change in the structure and properties of the resulting graph.

For the remainder of this section, we prove the following theorem.

Theorem 4.5. *For every metric space $(P, |\cdot|)$ with n points, a 5/3-tournament $\overrightarrow{K}$ on P can be computed in $\mathcal{O}(n^3)$ time.*

In particular, for any two distinct points $p, q \in P$, there is a cycle C in $\overrightarrow{K}$ through p and q which consists of at most four edges and whose length is $|C| \leq 5/3|\Delta^*(p,q)|$.[2]

Algorithm 4.2 Tournament with Dilation 5/3.

Require: Set P of n points in a metric space
Ensure: Tournament on P with dilation 5/3
  Set $m = \binom{n}{2}$
  Sort the m pairs of distinct points in P in non-decreasing order of their distances
  Denote the sorted sequence by $p_1q_1, p_2q_2, \ldots, p_mq_m$
  for $i = 1, 2, \ldots, m$ **do**
    if p_iq_i has not been oriented yet **then**
      if there is a triangle incident to p_iq_i that can be oriented consistently **then**
        Orient p_iq_i s.t. the smallest triangle $\Delta_{p_iq_ip'}$ with $p' \in P \setminus \{p_i, q_i\}$ can be oriented consistently
      else
        Orient p_iq_i arbitrarily
    for each $p \in P \setminus \{p_i, q_i\}$ **do**
      Orient $\Delta_{p_iq_ip}$ consistently, if possible, otherwise skip it

---

The algorithm that computes such an orientation is given in Algorithm 4.2. We sort all $m = \binom{n}{2}$ pairs of points ascending by their distance. In this order, we orient the current pair and, if possible, every adjacent triangle consistently. Observe that this algorithm orients all m edges of the complete graph on P. It is clear that the running time is $\mathcal{O}(n^3)$.

We proved a loop invariant for the outer loop of Algorithm 4.2. Since it is not used to show that a 5/3-tournament is returned, the result is moved to Appendix B.

The following observation is used repeatedly in the proof of Lemma 4.7.

[2]The number of edges is relevant for the proof of Corollary 7.3.

Observation 4.6. *If a pair pq is oriented in the i-th iteration, then*

- $|p_i q_i| \leq |pq|$,
- *at least one of p_i and q_i is in $\{p, q\}$, and*
- *if exactly one of $\{p_i, q_i\}$ is in $\{p, q\}$, say $p_i \in \{p, q\}$, then $\overrightarrow{G}$ contains the consistently oriented triangle Δ_{pqq_i}.*

By $\overrightarrow{E}_i$, we denote the set of edges that have an orientation at the beginning of the i-th iteration of the outer loop.

Lemma 4.7. *Let p and q be two distinct points in P. Let $\overrightarrow{K} = (P, \overrightarrow{E})$ be the tournament returned by Algorithm 4.2. It holds that*

$$\operatorname{odil}_{\overrightarrow{K}}(p, q) \leq 5/3.$$

Proof. Let r be a point in $P \setminus \{p, q\}$ such that $|\Delta_{pqr}| = |\Delta^*(p, q)|$. We prove that $\overrightarrow{K}$ contains a cycle through p and q whose length is at most $5/3|\Delta_{pqr}|$.

Let pq be oriented in the z-th be the iteration. W.l.o.g., let p_z be in $\{p, q\}$.

We consider the order in which the edges of Δ_{pqr} get an orientation. In particular, we distinguish whether pq is oriented as first, second, or third.

Case 1: pq is the first edge of Δ_{pqr} that gets an orientation
From the algorithm follows that

$$|p_z q_z| \leq \min(|pq|, |qr|, |pr|) \leq 1/3|\Delta_{pqr}|.$$

If $p_z = p$ and $q_z \notin \{p, q\}$, then $\overrightarrow{K}$ contains the consistently oriented triangle Δ_{pqq_z} of length

$$|\Delta_{pqq_z}| \leq 2|pq| + 2|p_z q_z| \leq |\Delta_{pqr}| + 2/3|\Delta_{pqr}| = 5/3|\Delta_{pqr}|.$$

The same holds for $p_z = q$ and $q_z \notin \{p, q\}$.

If $pq = p_z q_z$, then Δ_{pqr} is oriented consistently (in iteration z). In this case, $\overrightarrow{K}$ contains a cycle through p and q, whose length is equal to $|\Delta_{pqr}| \leq 5/3|\Delta_{pqr}|$.

Case 2: pq is the second edge of Δ_{pqr} that gets an orientation
Either Δ_{pqr} is oriented consistently or we have $|p_z q_z| \leq \min(|pq|, |qr|, |pr|)$. For the latter, we can show analogously to Case 1 that $\overrightarrow{K}$ contains a cycle through p and q, whose length is at most $5/3|\Delta_{pqr}|$.

Case 3: pq is the third edge of Δ_{pqr} that gets an orientation
Let pr be oriented in iteration x and qr be oriented in iteration y. W.l.o.g., we assume $x \leq y \leq z$, i.e., let pr be oriented as the first and qr as the second.

For $(p, r), (r, q) \in \overrightarrow{E}_z$ or $(r, p), (q, r) \in \overrightarrow{E}_z$, either Δ_{pqr} is oriented consistently or it holds that $|p_z q_z| \leq \min(|pq|, |qr|, |pr|)$ and $\overrightarrow{K}$ contains the oriented cycle Δ_{pqq_z}, whose length is at most $5/3|\Delta_{pqr}|$.

For $(p,r),(q,r) \in \overrightarrow{E}_z$ or $(r,p),(r,q) \in \overrightarrow{E}_z$, it is sufficient to show just one; the other case is symmetric, because, if we reverse the direction of all edges of $\overrightarrow{K}$, then the resulting tournament has the same dilation.

Assumption 1: For the remainder of this proof, we assume that $(p,r),(q,r) \in \overrightarrow{E}_z$.

If $p_x q_x = pr$, then Δ_{pqr} is oriented consistently (in iteration x).

Assumption 2: For the remainder of this proof, we assume that exactly one of p_x and q_x is in $\{p,r\}$. W.l.o.g., let $p_x \in \{p,r\}$ and $q_x \notin \{p,r\}$.

We distinguish the cases according to the orientation of pq.

Case 3.a: pq is oriented as (p,q)

Then $\overrightarrow{K}$ contains the cycle

$$C = p \to q \to r \to q_x \to p.$$

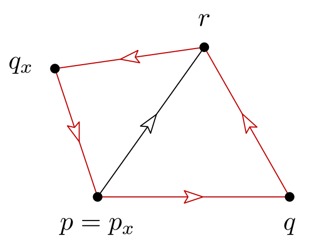
For $p_x = p$ and $q_x \notin \{p,r\}$ (see Figure 4.11), we have

$$\begin{aligned}|C| &= |pq| + |qr| + |rq_x| + |q_x p| \\ &\leq |pq| + |qr| + |rp| + |pq_x| + |q_x p| = |\Delta_{pqr}| + 2|p_x q_x|.\end{aligned}$$

The same holds for $p_x = r$ and $q_x \notin \{p,r\}$. Since pr is the first edge of Δ_{pqr} that gets an orientation, it follows from the algorithm that

$$|p_x q_x| \leq \min(|pq|, |pr|, |qr|) \leq 1/3|\Delta_{pqr}|.$$

This gives us $|C| \leq 5/3|\Delta_{pqr}|$.



Edge	Oriented in iteration
(p,r)	x (Case 3)
(q,r)	y (Case 3)
(p,q)	z (Case 3.a)

Figure 4.11: Case 3.a for $p_x = p$ and $q_x \notin \{p,r\}$. It holds that $|p_x q_x| \leq 1/3|\Delta_{pqr}|$. The length of the red cycle is at most $5/3|\Delta_{pqr}|$.

Case 3.b: pq is oriented as (q,p)

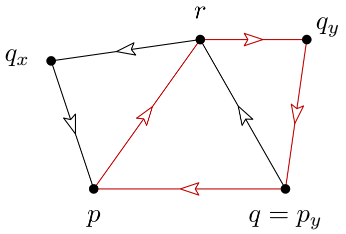
If exactly one of p_y and q_y is in $\{q,r\}$, say $p_y \in \{q,r\}$, then $\overrightarrow{K}$ contains the cycle

$$C = q \to p \to r \to q_y \to q.$$

For $p_y = q$ and $q_y \notin \{q,r\}$ (see Figure 4.12), we have

$$\begin{aligned}|C| &= |qp| + |pr| + |rq_y| + |q_y q| \\ &\leq |qp| + |pr| + |rq| + |qq_y| + |q_y q| = |\Delta_{pqr}| + 2|p_y q_y|.\end{aligned}$$

The same holds for $p_y = r$ and $q_y \notin \{q,r\}$.



Edge	Oriented in iteration
(p, r)	x (Case 3)
(q, r)	y (Case 3)
(q, p)	z (Case 3.b)

Figure 4.12: Case 3.b for $p_y = q$ and $q_y \notin \{q, r\}$. It holds that $|p_y q_y| \leq 1/3|\Delta_{pqr}|$. The length of the red cycle is at most $5/3|\Delta_{pqr}|$.

Since at the beginning of the y-th iteration pr is the only oriented edge in Δ_{pqr}, it holds that

$$|p_y q_y| \leq \min(|pq|, |pr|, |qr|) \leq 1/3|\Delta_{pqr}|.$$

So, if exactly one of $\{p_y, q_y\}$ is in $\{q, r\}$, then $|C| \leq 5/3|\Delta_{pqr}|$.

Assumption 3: For the remainder of this proof, we assume that $p_y q_y = qr$.

This also implies that $x < y$ (i.e., pr and qr are not oriented in the same iteration).

At the beginning of the y-th iteration, qr is unoriented. The first if-clause orients qr such that a triangle $\Delta_{qrp'}$ with $p' \in P \setminus \{q, r\}$ is oriented consistently. (There is such a triangle, since at the beginning of iteration y it is still possible to orient Δ_{pqr} consistently). Due to the algorithm, we have $|rp'| + |p'q| \leq |rp| + |pq|$.

If $p' = p$, then Δ_{pqr} is oriented consistently in iteration y.

Assumption 4: For the remainder of this proof, we assume that $p' \neq p$.

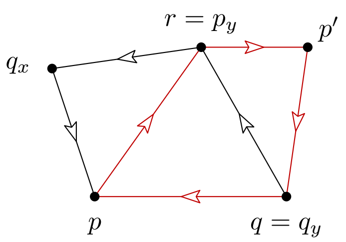
Consider Figure 4.13. If $p' \neq p$, then $\overrightarrow{K}$ contains the cycle

$$C = q \to p \to r \to p' \to q$$

of length

$$|C| = |qp| + |pr| + |rp'| + |p'q| \leq 2|pq| + 2|pr|. \qquad (*)$$

Since $2|pq| \leq |\Delta_{pqr}|$ and $2|pr| \leq |\Delta_{pqr}|$, if either $|pq|$ or $|pr|$ is bounded by $1/3|\Delta_{pqr}|$, then we have $|C| \leq 5/3|\Delta_{pqr}|$.



Edge	Oriented in iteration
(p, r)	x (Case 3)
(q, r)	y (Case 3)
(q, p)	z (Case 3.b)

Figure 4.13: Case 3.b for $p_y q_y = qr$. It holds that $|p_y q_y| \leq 1/3|\Delta_{pqr}|$. If $|pq| \leq 1/3|\Delta_{pqr}|$, $|pr| \leq 1/3|\Delta_{pqr}|$ or $|qr| > 1/6|\Delta_{pqr}|$, then the length of the red cycle is at most $5/3|\Delta_{pqr}|$.

Further, we have

$$|C| \leq 2|pq| + 2|pr| \leq 2|\Delta_{pqr}| - 2|qr|.$$

So, if $|qr| \geq 1/6|\Delta_{pqr}|$, then $|C| \leq 5/3|\Delta_{pqr}|$.

Assumption 5: For the remainder of this proof, we assume that $|qr| < 1/6|\Delta_{pqr}|$.

Then, for all $i < y$ holds that

$$|p_i q_i| \leq |p_y q_y| = |qr| < 1/6|\Delta_{pqr}|.$$

We distinguish whether $p_x = p$ or $p_x = r$ (recall $q_x \notin \{p, r\}$).

Case 3.b.i: $p_x = r$

It holds that $(q, q_x) \in \overrightarrow{E}_x$, otherwise qr is oriented in iteration x (or earlier), which contradicts that qr is oriented in iteration y.

If $|qq_x| < |p_z q_z|$, then, since $(q_x, p), (q, q_x) \in \overrightarrow{E}_x$, pq is oriented in the iteration where qq_x is the current pair, which contradicts that pq is oriented in iteration z.

If $|qq_x| \geq |p_z q_z|$, we have

$$|p_z q_z| \leq |qq_x| \leq |qr| + |rq_x| = |p_y q_y| + |p_x q_x| \leq 2/6|\Delta_{pqr}|.$$

For $pq = p_z q_z$, applying $|pq| \leq 1/3|\Delta_{pqr}|$ to Equation $(*)$ gives us a cycle containing p and q, whose length is at most $5/3|\Delta_{pqr}|$.

If $p_z = p$ and $q_z \notin \{p, q\}$, then $\overrightarrow{K}$ contains the consistently oriented triangle Δ_{pqq_z} of length

$$|\Delta_{pqq_z}| \leq 2|pq| + 2|p_z q_z| \leq |\Delta_{pqr}| + 4/6|\Delta_{pqr}| = 5/3|\Delta_{pqr}|.$$

The same holds for $p_z = q$ and $q_z \notin \{p, q\}$.

Case 3.b.ii: $p_x = p$

It holds that $(q_x, q) \in \overrightarrow{E}_x$, otherwise pq is oriented in iteration x (or earlier), which contradicts that pq is oriented in iteration z.

Let qq_x be oriented in iteration w with $w < x$. At least one of p_w and q_w is in $\{q, q_x\}$. W.l.o.g., let p_w be in $\{q, q_x\}$.

We distinguish whether $p_w q_w = qq_x$ or $q_w \notin \{q, q_x\}$ and p_w is either q_x or q.

Case 3.b.ii.A: $p_w q_w = qq_x$

Then, qr is oriented in iteration w, which contradicts that qr is oriented in iteration $y > w$.

Case 3.b.ii.B: $p_w = q_x$ and $q_w \notin \{q, q_x\}$

We distinguish the cases according to the orientation of $q_w p$.

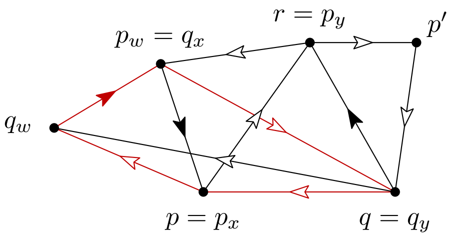
Case 3.b.ii.B.I: $q_w p$ is oriented as (p, q_w)

Consider Figure 4.14. In this case, $\overrightarrow{K}$ contains the cycle

$$C = q \to p \to q_w \to p_w \to q$$

of length

$$\begin{aligned}
|C| &= |qp| + |pq_w| + |q_w p_w| + |p_w q| \\
&\leq |qp| + |pp_w| + |p_w q_w| + |q_w p_w| + |p_w p| + |pq| \\
&\leq 2|pq| + 2|p_w q_w| + 2|p_x q_x| \leq |\Delta_{pqr}| + 4/6|\Delta_{pqr}| = 5/3|\Delta_{pqr}|.
\end{aligned}$$



Edge	Oriented in iteration	
(p,r)	x	(Case 3)
(q,r)	y	(Case 3)
(q,p)	z	(Case 3.b)
(q_x,q)	w	(Case 3.b.ii)

Figure 4.14: Case 3.b.ii.B.I: q_wp is oriented as (p, q_w). For all $i \leq y$, it holds that $|p_iq_i| \leq |\Delta_{pqr}|/6$ (such edges have a filled arrow tip). The length of the red cycle is at most $5/3|\Delta_{pqr}|$.

Case 3.b.ii.B.II: q_wp is oriented as (q_w, p)

If $|q_wp| < |p_zq_z|$, then pq is oriented in the iteration where q_wp is the current pair, which contradicts that pq is oriented in iteration z.

If $|q_wp| \geq |p_zq_z|$, we have

$$|p_zq_z| \leq |q_wp| \leq |q_wp_w| + |p_wp| = |q_wp_w| + |p_xq_x| \leq 2/6|\Delta_{pqr}|.$$

For $pq = p_zq_z$, applying $|pq| \leq 1/3|\Delta_{pqr}|$ to Equation $(*)$ gives us a cycle containing p and q, whose length is at most $5/3|\Delta_{pqr}|$.

If $p_z = p$ and $q_z \notin \{p,q\}$, then $\overrightarrow{K}$ contains the consistently oriented triangle Δ_{pqq_z} of length

$$|\Delta_{pqq_z}| \leq 2|pq| + 2|p_zq_z| \leq |\Delta_{pqr}| + 4/6|\Delta_{pqr}| = 5/3|\Delta_{pqr}|.$$

The same holds for $p_z = q$ and $q_z \notin \{p,q\}$.

Case 3.b.ii.C: $p_w = q$ and $q_w \notin \{q, q_x\}$

It holds that $(p, q_w) \in \overrightarrow{E}_w$, otherwise pq is oriented in iteration w, which contradicts that pq is oriented in iteration $z > w$. Further, it holds that $(r, q_w) \in \overrightarrow{E}_w$, otherwise qr is oriented in iteration w, which contradicts that qr is oriented in iteration y.

Let rq_w be oriented in iteration v with $v < w$. At least one of p_v and q_v is in $\{r, q_w\}$. W.l.o.g., let p_v be in $\{r, q_w\}$.

We distinguish whether $p_vq_v = rq_w$ or $q_v \notin \{r, q_w\}$ and p_v is either q_w or r.

Case 3.b.ii.C.I: $p_vq_v = rq_w$

It holds that $(q, q_w) \in \overrightarrow{E}_v$, otherwise qr is oriented in iteration v.

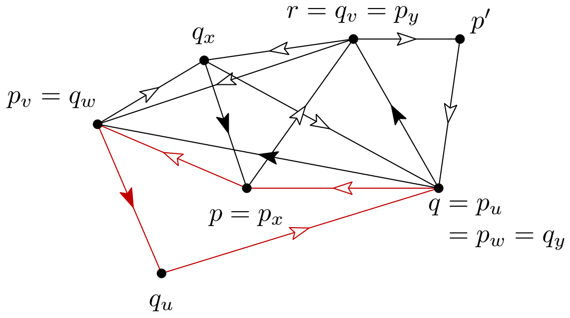
Let q_wq be oriented in iteration u with $u < w$. Since $q_wq = p_wq_w$ and $u < w$, exactly one of $\{p_u, q_u\}$ is in $\{q, q_w\}$, say $p_u \in \{q, q_w\}$. So, $\overrightarrow{K}$ contains the cycle

$$C = q \to p \to q_w \to q_u \to q.$$

For $p_u = q$ and $q_u \notin \{q, q_w\}$ (compare to Figure 4.15), we have

$$\begin{aligned} |C| &= |qp| + |pq_w| + |q_wq_u| + |q_uq| \\ &\leq |qp| + |pq| + |qq_w| + |q_wq| + |qq_u| + |q_up_u| \\ &\leq 2|pq| + 2|p_uq_u| + 2|p_wq_w| \\ &\leq |\Delta_{pqr}| + 4/6|\Delta_{pqr}| = 5/3|\Delta_{pqr}|. \end{aligned}$$

The same holds for $p_u = q_w$ and $q_u \notin \{q, q_w\}$.



Edge	Oriented in iteration	
(p, r)	x	(Case 3)
(q, r)	y	(Case 3)
(q, p)	z	(Case 3.b)
(q_x, q)	w	(Case 3.b.ii)
(p, q_w)	$< w$	(Case 3.b.ii.C)
(r, q_w)	v	(Case 3.b.ii.C)
(q, q_w)	u	(Case 3.b.ii.C.I)

Figure 4.15: Case 3.b.ii.C.I for $p_u = q$ and $q_u \notin \{q, q_w\}$. For all $i \leq y$, it holds that $|p_i q_i| \leq |\Delta_{pqr}|/6$ (such edges have a filled arrow tip). The length of the red cycle is at most $5/3|\Delta_{pqr}|$.

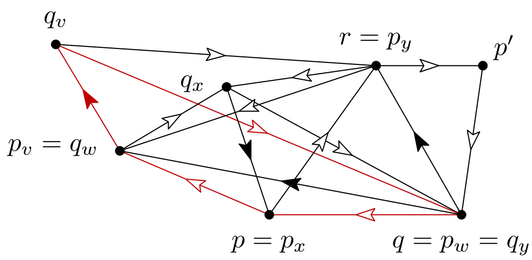
Case 3.b.ii.C.II: $p_v = q_w$ and $q_v \notin \{r, q_w\}$

If $q_v q$ is oriented as (q_v, q) (consider Figure 4.16), then $\overrightarrow{K}$ contains the cycle

$$C = q \to p \to q_w \to q_v \to q$$

of length

$$\begin{aligned} |C| &= |qp| + |pq_w| + |q_w q_v| + |q_v q| \\ &\leq |qp| + |pq| + |qq_w| + |p_v q_v| + |q_v p_v| + |p_v q| \\ &\leq 2|pq| + 2|p_v q_v| + 2|p_w q_w| \\ &\leq |\Delta_{pqr}| + 4/6|\Delta_{pqr}| = 5/3|\Delta_{pqr}|. \end{aligned}$$



Edge	Oriented in iteration	
(p, r)	x	(Case 3)
(q, r)	y	(Case 3)
(q, p)	z	(Case 3.b)
(q_x, q)	w	(Case 3.b.ii)
(p, q_w)	$< w$	(Case 3.b.ii.C)
(r, q_w)	v	(Case 3.b.ii.C)

Figure 4.16: Case 3.b.ii.C.II if $q_v q$ is oriented as (q_v, q). For all $i \leq y$, it holds that $|p_i q_i| \leq |\Delta_{pqr}|/6$ (such edges have a filled arrow tip). The length of the red cycle is at most $5/3|\Delta_{pqr}|$.

If $q_v q$ is oriented as (q, q_v), then $|rp'| + |p'q| \leq |rq_v| + |q_v q|$, because at the beginning of the y-th iteration, qr $(= p_y q_y)$ is oriented such that the triangle $\Delta_{qrp'}$ is oriented consistently (instead of Δ_{qrq_v}). So, $\overrightarrow{K}$ contains the cycle

$$C = q \to p \to r \to p' \to q$$

of length

$$\begin{aligned}
|C| &= |qp| + |pr| + |rp'| + |p'q| \\
&\leq |qp| + |pr| + |rq_v| + |q_v q| \\
&\leq |qp| + |pr| + |rq| + 2|q_v q| \\
&\leq |\Delta_{pqr}| + 2(|q_v p_v| + |p_v q|) = |\Delta_{pqr}| + 2|p_v q_v| + 2|p_w q_w| \\
&\leq |\Delta_{pqr}| + 4/6|\Delta_{pqr}| = 5/3|\Delta_{pqr}|.
\end{aligned}$$

Case 3.b.ii.C.III: $p_v = r$ and $q_v \notin \{r, q_w\}$

It holds that $(q_v, p) \in \overrightarrow{E}_v$, otherwise pr is oriented in iteration v, which contradicts that pr is oriented in iteration x. Further, it holds that $(q_v, q) \in \overrightarrow{E}_v$, otherwise qr is oriented in iteration v, which contradicts that qr is oriented in iteration y.

Let pq_v be oriented in iteration u with $u < v$.

If $p_u q_u = pq_v$, then it holds that

$$|pr| \leq |pq_v| + |q_v r| = |p_u q_u| + |q_v p_v| \leq 2/6|\Delta_{pqr}|.$$

Applying $|pr| \leq 1/3|\Delta_{pqr}|$ to Equation $(*)$ gives us a cycle containing p and q, whose length is at most $5/3|\Delta_{pqr}|$.

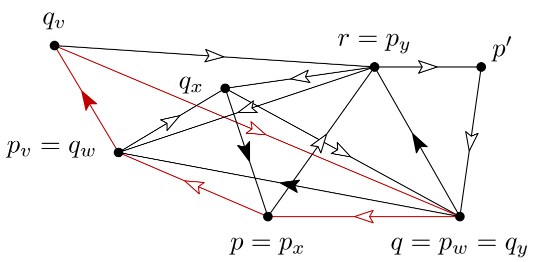
If exactly one of $\{p_u, q_u\}$ is in $\{p, q_v\}$, say $p_u \in \{p, q_v\}$, then $\overrightarrow{K}$ contains the cycle

$$C = q \to p \to q_u \to q_v \to q.$$

For $p_u = p$ and $q_u \notin \{p, q_v\}$ (consider Figure 4.17), we have

$$\begin{aligned}
|C| &= |qp| + |pq_u| + |q_u q_v| + |q_v q| \\
&\leq |qp| + |p_u q_u| + |q_u p| + |pr| + |rq_v| + |q_v r| + |rq| \\
&= |\Delta_{pqr}| + 2|p_u q_u| + 2|p_v q_v| \\
&\leq |\Delta_{pqr}| + 4/6|\Delta_{pqr}| = 5/3|\Delta_{pqr}|.
\end{aligned}$$

The same holds for $p_u = q_v$ and $q_u \notin \{p, q_v\}$. □



Edge	Oriented in iteration	
(p, r)	x	(Case 3)
(q, r)	y	(Case 3)
(q, p)	z	(Case 3.b)
(q_x, q)	w	(Case 3.b.ii)
(p, q_w)	$< w$	(Case 3.b.ii.C)
(r, q_w)	v	(Case 3.b.ii.C)
(q_v, p)	u	(Case 3.b.ii.C.III)
(q_v, q)	$< v$	(Case 3.b.ii.C.III)

Figure 4.17: Case 3.b.ii.C.III for $p_u = p$ and $q_u \notin \{p, q_v\}$. For all $i \leq y$, it holds that $|p_i q_i| \leq |\Delta_{pqr}|/6$ (such edges have a filled arrow tip). The length of the red cycle is at most $5/3|\Delta_{pqr}|$.

Lemma 4.7 implies that the returned tournament $\overrightarrow{K}$ has oriented dilation of at most 5/3. It follows from the proof that for any two distinct points p and q in P, there exists a cycle through p and q in $\overrightarrow{K}$, whose length is at most $5/3|\Delta^*(p,q)|$ and that contains at most four edges. This is tight, since any tournament on four points has a pair that is not incident on any cycle of length 3. This completes the proof of Theorem 4.5.

We finish this section by giving an example of a set of four points in the Euclidean plane, such that the tournament computed by Algorithm 4.2 has oriented dilation equal to 5/3.

Consider Figure 4.18. Let p_1, p_2 and p_3 be collinear and $|p_1p_2| = |p_2p_3| = 1$. Let p_2, p_3, and p_4 form an equilateral triangle with unit length. Algorithm 4.2 starts with p_1p_2 and orients $\Delta_{p_1p_2p_3}$ and $\Delta_{p_1p_2p_4}$ consistently. Now, the edge p_3p_4 remains to be oriented. Since p_3p_4 is not neighboring any triangle that can be oriented consistently any more, p_3p_4 is oriented arbitrarily. Let (p_4, p_3) be its orientation. We have

$$\begin{aligned} \mathrm{odil}(p_2,p_4) &= \frac{|p_2p_4|+|p_4p_1|+|p_1p_2|}{|\Delta_{p_2p_3p_4}|} = \frac{2+\sqrt{3}}{3} \approx 1.244, \text{ and} \\ \mathrm{odil}(p_3,p_4) &= \frac{|p_4p_3|+|p_3p_1|+|p_1p_2|+|p_2p_4|}{|\Delta_{p_2p_3p_4}|} = 5/3. \end{aligned}$$

The dilation of any other pair is 1. Thus, the dilation of the returned graph is 5/3.

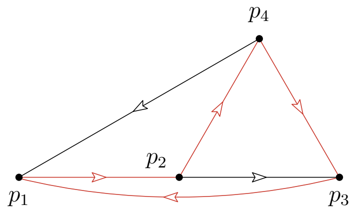


Figure 4.18: Set of four points and its tournament computed by Algorithm 4.2. The edge (p_3, p_1) is curved for clarity though it represents a straight segment from p_3 to p_1. Its dilation is $t = \mathrm{odil}(p_3, p_4) = 5/3$. The shortest closed walk through p_3 and p_4 is shown in red.

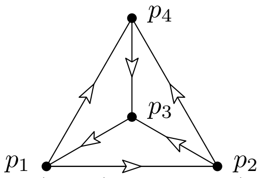
4.4 Bounding the Minimum Dilation

There are point sets for which no oriented 1-spanner exists. Even more, there is a point set of four points in the plane for which no 1.465-spanner exists.

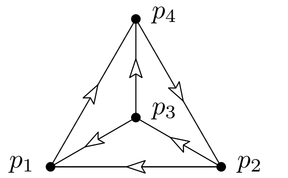
Observation 4.8. *There are point sets in Euclidean space for which no oriented t-spanner exists for $t < 2\sqrt{3} - 2 < 1.465$.*

Proof. We show that $t = 2\sqrt{3} - 2$ holds for every t-tournament on the point set $P = \{p_1, p_2, p_3, p_4\}$ where p_1, p_2 and p_4 form an equilateral triangle centered at p_3.

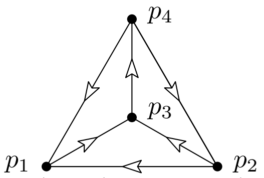
Taking into account mirroring and rotation, Figure 4.19 lists all strongly connected orientations of the complete graph on P. For each tournament, we give the oriented dilation of each pair of points. Every shown graph has dilation $2\sqrt{3}-2$. □



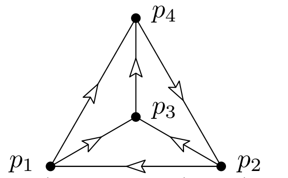
odil	p_1	p_2	p_3
p_4	1	$2\sqrt{3}-2$	1
p_3	1	1	
p_2	1		



odil	p_1	p_2	p_3
p_4	$6\sqrt{3}-9$	1	1
p_3	$2\sqrt{3}-2$	1	
p_2	$6\sqrt{3}-9$		



odil	p_1	p_2	p_3
p_4	1	$2\sqrt{3}-2$	1
p_3	1	1	
p_2	1		



odil	p_1	p_2	p_3
p_4	$6\sqrt{3}-9$	1	1
p_3	$2\sqrt{3}-2$	1	
p_2	$6\sqrt{3}-9$		

Figure 4.19: All tournaments on the point set, where p_1, p_2 and p_4 form an equilateral triangle centered at p_3.

Lemma 4.10 gives a lower bound on the minimum dilation of metric point sets. The following properties of the complete graph on four points and its tournaments are used repeatedly in the proof of Lemma 4.10.

Observation 4.9. *For every undirected complete graph K_4, it holds that*

- *K_4 contains $\binom{4}{3} = 4$ triangles,*
- *every pair of these triangles shares exactly one edge, and*
- *every strongly connected orientation of K_4 contains exactly two consistently oriented triangles.*

Lemma 4.10. *For any metric point set P of size 4 there is a $3/2$-tournament on P. This bound is tight.*

Proof. We prove that the following algorithm computes a tournament $\overrightarrow{K}$ with dilation $\mathrm{odil}(\overrightarrow{K}) \leq 3/2$ for the point set $P = \{p_1, p_2, p_3, p_4\}$:

1. Let $\Delta_{p_1p_2p_3}$ be the shortest and $\Delta_{p_1p_2p_4}$ the second shortest triangle of the four triangles in K_4. Orient $\Delta_{p_1p_2p_3}$ and $\Delta_{p_1p_2p_4}$ consistently. That is always possible (compare to Observation 4.9).
2. Orient the remaining edge p_3p_4 such that the length of the shortest oriented cycle containing p_3 and p_4 is minimized.

We distinguish cases by the orientation of the edge p_3p_4, meaning

- $|C_{\overrightarrow{K}}(p_3, p_4)| = |p_1p_2| + |p_2p_3| + |p_3p_4| + |p_1p_4|$ if
$$|p_1p_3| + |p_2p_4| \leq |p_2p_3| + |p_1p_4|, \text{ or} \tag{4.1}$$
- $|C_{\overrightarrow{K}}(p_3, p_4)| = |p_1p_2| + |p_2p_4| + |p_3p_4| + |p_1p_3|$ if
$$|p_1p_3| + |p_2p_4| > |p_2p_3| + |p_1p_4|. \tag{4.2}$$

We show the first case, the other case can be proven analogously.

Since $\Delta_{p_1p_2p_3}$ and $\Delta_{p_1p_2p_4}$ are the shortest triangles and oriented consistently, the dilation of every pair of points is 1, except the pair p_3p_4. So, we prove

$$\text{odil}(\overrightarrow{K}) = \text{odil}(p_3, p_4) = \frac{|p_1p_2| + |p_2p_3| + |p_3p_4| + |p_1p_4|}{\min(|\Delta_{p_3p_4p_1}|, |\Delta_{p_3p_4p_2}|)} \leq 3/2.$$

Assume $|\Delta_{p_3p_4p_2}| \leq |\Delta_{p_3p_4p_1}|$ otherwise the names of the points belonging to the shortest and second shortest triangle can be swapped. Since $\Delta_{p_1p_2p_3}$ and $\Delta_{p_1p_2p_4}$ are the two shortest triangles, it holds that

$$|p_1p_2| + |p_1p_4| \leq |p_3p_4| + |p_2p_3|, \text{ and} \tag{4.3}$$
$$|p_1p_2| + |p_1p_3| \leq |p_3p_4| + |p_2p_4|. \tag{4.4}$$

Summing up Equations (4.1), (4.3) and (4.4), we achieve

$$\begin{aligned}
& 2|p_1p_3| + 2|p_1p_2| + |p_2p_4| && \leq 2|p_3p_4| + 2|p_2p_3| + |p_2p_4| \\
\Leftrightarrow \quad & 2\left(|p_1p_3| + |p_1p_2| + |p_2p_4| + |p_3p_4|\right) && \leq 4|p_3p_4| + 2\left(|p_2p_3| + |p_2p_4|\right) \\
& && \overset{\Delta\text{-ineq.}}{\leq} 3\left(|p_3p_4| + |p_2p_3| + |p_2p_4|\right) \\
\Leftrightarrow \quad & \text{odil}(p_3, p_4) = \tfrac{|p_3p_4| + |p_1p_3| + |p_1p_2| + |p_2p_4|}{|p_3p_4| + |p_2p_3| + |p_2p_4|} && \leq 3/2.
\end{aligned}$$

For tightness, we show there is a point set P of size 4, such that every tournament on P has dilation 3/2. The following metric gives such a point set: $|p_1p_3| = |p_2p_3| = |p_3p_4| = 1$ and $|p_1p_2| = |p_1p_4| = |p_2p_4| = 2$. Taking into account mirroring and rotation, Figure 4.20 lists all tournaments on P. All of them have dilation 3/2. $\square$

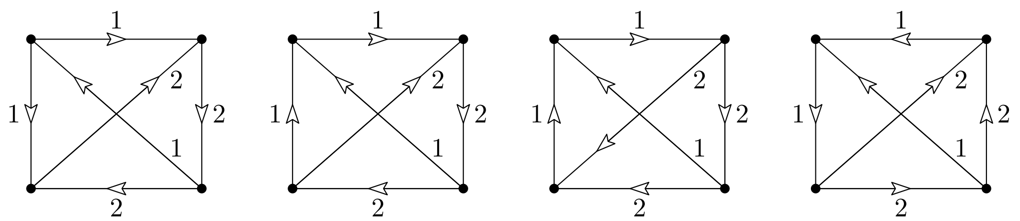


Figure 4.20: A metric point set where every strongly connected orientation of the complete graph on P is a 3/2-tournament.

Observation 4.8 and Lemma 4.10 present lower bounds on the largest possible dilation of any oriented minimum dilation spanner. Our algorithm to compute a 5/3-tournament for any metric point set (Theorem 4.5) offers an upper bound:

Proposition 4.11. *For any metric space* $(P, |\cdot|)$ *the dilation* t *of a minimum dilation spanner on* P *is at most* 5/3.

Chapter 5
One-Dimensional Point Sets

In this chapter, we focus on points in the one-dimensional Euclidean space $\mathbb{R}$. We embed the points on a horizontal line with the minimum point leftmost, and the maximum point rightmost. One-dimensional point sets are indexed in increasing order, i.e., p_1 is the leftmost and p_n the rightmost point.

First, we prove that for spanners on one-dimensional point sets only the dilation of a linear number of candidate pairs needs to be checked.

Lemma 5.1. *Let P be a set of n points in $\mathbb{R}$ and let $\overrightarrow{G}$ be an oriented graph on P. It holds*

$$\operatorname{odil}(\overrightarrow{G}) = \max_{\substack{1\leq i\leq n-1\\ 1\leq j\leq n-2\\ 1\leq k\leq n-3}} (\operatorname{odil}(p_i, p_{i+1}), \operatorname{odil}(p_j, p_{j+2}), \operatorname{odil}(p_k, p_{k+3})).$$

Proof. Let p_ℓ, p_r be any two distinct points in P such that $\ell < r$.

As the cases $r = \ell + 1$, $r = \ell + 2$ and $r = \ell + 3$ are given by the Lemma's statement, the statement is proven for $r < \ell + 4$ or $\ell > n - 3$.

For $r \geq \ell + 4$ and $\ell \leq n - 3$, by induction on $r - \ell$, we show

$$\operatorname{odil}(p_\ell, p_r) \leq \max_{\substack{1\leq j\leq n-2\\ 1\leq k\leq n-3}} (\operatorname{odil}(p_j, p_{j+2}), \operatorname{odil}(p_k, p_{k+3})).$$

Since the union of $C_{\overrightarrow{G}}(p_\ell, p_{\ell+2})$ and $C_{\overrightarrow{G}}(p_{\ell+2}, p_r)$ forms a closed walk containing p_ℓ and p_r, the length of their shortest closed walk is bounded by

$$|C_{\overrightarrow{G}}(p_\ell, p_r)| \leq |C_{\overrightarrow{G}}(p_\ell, p_{\ell+2})| + |C_{\overrightarrow{G}}(p_{\ell+2}, p_r)|.$$

Further, for $r \geq \ell + 4$, the length of the minimum perimeter triangle of p_ℓ and p_r is two times the distance between them, i.e., $|\Delta^*(p_\ell, p_r)| = 2|p_\ell p_r|$. This permits

$$\begin{aligned} \operatorname{odil}(p_\ell, p_r) &= \frac{|C_{\overrightarrow{G}}(p_\ell, p_r)|}{2|p_\ell p_r|} \leq \frac{|C_{\overrightarrow{G}}(p_\ell, p_{\ell+2})| + |C_{\overrightarrow{G}}(p_{\ell+2}, p_r)|}{2|p_\ell p_{\ell+2}| + 2|p_{\ell+2} p_r|} \\ &\leq \max\left(\frac{|C_{\overrightarrow{G}}(p_\ell, p_{\ell+2})|}{|\Delta^*(p_\ell, p_{\ell+2})|}, \frac{|C_{\overrightarrow{G}}(p_{\ell+2}, p_r)|}{|\Delta^*(p_{\ell+2}, p_r)|}\right) \\ &= \max(\operatorname{odil}(p_\ell, p_{\ell+2}), \operatorname{odil}(p_{\ell+2}, p_r)). \end{aligned}$$

Recursively, this bounds the dilation of any tuple p_ℓ, p_r with $r \geq \ell + 4$. $\square$

Lemma 5.1 directly leads to an oriented 1-spanner with $3n-6$ edges.

Corollary 5.2. *For every set P of n points in $\mathbb{R}$, the graph $\overrightarrow{G} = (P, \overrightarrow{E})$ with*

$$\overrightarrow{E} = \{(p_i, p_{i+1}), (p_{j+2}, p_j), (p_{k+3}, p_k) \mid 1 \le i \le n-1, 1 \le j \le n-2, 1 \le k \le n-3\}$$

is an oriented 1*-spanner on P (see Figure 5.1).*

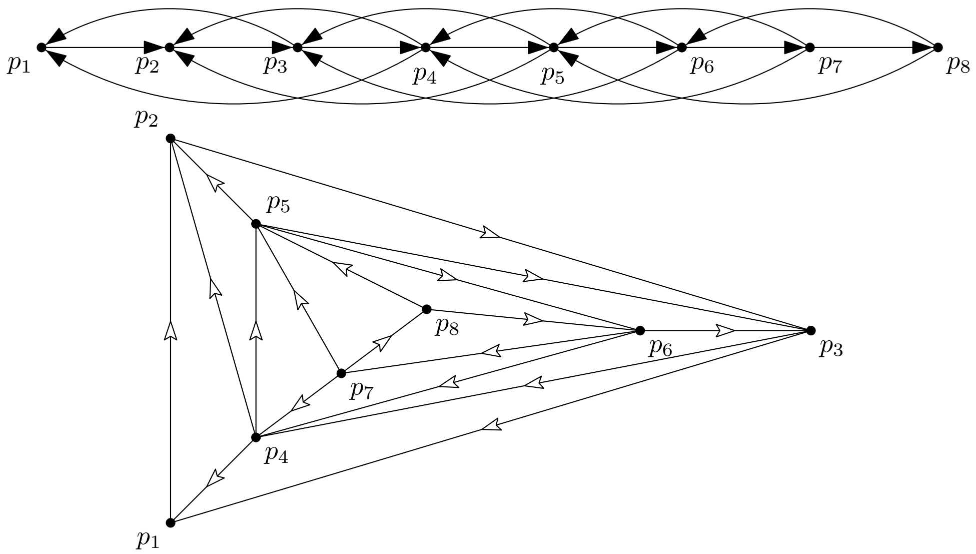


Figure 5.1: Oriented 1-spanner on a point set in $\mathbb{R}$ and its plane embedding in $\mathbb{R}^2$.

In $\mathbb{R}^2$, plane (straight-line) spanners are of particular interest [26]. The natural one-dimensional analogue to plane graphs are one- and two-page book embeddings [11, 35, 45, 89].

A *one-page book embedding* [11, 35, 45] of a graph corresponds to an embedding of the vertices as points on a horizontal line with the edges drawn without crossings as circular arcs above the line; and also below in a *two-page book embedding.* In such a (one- or two-page) book embedding, for consecutive points on the line, we may draw their edge straight on the line. We call a graph *one-page plane* (respectively *two-page plane*) if it has a one-page (resp. two-page) book embedding.

In particular, one-page plane graphs are outerplanar graphs and correspond to plane straight-line graphs if we embed the points on a (slightly) convex arc instead of a line. Two-page plane graphs are a subclass of planar graphs, since any planar graph has a four-page book embedding [89].

As we argue next, the graph given in Corollary 5.2 has no two-page book embedding with the given one-dimensional embedding (and thus also no one-page book embedding). This follows from an even stronger statement:

Observation 5.3. *The graph $\overrightarrow{G} = (P, \overrightarrow{E})$ on $n > 4$ points and*

$$\overrightarrow{E} = \{(p_i, p_{i+1}), (p_{j+2}, p_j), (p_{k+3}, p_k) \mid 1 \le i \le n-1, 1 \le j \le n-2, 1 \le k \le n-3\}$$

has no plane embedding with the first and last point on the outer face (see Figure 5.1).

Proof. Suppose the graph has such an embedding. It has $3n - 6$ edges, but no edge between the first and last point for $n > 4$. Thus, adding this edge preserves planarity, which contradicts that a planar graph has at most $3n - 6$ edges. $\square$

Interestingly, the graph of Corollary 5.2 is planar. We construct a stack triangulation by adding points from left to right. The first three points form a triangle. Then, we inductively add the next point into the triangle formed by the previous three points. See Figure 5.1.

Since Lemma 5.1 directly leads to a 1-spanner with a three-page book embedding (i.e., a crossing-free embedding using a third half-plane) on any point set in $\mathbb{R}$, only one- and two-page plane spanners are interesting to study.

5.1 Page Planarity

A *maximal* one-page plane graph is a one-page plane graph $\overrightarrow{G} = (P, \overrightarrow{E})$ such that for every edge $e \notin \overrightarrow{E}$, the graph $\overrightarrow{G'} = (P, \overrightarrow{E} \cup \{e\})$ is not one-page plane. A maximal two-page plane graph is defined analogously.

A forward edge (p_i, p_{i+1}) between consecutive points is called a *baseline edge* and a backwards edge (p_j, p_i) with $i < j$ a *back edge.* A natural restriction of oriented graphs for one-dimensional point sets is to include a *baseline* and all other edges are back edges. We call such a graph a *one-page-plane-baseline graph*, for short 1-*PPB graph.* Analogously we define 2-*PPB graphs.*

Throughout this thesis, baseline edges are always assumed present in a 1- or 2-PPB graph; hence we define such graphs solely by their back edges.

The pairs that have to be considered to compute the dilation of an oriented spanner on a one-dimensional point set (compare to Lemma 5.1) can be reduced even further for page plane graphs with a baseline:

Lemma 5.4. *Let P be a point set in $\mathbb{R}$ of size n and let $\overrightarrow{G}$ be a* 1- *or* 2-*PPB graph on P. It holds that*

$$\operatorname{odil}(\overrightarrow{G}) = \max_{\substack{1 \le i \le n-2 \\ 1 \le j \le n-3}} (\operatorname{odil}(p_i, p_{i+2}), \operatorname{odil}(p_j, p_{j+3})).$$

Proof. We show $\operatorname{odil}(p_j, p_{j+1}) \le \max(\operatorname{odil}(p_j, p_{j+2}), \operatorname{odil}(p_{j-1}, p_{j+1}))$. Combining this with Lemma 5.1 completes the proof of this lemma.

The third point of the minimum perimeter triangle $\Delta^*(p_j, p_{j+1})$ is p_{j-1} or p_{j+2}. For graphs with baseline, it holds that

$$|C_{\overrightarrow{G}}(p_j, p_{j+1})| \le |C_{\overrightarrow{G}}(p_j, p_{j+2})| \text{ and } |C_{\overrightarrow{G}}(p_j, p_{j+1})| \le |C_{\overrightarrow{G}}(p_{j-1}, p_{j+1})|.$$

If $\Delta^*(p_j, p_{j+1})$ contains p_{j+2}, the dilation is bounded by

$$\operatorname{odil}(p_j, p_{j+1}) = \frac{|C_{\overrightarrow{G}}(p_j, p_{j+1})|}{|\Delta^*(p_j, p_{j+1})|} = \frac{|C_{\overrightarrow{G}}(p_j, p_{j+1})|}{2|p_j p_{j+2}|} \le \frac{|C_{\overrightarrow{G}}(p_j, p_{j+2})|}{|\Delta^*(p_j, p_{j+2})|} = \operatorname{odil}(p_j, p_{j+2}).$$

Otherwise, we can show analogously that $\operatorname{odil}(p_k, p_{k+1}) \le \operatorname{odil}(p_{k-1}, p_{k+1})$. $\square$

Lemma 5.4 holds for every page plane graph, even if it is not maximal.

Further, we offer additional structural insights on the shape of the shortest closed walk containing the worst-case pair of a 1- or 2-PPB graph.

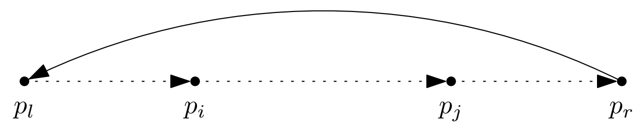
Lemma 5.5. *Let P be a point set in $\mathbb{R}$ and $\overrightarrow{G} = (P, \overrightarrow{E})$ a 1- or 2-PPB graph on P. Let p_i, p_j be two distinct points in P with $j \in \{i+2, i+3\}$. Let p_ℓ with $\ell \leq i$ be leftmost point in $C_{\overrightarrow{G}}(p_i, p_j)$ and p_r with $j \leq r$ be the rightmost point in $C_{\overrightarrow{G}}(p_i, p_j)$. The closed walk $C_{\overrightarrow{G}}(p_i, p_j)$ consists of the baseline edges from p_ℓ to p_r, i.e., the edges (p_m, p_{m+1}) with $\ell \leq m \leq r-1$, and a path Π from p_r to p_ℓ with either*

(a) $\Pi = (p_r, p_\ell)$ *(see Figure 5.2a),*

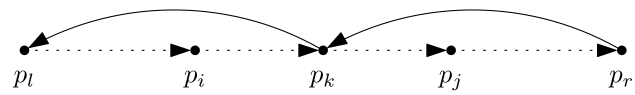
(b) $\Pi = (p_r, p_k), (p_k, p_\ell)$ *with $i < k < j$ (see Figure 5.2b), or*

(c) $\Pi = (p_r, p_{i+1}), (p_{i+1}, p_{i+2}), (p_{i+2}, p_\ell)$ *(see Figure 5.2c).*

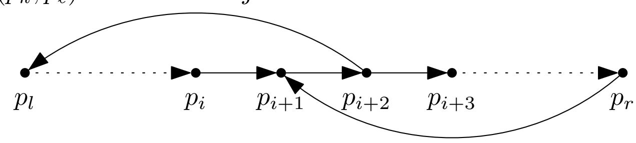
Note that, due to planarity, Case c arises only if $\overrightarrow{G}$ is a 2-PPB graph and $j = i+3$. Further, if $\overrightarrow{G}$ is maximal, for Case a holds either $\Pi = (p_r, p_i)$ or $\Pi = (p_j, p_\ell)$.



(a) $\Pi = (p_r, p_\ell)$



(b) $\Pi = (p_r, p_k), (p_k, p_\ell)$ with $i < k < j$



(c) $\Pi = (p_r, p_{i+1}), (p_{i+1}, p_{i+2}), (p_{i+2}, p_\ell)$

Figure 5.2: $C_{\overrightarrow{G}}(p_i, p_j)$ consists of the baseline from p_ℓ to p_r and a path Π from p_r to p_ℓ with $j \in \{i+2, i+3\}$ and $\ell \leq i < j \leq r$.

Proof. By definition, every 1- or 2-PPB graph includes a baseline. Note that a shortest closed walk through p_i and p_j is simply the union of a shortest path from p_ℓ to p_r and a shortest path from p_r to p_ℓ. As the baseline provides a shortest path from p_ℓ to p_r, we now construct a shortest path Π from p_r to p_ℓ to complete $C_{\overrightarrow{G}}(p_i, p_j)$.

If $(p_r, p_\ell) \in \overrightarrow{E}$, it holds $\Pi = (p_r, p_\ell)$, thus Case a.

For $(p_r, p_\ell) \notin \overrightarrow{E}$, we consider the first edge of Π. Since p_r is the rightmost point in $C_{\overrightarrow{G}}(p_i, p_j)$, this edge must be a back edge (p_r, p_k) and therefore $k < r$. Note that $k > i$, otherwise p_ℓ is not the leftmost point in a shortest closed walk through p_i and p_j. Furthermore, $k < j$, otherwise p_r is not the rightmost point.

For $j = i+2$ (and therefore $k = i+1$), if (p_{i+1}, p_{i+2}) is the second edge in Π, then p_r is not rightmost point in a shortest closed walk through p_i and p_j.

Therefore, a back edge $(p_k, p_{k'})$ with $k' < i$ is the second edge in Π (see Figure 5.2b). Since p_ℓ is the leftmost point in $C_{\overrightarrow{G}}(p_i, p_j)$, this implies $k' = \ell$ and therefore $\Pi = (p_r, p_{i+1}), (p_{i+1}, p_\ell)$, thus Case b.

If $j = i+3$ and $k = i+2$, analogous, the second edge in Π is a back edge $(p_k, p_{k'})$ with $k' \leq i$. Again, it holds that $\Pi = (p_r, p_{i+2}), (p_{i+2}, p_\ell)$, thus Case b.

For $j = i+3$ and $k = i+1$, if the second edge in Π is a back edge $(p_k, p_{k'})$ with $k' \leq i$, it holds $\Pi = (p_r, p_{i+1}), (p_{i+1}, p_\ell)$, thus Case b. Otherwise, if the second edge is (p_{i+1}, p_{i+2}), it holds $\Pi = (p_r, p_{i+1}), (p_{i+1}, p_{i+2}), (p_{i+2}, p_\ell)$, thus Case c. $\square$

Combining Lemmas 5.4 and 5.5, computing the oriented dilation of a 1- or 2-PPB spanner is possible in $\mathcal{O}(n)$ time, if the shortest incoming and outgoing back edge of any point is accessible in constant time.

5.2 One-Page Plane Spanners

As noted above, one-page plane graphs correspond to plane straight-line graphs if we interpret the point set as being convex. Thus, they are not only of interest in their own right but also aid us in constructing oriented plane spanners in $\mathbb{R}^2$.

Lemma 5.6 shows that a 1-PPB graph has at most the same dilation as an oriented graph with the same edge set but another orientation. For the remainder of this chapter, we focus exclusively on 1-PPB graphs rather than on general one-page plane graphs. This restriction is justified by Lemma 5.6, which implies that it does not affect the minimum attainable dilation. Consequently, the construction of one-page plane spanners is simplified. In particular, computing a minimum dilation 1-PPB graph yields a minimum one-page plane spanner (see Section 5.2.3).

Lemma 5.6. *Let P be a point set in $\mathbb{R}$ of size n and let $\overrightarrow{G} = (P, \overrightarrow{E})$ be a one-page plane graph on P. Let $\overrightarrow{G'} = (P, \overrightarrow{E'})$ be a 1-PPB graph, where E' contains the baseline edges and for every edge in $\overrightarrow{E}$ the same edge but oriented according to the conditions of a 1-PPB graph. It holds that* $\operatorname{odil}(\overrightarrow{G'}) \leq \operatorname{odil}(\overrightarrow{G})$.

Proof. W.l.o.g., we assume that for all i with $1 \leq i \leq n-1$ either (p_i, p_{i+1}) or (p_{i+1}, p_i) is in $\overrightarrow{E}$, since adding these would only decrease the dilation while $\overrightarrow{E'}$ contains the baseline anyway. Let $G^u = (P, E^u)$ be the underlying undirected graph of $\overrightarrow{G}$. Let p and q be two distinct points in P and let $C_{\overrightarrow{G}}(p, q)$ be their shortest closed walk in $\overrightarrow{G}$. Let p_ℓ be the leftmost and p_r be the rightmost point of $C_{\overrightarrow{G}}(p, q)$, respectively. Consider the edges in G^u corresponding to $C_{\overrightarrow{G}}(p, q)$, that is $E^u_C = \{p_i p_j \in E^u \mid (p_i, p_j) \in C_{\overrightarrow{G}}(p, q) \text{ or } (p_j, p_i) \in C_{\overrightarrow{G}}(p, q)\}$.

We define the *inclusion maximal edges* in this set as those edges, which are not covered by longer edges, that is $\operatorname{imax}(E^u_C) = \{p_i p_j \in E^u_C \mid p_{i'}, p_{j'} \in P \text{ with } i' \leq i < j \leq j' : \; p_{i'} p_{j'} \notin E^u_C\}$. For this set, we can state that

i) For all $p_i p_j \in \operatorname{imax}(E^u_C)$ we have $j > i+1$, because an edge $p_i p_{i+1}$ cannot be inclusion maximal, since the interval between p_i and p_{i+1} has to be covered by at least two edges to allow a connected graph.

ii) The edges in $\operatorname{imax}(E^u_C)$ partition an interval $[p_\ell, p_r]$, as $\overrightarrow{G}$ is one-page plane and thus $C_{\overrightarrow{G}}(p, q)$ would be disconnected otherwise.

By using only the inclusion maximal and baseline edges, it follows that the orientation of E_C^u given by $\overrightarrow{G'}$ includes a closed walk of length $2|p_\ell p_r|$ and thus

$$|C_{\overrightarrow{G'}}(p,q)| \leq 2|p_\ell p_r| \leq |C_{\overrightarrow{G}}(p,q)|.$$

Therefore, the oriented dilation of p and q in the 1-PPB $\overrightarrow{G'}$ is smaller or equals their oriented dilation in $\overrightarrow{G}$. $\square$

The restriction, given Lemma 5.4, on pairs to check for computing the oriented dilation can be tightened for 1-PPB graphs:

Lemma 5.7. *Let P be a point set in $\mathbb{R}$ of size n and let $\overrightarrow{G} = (P, \overrightarrow{E})$ be a 1-PPB graph on P. It holds* $\operatorname{odil}(\overrightarrow{G}) = \max_{1 \leq i \leq n-2}(\operatorname{odil}(p_i, p_{i+2}))$.

Proof. Let $\overrightarrow{G}$ be a 1-PPB spanner. For any point $p_j \in P$, we will show that

$$\operatorname{odil}(p_j, p_{j+3}) \leq \max(\operatorname{odil}(p_j, p_{j+2}), \operatorname{odil}(p_{j+1}, p_{j+3})).$$

By Lemma 5.4, this completes the proof of this lemma.

For graphs with baseline, we have $|C_{\overrightarrow{G}}(p_{j+1}, p_{j+3})| \leq |C_{\overrightarrow{G}}(p_j, p_{j+3})|$.

If $|C_{\overrightarrow{G}}(p_j, p_{j+3})| = |C_{\overrightarrow{G}}(p_{j+1}, p_{j+3})|$, it holds that

$$\begin{aligned}\operatorname{odil}(p_j, p_{j+3}) = \frac{|C_{\overrightarrow{G}}(p_j, p_{j+3})|}{|\Delta^*(p_j, p_{j+3})|} &= \frac{|C_{\overrightarrow{G}}(p_{j+1}, p_{j+3})|}{2|p_j p_{j+3}|} \\ &\leq \frac{|C_{\overrightarrow{G}}(p_{j+1}, p_{j+3})|}{2|p_{j+1} p_{j+3}|} = \operatorname{odil}(p_{j+1}, p_{j+3}).\end{aligned}$$

If $|C_{\overrightarrow{G}}(p_j, p_{j+3})| > |C_{\overrightarrow{G}}(p_{j+1}, p_{j+3})|$, the leftmost point of $C_{\overrightarrow{G}}(p_{j+1}, p_{j+3})$ is p_{j+1}. This implies $|C_{\overrightarrow{G}}(p_j, p_{j+2})| = |C_{\overrightarrow{G}}(p_j, p_{j+3})|$ and therefore

$$\operatorname{odil}(p_j, p_{j+3}) = \frac{|C_{\overrightarrow{G}}(p_j, p_{j+3})|}{|\Delta^*(p_j, p_{j+3})|} = \frac{|C_{\overrightarrow{G}}(p_j, p_{j+2})|}{2|p_j p_{j+3}|} \leq \frac{|C_{\overrightarrow{G}}(p_j, p_{j+2})|}{2|p_j p_{j+2}|} = \operatorname{odil}(p_j, p_{j+2}).$$

The maximum of $\operatorname{odil}(p_{j+1}, p_{j+3})$ and $\operatorname{odil}(p_j, p_{j+2})$ upper bounds both cases. $\square$

Lemma 5.7 holds for every 1-PPB graph, even if it is not maximal.

5.2.1 Bounding the Dilation of One-Page Plane Graphs

Due to planarity, every one-page plane graph $\overrightarrow{G}$ on a point set P with $|P| > 3$ contains a tuple $p_i, p_{i+2} \in P$ where $C_{\overrightarrow{G}}(p_i, p_{i+2}) \neq \Delta^*(p_i, p_{i+2})$. So, there is no one-page plane 1-spanner on any point set P with $|P| > 3$.

However, there are point sets with a one-page plane $(1+\varepsilon)$-spanner. Let $\overrightarrow{G} = (P, \overrightarrow{E})$ be a 1-PPB graph on five points with the distances $|p_2 p_3| = |p_3 p_4| = 1$ and $|p_1 p_2| = |p_4 p_5| = \varepsilon$ for an arbitrary small $\varepsilon > 0$ and with the back edges (p_5, p_3) and (p_3, p_1) (see Figure 5.3). It holds that $\operatorname{odil}(\overrightarrow{G}) = \operatorname{odil}(p_2, p_4) = \frac{2+2\varepsilon}{2} = 1 + \varepsilon$.

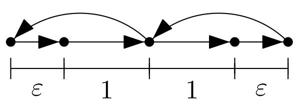


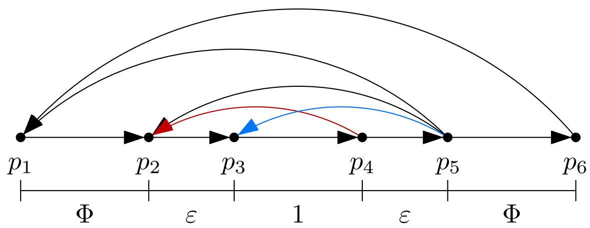
Figure 5.3: One-page plane $(1+\varepsilon)$-spanner.

In the following, we study the worst-case (i.e., supremum of) the minimum dilation taken over all point sets. We present an instance of six points, which gives a lower bound on the dilation of minimum one-page plane spanners of 2.618.

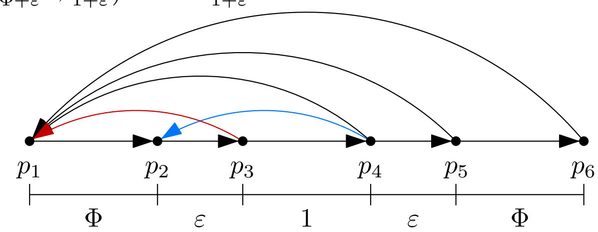
Observation 5.8. *There are point sets in $\mathbb{R}$ where no one-page plane t-spanner exists for $t < 1 + \Phi - \delta$, where Φ is the golden ratio $\Phi = 1.618033\ldots$ and $\delta > 0$ is an arbitrarily small value.*

Proof. Given $\delta > 0$, we define a point set P such that for every 1-PPB t-spanner on P holds $t \geq 1 + \Phi - \delta$. Let P be a set of six points in $\mathbb{R}$ with the following pairwise distances: $|p_1p_2| = |p_5p_6| = \Phi$, $|p_2p_3| = |p_4p_5| = \varepsilon$ and $|p_3p_4| = 1$ for a small $\varepsilon > 0$ (depending on δ).

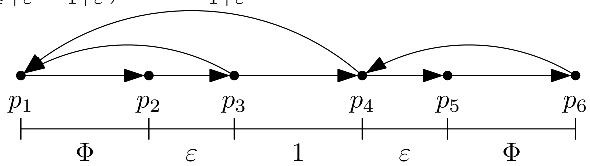
It suffices to look at all maximal 1-PPB graphs on P. We list those and, since P is symmetric, prune out every graph, which has an axial symmetric copy. Further, we ignore the graph that contains (p_5, p_1),(p_5, p_3) and (p_3, p_1) as it is trivially not optimal. Computing the dilation of every remaining graph (listed in Figure 5.4), we conclude that a minimum 1-PPB spanner on P has dilation $t = 1 + \frac{\Phi+\varepsilon}{1+\varepsilon} < 1 + \Phi$. For an arbitrary small $\varepsilon > 0$, this value approaches the bound arbitrarily closely. $\square$



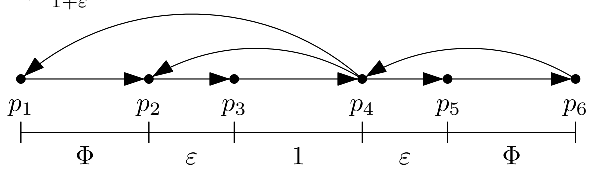
(a) $t = 1 + \max\left(\frac{1+\Phi+\varepsilon}{\Phi+\varepsilon}, \frac{\varepsilon}{1+\varepsilon}\right) \geq 1 + \frac{\Phi+\varepsilon}{1+\varepsilon}$



(b) $t = 1 + \max\left(\frac{1+\Phi+\varepsilon}{\Phi+\varepsilon}, \frac{\Phi+\varepsilon}{1+\varepsilon}\right) \geq 1 + \frac{\Phi+\varepsilon}{1+\varepsilon}$



(c) $t = 1 + \frac{2\Phi+\varepsilon}{1+\varepsilon} \geq 1 + \frac{\Phi+\varepsilon}{1+\varepsilon}$



(d) $t = 1 + \frac{\Phi+\varepsilon}{1+\varepsilon} < 1 + \Phi$

Figure 5.4: Remaining candidates for a minimum 1-PPB spanner on the point set of size 6 with distances $|p_1p_2| = |p_5p_6| = \Phi$, $|p_2p_3| = |p_4p_5| = \varepsilon$, $|p_3p_4| = 1$. Graphs with the same dilation t are drawn in one subfigure with different colored edges.

5.2.2 Constant Dilation One-Page Plane Spanners

A first approach to construct a one-page plane oriented spanner is orienting the greedy triangulation. We show that this leads to a maximal 1-PPB 5-spanner.

In Chapter 6, we analyze plane spanners on points in $\mathbb{R}^2$. The proof, that an orientation of the greedy triangulation yields a constant dilation spanner on convex point sets, adapts and builds upon our proof for one-page plane graphs.

Algorithm. Algorithm 5.1 computes a maximal 1-PPB graph by starting with the baseline and greedily adding back edges sorted increasing by length while preserving planarity. The first added edge is the shortest edge between two points with exactly one point in between. Imagine we delete the point that was in-between. Then again, we add the shortest edge with exactly one point in between, and so on, until only two points are left. This deletion is implemented by a list L, which initially contains the entire point set. We delete a point p from L when we add an edge that covers p. In order to quickly compute the shortest edge that covers p, its length is stored in a priority queue for each point p in L. The resulting graph is an orientation of the greedy triangulation.

Algorithm 5.1 Greedy One-Page Plane 5-Spanner.

Require: Point set $P = \{p_1, \dots, p_n\}$ in $\mathbb{R}$ (numbered from left to right)
Ensure: One-page plane 5-spanner on P

$\overrightarrow{E} = \{(p_i, p_{i+1}) \mid 1 \leq i \leq n-1\}$ *// start with the baseline*
$L =$ doubly linked list with the elements $p_1, \dots, p_n$
$T =$ empty priority queue – T stores tuples (p, v) where p is a pointer to a point in L and the value $v = L(p)$.`next` $- L(p)$.`prev` (distance of p's neighbors in L, which is the length of the edge which covers p)
for $i = 2$ **to** $n-1$ **do**
  add $(p_i, |p_{i-1}p_{i+1}|)$ to T *// initially all edges with 1 point in between*
while $T \neq \emptyset$ **do**
  $p = T$.`extractMin()`
  $p_\ell = L(p)$.`prev`
  $p_r = L(p)$.`next`
  $\overrightarrow{E} = \overrightarrow{E} \cup \{(p_r, p_\ell)\}$
  // p is covered by (p_r, p_ℓ), which is the current shortest edge
  delete p in L and T
  update the values in T of p_ℓ and p_r
return $\overrightarrow{G} = (P, \overrightarrow{E})$

In the proof of Theorem 5.9, we use the concept of a *blocker*.

Definition (blocker)**.** *Let $\overrightarrow{E}$ be the greedily computed edge set. Since the resulting graph $\overrightarrow{G} = (P, \overrightarrow{E})$ is maximal, $(p_j, p_i) \notin \overrightarrow{E}$ for $i + 2 \leq j$ implies that there is a shorter edge in $\overrightarrow{E}$ which intersects (p_j, p_i). (The greedy algorithm added this edge first and discarded (p_j, p_i) in a later iteration.) For the shortest edge intersecting (p_j, p_i), we say it* blocks *(p_j, p_i). For the* blocker *(p_k, p_m) holds either $k > j$ and $i < m < j$ or $m < i$ and $i < k < j$.*

Theorem 5.9. *Given a set P of n points in $\mathbb{R}$, a one-page plane 5-spanner on P can be constructed in $\mathcal{O}(n \log n)$ time.*

Proof. Let $\overrightarrow{G} = (P, \overrightarrow{E})$ be the maximal 1-PPB spanner constructed by Algorithm 5.1. For any pair of points $p_i, p_{i+2} \in P$, we show $\operatorname{odil}(p_i, p_{i+2}) \leq 5$. Due to Lemma 5.7, this guarantees that $\overrightarrow{G}$ is a 5-spanner.

If $(p_{i+2}, p_i) \in \overrightarrow{E}$, it holds that $\operatorname{odil}(p_i, p_{i+2}) = 1$.

For $(p_{i+2}, p_i) \notin \overrightarrow{E}$, we assume that (p_{i+2}, p_i) is blocked by an edge incident to a point right of p_{i+2}. The case that (p_{i+2}, p_i) is blocked from the left is analogous.

Let (b_1, p_{i+1}) be the edge that blocks (p_{i+2}, p_i) (see Figure 5.5). Due to maximality, b_1 is the leftmost point with a back edge to p_{i+1}. Since (b_1, p_{i+1}) blocks (p_{i+2}, p_i), it holds that

$$|p_{i+1}b_1| \leq |p_i p_{i+2}|. \tag{5.1}$$

In the following, we determine the leftmost and the rightmost point of $C_{\overrightarrow{G}}(p_i, p_{i+2})$ and their distances to p_i and p_{i+1}.

Let $b_1, \ldots, b_j$ be the points with a back edge to p_{i+1} ordered from left to right (see Figure 5.5). So, b_j is the rightmost point with a back edge to p_{i+1}.

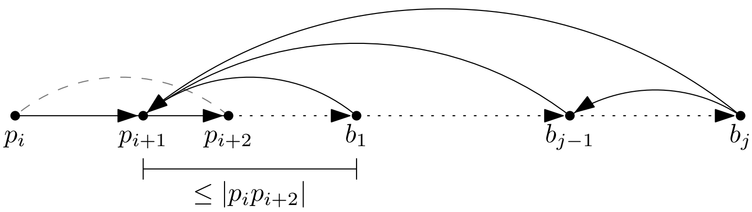


Figure 5.5: Let $b_1, \ldots, b_j$ be the points with a back edge to p_{i+1}. Note that there may be points between b_j and b_{j-1}, but then maximality implies $(b_j, b_{j-1}) \in \overrightarrow{E}$.

For each $2 \leq k \leq j$, the edge (b_k, p_{i+1}) blocks (b_{k-1}, p_i) (see Figure 5.6). Therefore, it holds that $|p_{i+1}b_k| \leq |p_i b_{k-1}|$, which bounds their distance by

$$|b_{k-1}b_k| \leq |p_i p_{i+1}|. \tag{5.2}$$

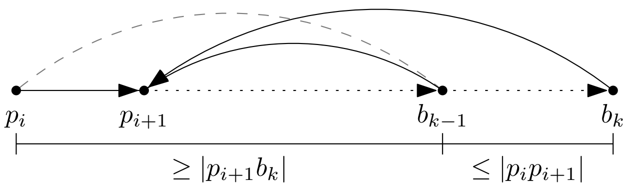


Figure 5.6: (b_k, p_{i+1}) blocks (b_{k-1}, p_i) for $2 \leq k \leq j$.

Further, for $2 \leq k \leq j-1$, the greedy algorithm added (b_k, p_{i+1}) to $\overrightarrow{E}$ and discarded (b_{k+1}, b_{k-1}) (compare to Figure 5.7). Thus, their distance is bounded by

$$|p_{i+1}b_k| \leq |b_{k-1}b_{k+1}|. \tag{5.3}$$

Now, we use these equations to bound the length of the shortest closed walk in $\overrightarrow{G}$ containing p_i and p_{i+2}. Let p_ℓ be the leftmost and p_r the rightmost point of $C_{\overrightarrow{G}}(p_i, p_{i+2})$. We do a case distinction on the position of p_ℓ.

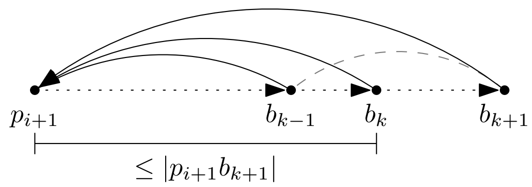


Figure 5.7: (b_k, p_{i+1}) blocks (b_{k+1}, b_{k-1}) for $2 \leq k \leq j-1$.

Case 1: $\ell = i$
Due to maximality, it holds that $r = j$ (see Figure 5.8). Therefore, we have

$$|C_{\overrightarrow{G}}(p_i, p_{i+2})| = 2|p_ib_j| = 2(|p_ip_{i+1}| + |p_{i+1}b_{j-1}| + |b_{j-1}b_j|).$$

If $j \geq 3$, using Equations (5.2) and (5.3), gives us

$$|p_{i+1}b_{j-1}| \leq |b_{j-2}b_j| = |b_{j-2}b_{j-1}| + |b_{j-1}b_j| \leq 2|p_ip_{i+1}| \leq 2|p_ip_{i+2}|.$$

Otherwise, it holds that $j < 3$. If $j = 2$, we apply Equation (5.2) to bound $|b_1b_2| \leq |p_ip_{i+1}|$. So, for $1 < j < 3$ and $b_0 = p_{i+1}$, we also have

$$|p_{i+1}b_{j-1}| \leq |p_{i+1}b_1| + |p_ip_{i+1}| \overset{\text{Eq. (5.1)}}{\leq} 2|p_ip_{i+2}|.$$

This provides the dilation

$$\begin{aligned}\operatorname{odil}(p_i, p_{i+2}) = \frac{|C_{\overrightarrow{G}}(p_i, p_{i+2})|}{|\Delta^*(p_i, p_{i+2})|} = \frac{2|p_ib_j|}{2|p_ip_{i+2}|} &\leq \frac{|p_ip_{i+1}| + 2|p_ip_{i+2}| + |b_{j-1}b_j|}{|p_ip_{i+2}|} \\ &\overset{\text{Eq. (5.2)}}{\leq} \frac{2|p_ip_{i+2}| + 2|p_ip_{i+1}|}{|p_ip_{i+2}|} \leq 4.\end{aligned}$$

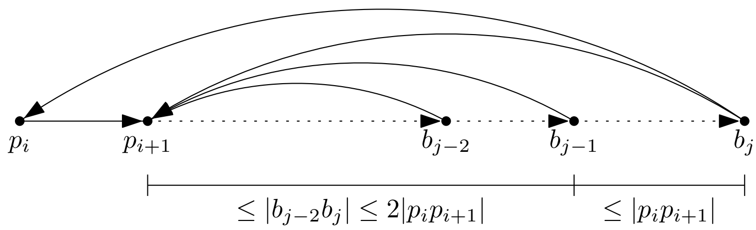


Figure 5.8: Case 1: If $\ell = i$, then $r = j$.

Case 2: $\ell < i$
The maximality of $\overrightarrow{G}$ implies $(p_{i+1}, p_\ell) \in C_{\overrightarrow{G}}(p_i, p_{i+2})$. In this case, since b_1 is the leftmost point with a back edge to p_{i+1}, it holds that $p_r = b_1$ (see Figure 5.9). By definition, (p_{i+1}, p_ℓ) is the blocker of (b_j, p_i). Thus, we have

$$|p_\ell p_{i+1}| \leq |p_ib_j| \leq 4|p_ip_{i+2}|, \tag{5.4}$$

where the last step follows from Case 1. Therefore, the dilation is

$$\operatorname{odil}(p_i, p_{i+2}) = \frac{|C_{\overrightarrow{G}}(p_i, p_{i+2})|}{|\Delta^*(p_i, p_{i+2})|} = \frac{2|p_\ell b_1|}{2|p_i p_{i+2}|} = \frac{|p_\ell p_{i+1}| + |p_{i+1} b_1|}{|p_i p_{i+2}|}$$
$$\overset{\text{Eq. (5.1), (5.4)}}{\leq} \frac{|p_i p_{i+1}| + 4|p_i p_{i+2}|}{|p_i p_{i+2}|} \leq 5.$$

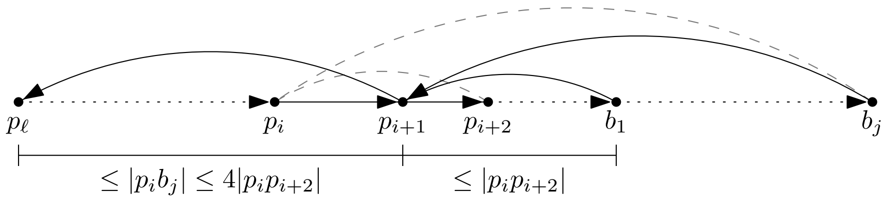


Figure 5.9: Case 2: If $\ell < i$, then $r = 1$. Note that (p_{i+1}, p_ℓ) blocks (b_j, p_i).

By maintaining the points in a doubly linked list and the relevant distances in a priority queue the runtime of Algorithm 5.1 is $\mathcal{O}(n \log n)$. $\square$

Figure 5.10 illustrates a greedily constructed spanner $\overrightarrow{G}$ with oriented dilation $t = \frac{5-7\varepsilon}{1+\varepsilon}$ for a small $\varepsilon > 0$. This configuration is ε-close to the proven upper bound of 5. Furthermore, the figure shows a t'-spanner on the same point set with $t' = \frac{2-2\varepsilon}{1+\varepsilon} < t$. Thus, Algorithm 5.1 does not necessarily return a minimum spanner.

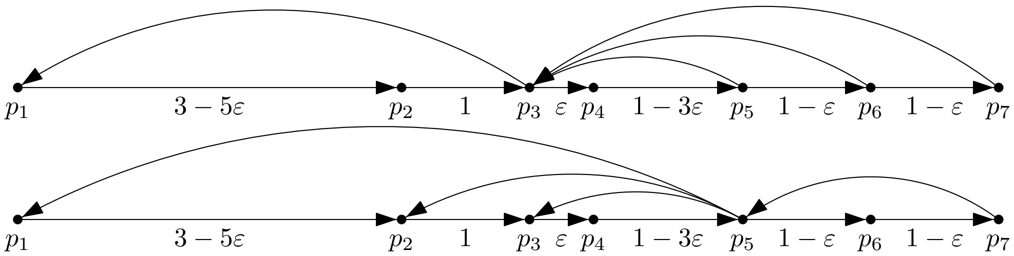


Figure 5.10: Above: Spanner constructed by Algorithm 5.1 with dilation $t = \frac{5-7\varepsilon}{1+\varepsilon}$. Below: Spanner on the same point set with dilation $t' = \frac{2-2\varepsilon}{1+\varepsilon} < t$. For both, the worst-case pair is $p_2 p_4$.

5.2.3 Minimum One-Page Plane Spanners

Next, we focus on minimum dilation one-page plane spanners. Due to Lemma 5.6, it suffices to look at maximal 1-PPB graphs to construct minimum one-page plane spanners in general. Given a point set P in $\mathbb{R}$, the following dynamic program computes a maximal 1-PPB spanner $\overrightarrow{G} = (P, \overrightarrow{E})$ with minimum oriented dilation.

Idea. Due to one-page planarity, a point set $\{p_\ell, \ldots, p_r\}$ can be separated into two (almost) independent subproblems, $\{p_\ell, \ldots, p_k\}$ and $\{p_k, \ldots, p_r\}$, by adding the edges (p_r, p_k) and (p_k, p_ℓ) to a split point p_k between p_ℓ and p_r (consider Figure 5.11a). Due to Lemma 5.7, the dilation t of the minimum spanner on $\{p_\ell, \ldots, p_r\}$

is $t = \max(t', t'', \mathrm{odil}(p_{k-1}, p_{k+1}))$, where t' is the minimum dilation for $\{p_\ell, \dots, p_k\}$ and t'' is the minimum dilation for $\{p_k, \dots, p_r\}$. We test all candidates for p_k to minimize t. Computing $\mathrm{odil}(p_{k-1}, p_{k+1})$ requires the length of $C_{\overrightarrow{G}}(p_{k-1}, p_{k+1})$ which is, since $(p_r, p_k), (p_k, p_\ell) \in \overrightarrow{E}$, the union of $C_{\overrightarrow{G}}(p_k, p_{k+1})$ and $C_{\overrightarrow{G}}(p_{k-1}, p_k)$. By the construction holds $|C_{\overrightarrow{G}}(p_{k-1}, p_k)| = 2|p_{k_r} p_k|$, where p_{k_r} is the rightmost point with $(p_k, p_{k_r}) \in \overrightarrow{E}$, and analogously, we have $|C_{\overrightarrow{G}}(p_k, p_{k+1})| = 2|p_k p_{k_\ell}|$, where p_{k_ℓ} is the leftmost point with $(p_{k_\ell}, p_k) \in \overrightarrow{E}$. To include those two values in the dynamic program, we consider the subproblem of computing $\mathrm{odil}(\ell, \ell', r', r)$, which denotes the oriented dilation of the subgraph on $\{p_\ell, \dots, p_r\}$ under the assumption that $(p_{\ell'}, p_\ell) \in C_{\overrightarrow{G}}(p_\ell, p_{l+1})$ and $(p_r, p_{r'}) \in C_{\overrightarrow{G}}(p_{r-1}, p_r)$.

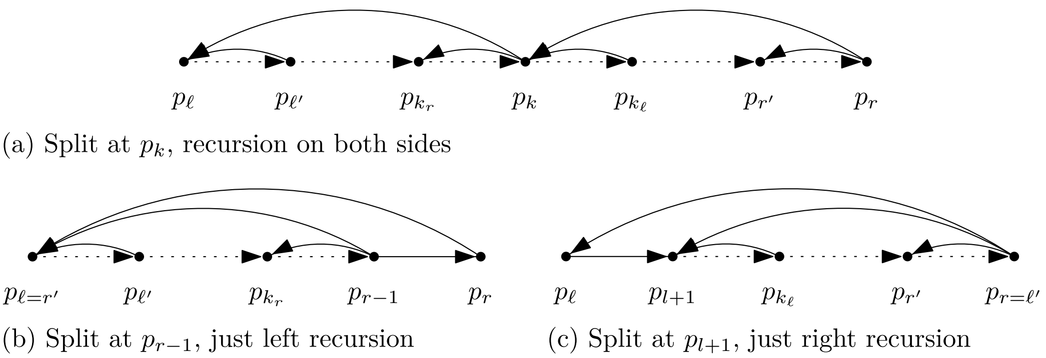


Figure 5.11: Visualization of the recursion of the dynamic program.

Recursion. We add (p_r, p_ℓ) and distinct cases by the indices ℓ, ℓ', r' and r.

(a) For $\ell' < r$ and $r' > \ell$, we test every point p_k with $\ell + 2 \leq k \leq r - 2$ and recurse on $\{p_\ell, \dots, p_k\}$ and $\{p_k, \dots, p_r\}$ (compare to Figure 5.11a). We test every point p_{k_r} with $\ell \leq k_r \leq k - 2$ as leftmost point of $C_{\overrightarrow{G}}(p_{k-1}, p_k)$ and every point p_{k_ℓ} with $k + 2 \leq k_\ell \leq r$ as rightmost point of $C_{\overrightarrow{G}}(p_k, p_{k+1})$. We choose the combination k_r, k, k_ℓ that minimizes the dilation

$$\mathrm{odil}(\ell, \ell', r', r) = \min_{k_r, k, k_\ell} \max\left(\mathrm{odil}(\ell, \ell', k_r, k), \mathrm{odil}(k, k_\ell, r', r), \frac{|p_{k_r} p_{k_\ell}|}{|p_{k-1} p_{k+1}|}\right).$$

(b) For $r' = \ell$ (and $\ell' < r$), maximality implies that (p_{r-1}, p_ℓ) is in the back edge set (compare to Figure 5.11b). Thus, we recurse on $\{p_\ell, \dots, p_{r-1}\}$. As leftmost point for $C_{\overrightarrow{G}}(p_{r-2}, p_{r-1})$ we choose the point p_{k_r} that minimizes the dilation

$$\mathrm{odil}(\ell, \ell', r', r) = \min_{\ell \leq k_r \leq r-3} \max\left(\mathrm{odil}(\ell, \ell', k_r, r-1), \frac{|p_\ell p_r|}{|p_{r-2} p_r|}\right).$$

(c) Analogous, for $\ell' = r$ (and $r' > \ell$), we consider recursively $\{p_{\ell+1}, \dots, p_r\}$ (compare to Figure 5.11c). We test every point p_{k_ℓ} as rightmost point for $C_{\overrightarrow{G}}(p_{\ell+1}, p_{\ell+2})$ to minimize the dilation

$$\mathrm{odil}(\ell, \ell', r', r) = \min_{\ell+3 \leq k_l \leq r} \max\left(\mathrm{odil}(\ell+1, k_\ell, r', r), \frac{|p_\ell p_r|}{|p_\ell p_{\ell+2}|}\right).$$

Termination. Each recursion terminates with a forbidden subproblem (ℓ, ℓ', r', r) or with a set of three points. Invalid combinations are

i) $\ell' < \ell + 2$, $r' > r - 2$ or $\ell > r$ which contradicts the definition of the indices,

ii) $r < \ell + 2$ which implies $|P| < 3$, or

iii) $\ell' < r$, $r' > \ell$ and $\ell' > r'$ which contradicts planarity.

If $\ell' = r$ and $r' = \ell$, due to maximality, it holds that $|P| = 3$. The only (and optimal) solution for a set of three points is the triangle $\Delta_{p_\ell p_{l+1} p_{r=l+2}}$.

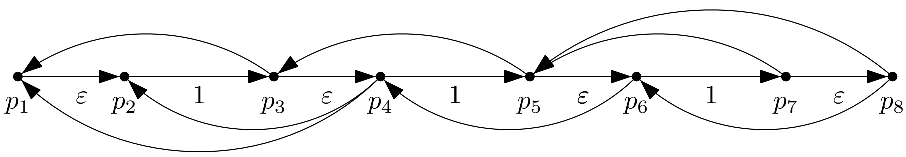
Runtime and Space. The dynamic program goes through all $\mathcal{O}(n^7)$ combinations of $\ell, \ell', k_r, k, k_\ell, r'$ and r. By storing the dilation of each subproblem (ℓ, ℓ', r', r), choosing the optimal combination of k, k_r and k_ℓ does not take additional time. By back tracking, the edge set of a minimum 1-PPB spanner is calculated in $\mathcal{O}(n^7)$ time and $\mathcal{O}(n^4)$ space.

We conclude that this dynamic program returns a minimum 1-PPB graph.

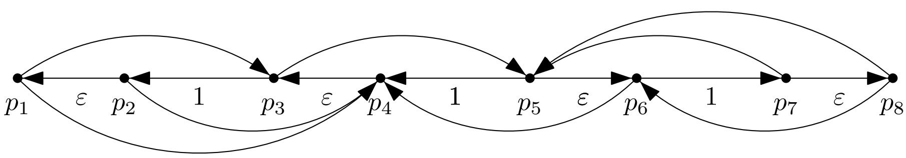
Theorem 5.10. *Given a set P of n points in $\mathbb{R}$, a minimum one-page plane spanner on P can be computed in $\mathcal{O}(n^7)$ time and $\mathcal{O}(n^4)$ space.*

5.3 Two-Page Plane Spanners

Lemma 5.6 states that a 1-PPB graph has smaller or equal dilation than an oriented graph with the same edge set but another orientation. This statement does not hold for two-page plane graphs: Figure 5.12 shows for the same point set a two-page plane $(1+\varepsilon)$-spanner and a minimum 2-PPB spanner with dilation $2+\varepsilon$. However, due to their rich structure, we frequently restrict to 2-PPB graphs. Note that these results do not imply results for general two-page plane graphs.



(a) Minimum 2-PPB graph $\overrightarrow{G} = (P, \overrightarrow{E})$ with oriented dilation $t = \mathrm{odil}(p_3, p_6) = \frac{2+2\varepsilon}{1+2\varepsilon}$.



(b) Two-page plane graph $\overrightarrow{G'} = (P, \overrightarrow{E'})$, where the edges in $\overrightarrow{E'}$ are incident to the same vertices as in $\overrightarrow{E}$, but the orientation may be different. The dilation of $\overrightarrow{G'}$ is $t' = \mathrm{odil}(p_3, p_6) = \frac{1+2\varepsilon}{1+\varepsilon}$. For $\varepsilon < 1/\sqrt{2}$, it holds $t' < t$.

Figure 5.12: A minimum 2-PPB spanner and a general two-page plane graph on the same point set with smaller dilation.

5.3.1 Bounding the Dilation of Two-Page Plane Graphs

By definition, no two-page plane 1-spanner exists for sets of at least four points. We present an instance of six points, which bounds the minimum dilation of 2-PPB spanners by 1.414.

Observation 5.11. *There are one-dimensional point sets where no 2-PPB t-spanner exists for $t < \sqrt{2}$.*

Proof. Let P be a set of six points in $\mathbb{R}$ with the following pairwise distances: $|p_1p_2| = |p_3p_4| = |p_5p_6| = 1$ and $|p_2p_3| = |p_4p_5| = 1/\sqrt{2}$. We show that for every 2-PPB t-spanner for P holds $t \geq \sqrt{2}$.

In the following, we formalize the evaluation of all possible 2-PPB spanners on P. Let the Boolean variable x_{ji} encode if the back edge (p_j, p_i) is in the edge set $\overrightarrow{E}$ or not. By this, every 2-PPB spanner on 6 points is encoded as 10-bit string. Since maximality decreases dilation, it suffices to look at maximal 2-PPB graphs. Every maximal 2-PPB graph on six points contains (p_6, p_1) and exactly 6 further edges. To ensure two-page planarity, we take into account to the following rules:

- $x_{41} + x_{52} + x_{63} \leq 2$
- $x_{31} + x_{41} + x_{42} + x_{52} + x_{53} \leq 4$
- $x_{31} + x_{41} + x_{42} + x_{62} + x_{63} \leq 4$
- $x_{31} + x_{51} + x_{52} + x_{63} + x_{62} \leq 4$
- $x_{41} + x_{51} + x_{52} + x_{62} + x_{64} \leq 4$
- $x_{41} + x_{51} + x_{53} + x_{63} + x_{64} \leq 4$
- $x_{41} + x_{52} + x_{53} + x_{62} + x_{64} \leq 4$
- $x_{42} + x_{52} + x_{53} + x_{63} + x_{64} \leq 4$

Further, we prune the list of all bit encodings by the rule

$$x_{ji} + x_{j'j} = 2 \implies x_{j'i} = 0 \text{ for } i + 1 < j < j' - 1,$$

because $(p_{j'}, p_i)$ does not decrease the dilation if $(p_{j'}, p_j)$ and (p_j, p_i) are in $\overrightarrow{E}$.

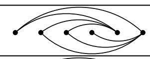
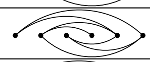
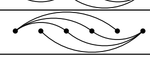
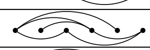
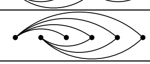
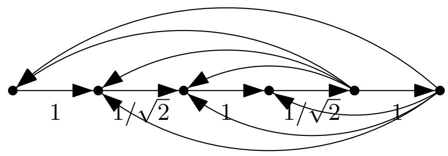
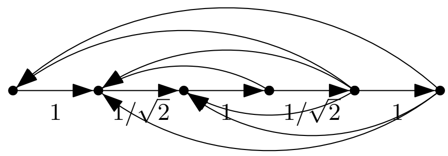
	x_{31}	x_{41}	x_{42}	x_{51}	x_{52}	x_{53}	x_{62}	x_{63}	x_{64}	x_{61}	graph
1	0	0	0	1	1	1	1	1	1	1	
2	0	0	1	1	1	1	1	1	0	1	
3	0	1	1	1	0	1	1	1	0	1	
4	0	1	1	1	1	1	0	0	1	0	
5	0	1	1	1	1	1	1	0	0	1	
6	1	0	1	1	1	0	0	1	1	0	
7	1	1	0	1	0	0	1	1	1	0	
8	1	1	1	0	0	1	0	1	1	0	
9	1	1	1	1	0	0	0	1	1	0	
10	1	1	1	1	1	0	0	0	1	0	
11	1	1	1	1	1	0	1	0	0	1	

Table 5.1: Pruned bit-string encodings of candidates for a minimum 2-PPB graph.

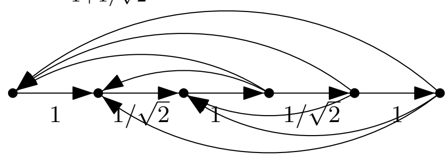
The remaining candidates for a minimum 2-PPB graph are listed in Table 5.1. Note that, as the point set is symmetric, under mirroring and rotation the graphs in line 1 and 11 as well as line 2 and 5 are the same. Computing the dilation of every remaining graph (see Figure 5.13), we conclude a lower bound of $t \geq \sqrt{2}$. $\square$



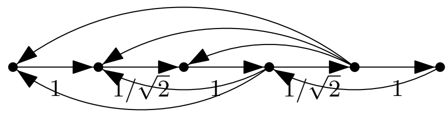
(a) $t = \frac{2+2/\sqrt{2}}{1+1/\sqrt{2}} = 2$



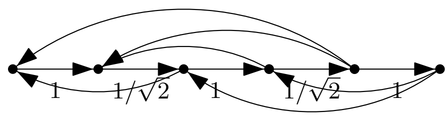
(b) $t = \frac{2+2/\sqrt{2}}{1+1/\sqrt{2}} = 2$



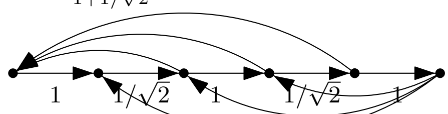
(c) $t = \frac{2+1/\sqrt{2}}{1+1/\sqrt{2}} = 3 - \sqrt{2}$



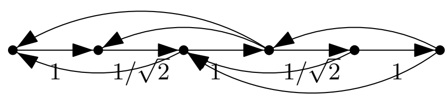
(d) $t = \frac{2+1/\sqrt{2}}{1+1/\sqrt{2}} = 3 - \sqrt{2}$



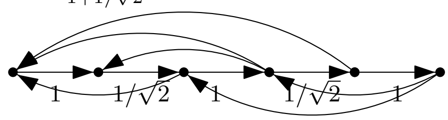
(e) $t = \frac{1+2/\sqrt{2}}{1+1/\sqrt{2}} = \sqrt{2}$



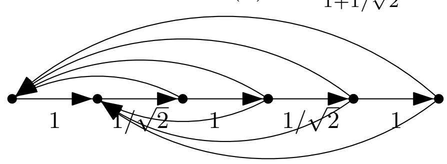
(f) $t = \frac{2+1/\sqrt{2}}{1+1/\sqrt{2}} = 3 - \sqrt{2}$



(g) $t = \frac{2+2/\sqrt{2}}{1+2/\sqrt{2}} = \sqrt{2}$



(h) $t = \frac{2+1/\sqrt{2}}{1+1/\sqrt{2}} = 3 - \sqrt{2}$



(i) $t = \frac{2+2/\sqrt{2}}{1+1/\sqrt{2}} = 2$

Figure 5.13: Remaining candidates for a minimum 2-PPB graph on the point set of size 6 with distances $|p_1p_2| = |p_3p_4| = |p_5p_6| = 1$ and $|p_2p_3| = |p_4p_5| = 1/\sqrt{2}$ and their dilation t.

5.3.2 Constant Dilation Two-Page Plane Spanners

There exists a two-page plane 2-spanner on every one-dimensional point set.

Theorem 5.12. *For every set P of n points in $\mathbb{R}$, the graph $\overrightarrow{G} = (P, \overrightarrow{E})$ with the back edges (p_{i+2}, p_i) for $1 \leq i \leq n-2$ is a two-page plane 2-spanner on P (see Figure 5.14).*

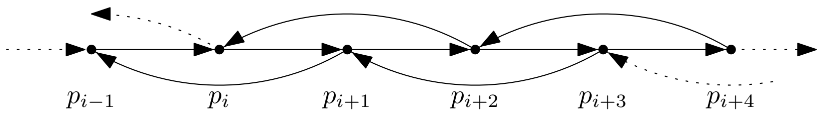


Figure 5.14: Part of a two-page plane 2-spanner.

Proof. Since $(p_{i+2}, p_i) \in \overrightarrow{E}$ for $1 \leq i \leq n-2$, it holds that $\text{odil}(p_i, p_{i+2}) = 1$ and also $\text{odil}(p_j, p_{j+1}) = 1$ for $1 \leq j \leq n-1$. Due to Lemma 5.1, proving $\text{odil}(p_k, p_{k+3}) \leq 2$ for $1 \leq k \leq n-3$, completes the proof that $\overrightarrow{G}$ is a 2-spanner on P.

We consider the closed walk

$$C = p_{k+3} \to p_{k+1} \to p_{k+2} \to p_k \to p_{k+1} \to p_{k+2} \to p_{k+3}.$$

Therefore, we have

$$\begin{aligned}
\text{odil}(p_k, p_{k+3}) &= \frac{|C_{\overrightarrow{G}}(p_k, p_{k+3})|}{|\Delta^*(p_k, p_{k+3})|} \leq \frac{|C|}{|\Delta^*(p_k, p_{k+3})|} \\
&= \frac{|p_{k+1}p_{k+3}| + |p_{k+1}p_{k+2}| + |p_k p_{k+2}| + |p_k p_{k+3}|}{2|p_k p_{k+3}|} \\
&= \frac{2|p_k p_{k+3}| + 2|p_{k+1}p_{k+2}|}{2|p_k p_{k+3}|} = 1 + \frac{|p_{k+1}p_{k+2}|}{|p_k p_{k+3}|} \leq 2. \qquad \square
\end{aligned}$$

Theorem 5.12 implies that Lemma 5.7 does not hold for two-page plane graphs.

Chapter 6
Plane Spanners

Planar graphs are a very widely applied graph class and planarity provides many useful properties for network design. Since adding edges can only reduce the dilation, computing a minimum plane spanner is equivalent to finding a minimum triangulation.

To construct plane oriented spanners, we analyze orientations of triangulations with constant undirected dilation. Das and Joseph [39] proved that every triangulation that fulfills the α-diamond-property has constant dilation.

Definition (α-diamond property)**.** *Let α be a real number with $0 < \alpha < \pi/2$. Let $G = (P, E)$ be a plane undirected graph. We say that G* fulfills the α-diamond property*, when for each edge $e \in E$ at least one of the two isosceles triangles with base edge e and base angle α, does not contain any points of P.*

The currently best upper bound on the dilation t of such a triangulation is $t \leq \frac{8(\pi-\alpha)^2}{\alpha^2 \sin^2(\alpha/4)}$ [25]. The following triangulations fulfill the α-diamond property:

- the minimum weight triangulation for $\alpha = \pi/4.6$ [43],
- the Delaunay triangulation for $\alpha = \pi/4$ [66], and
- the greedy triangulation for $\alpha = \pi/6$ [25].

In Section 6.1, we show that computing a plane oriented spanner with minimum dilation is as hard as computing a minimum dilation triangulation. In Section 6.2, we prove that a suitable orientation of the greedy triangulation yields an $\mathcal{O}(n)$-spanner in general and an $\mathcal{O}(1)$-spanner for points in convex position. In Section 6.3 we provide lower bounds on the dilation of the orientation of other prominent triangulations, namely the Delaunay, minimum dilation and minimum weight triangulation. In Section 6.4, we transfer our results to 1-edge fault-tolerant spanners, i.e., spanners that guarantee a certain dilation under the failure of one arbitrary edge. We prove that the greedy triangulation is a 1-edge fault-tolerant $\mathcal{O}(n)$-spanner in general and an $\mathcal{O}(1)$-spanner for points in convex position.

6.1 Hardness

We study the problem of computing a minimum oriented triangulation on a given point set. The computational complexity of the equivalent undirected problem is a long-standing open problem [27, 46, 51]. To the best of our knowledge, it is not even known whether a PTAS for this problem exists, and for several closely related problems no FPTAS exists, unless P = NP [51]. The following relative hardness result shows that the existence of an FPTAS for computing a minimum oriented triangulation implies an FPTAS for computing a minimum dilation triangulation.

By scaling, w.l.o.g., we may assume that the distance of the closest pair is 1.

Theorem 6.1. *Let P be a set of n points in $\mathbb{R}^2$. We assume that the distance of a closest pair in P is 1. Assume we can compute a $(1+\varepsilon')$-approximation of a minimum oriented triangulation on a given set of n' points in $T(n', \varepsilon')$ time. For any $0 < \varepsilon < 1$ and any $0 < \delta < \varepsilon^2$, we can compute a $(1+\varepsilon)$-approximation of the minimum dilation triangulation on P in $T(4n^2, \varepsilon - \delta)$ time.*

Proof. We reduce from the minimum dilation triangulation problem.

We replace each point in P by a $c_1\varepsilon/n$-small gadget as depicted in Figure 6.1. The gadget consists of $3 \cdot \lceil \frac{4n}{3} \rceil$ points, positioned along a line in small triangles. The distance between two such triangles is $c_2\varepsilon/n^2$. Each triangle is an equilateral triangle with length $c_3\varepsilon^2/n^2$. Let P' be the constructed point set.

Orienting each triangle consistently ensures an oriented dilation of 1 for points of the same triangle. By adding at least two edges between two triangles to get back and forth, the dilation of any two distinct points in the same gadget is less than

$$\frac{2c_2\varepsilon/n^2 + 2 \cdot 3c_3\varepsilon^2/n^2}{2c_2\varepsilon/n^2} \leq 1 + \frac{3c_3}{c_2}\varepsilon. \tag{6.1}$$

Thus, adding the edges as shown in Figure 6.1 does not influence the minimum oriented dilation of the whole graph.

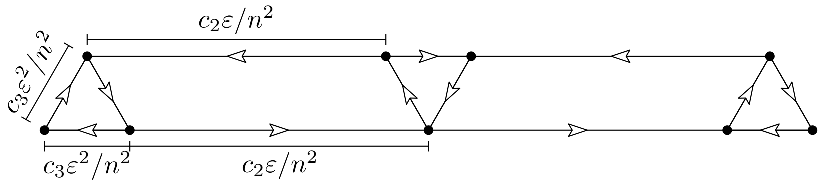


Figure 6.1: Construction in the proof of Theorem 6.1: We replace each point with the shown gadget.

Consider Figure 6.2. As a gadget consists of $3 \cdot \lceil \frac{4n}{3} \rceil$ points, there are at least $2n$ points along the top and bottom of each gadget. This leaves us with room for two edges to every one of the at most $n-1$ adjacent points, one in each direction. By sorting all original points in clock-wise order around each point, we can place up to n edge pairs without disturbing planarity.

Let the oriented graph $\overrightarrow{T}$ be a $(1+\varepsilon')$-approximation of a minimum oriented triangulation on P' and T be the corresponding undirected graph on P. We first show that with a suitable choice of ε' and the constants c_1, c_2 and c_3 the triangulation T is a $(1+\varepsilon)$-approximation of a minimum dilation triangulation on P. Then, we analyze the runtime of this approximation depending on the runtime of computing $\overrightarrow{T}$.

Let $\overrightarrow{T}_{\min}$ be the minimum oriented triangulation on P' and T^* be the minimum-dilation triangulation on P. We will show

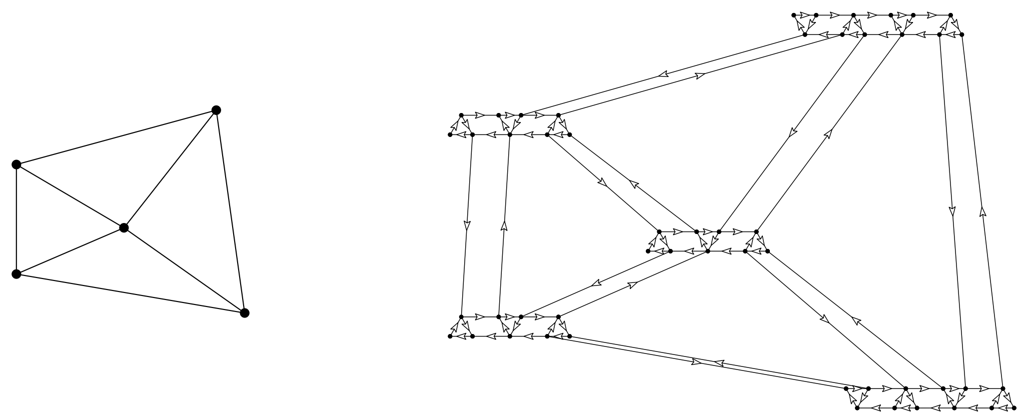


Figure 6.2: Left: Minimum dilation triangulation on a given point set.
Right: Constructed point set and its minimum oriented triangulation. For clarity, we replaced each point only by the necessary amount of triangles instead of 7 triangles per gadget.

i) $\operatorname{dil}(T) \leq (1 + c_4\varepsilon)\operatorname{odil}(\overrightarrow{T})$, and

ii) $\operatorname{odil}(\overrightarrow{T}_{\min}) \leq (1 + c_5\varepsilon)\operatorname{dil}(T^*)$,

where $\operatorname{dil}(\cdot)$ denotes the undirected dilation. The constants $c_4 > 0$ and $c_5 > 0$ are specified in the following steps and depend on c_2 and c_3. We will show that c_4 and c_5 can be chosen arbitrarily close to 0.

Step i) $\operatorname{dil}(T) \leq (1 + c_4\varepsilon)\operatorname{odil}(\overrightarrow{T})$

Let $p, q \in P$ be a worst-case pair in T, i.e., $\operatorname{dil}(T) = \frac{|d_T(p,q)|}{|pq|}$. Let p' and q' be two arbitrary points in P' from the gadgets belonging to p and q and $C_{\overrightarrow{T}}(p', q')$ their shortest closed walk in the minimum oriented triangulation on P'.

Let C be a closed walk through p and q in T constructed from $C_{\overrightarrow{T}}(p', q')$ by the following: for each edge (u', v') in $C_{\overrightarrow{T}}(p', q')$,

- if u' and v' are in the same gadget, we ignore (u', v'), and,
- otherwise, we add uv to C, where u and v are the corresponding points in P.

Note that C can be the same path from p to q and back again. Due to the $c_1\varepsilon/n$-sized gadgets, each edge in C is at most $2c_1\varepsilon/n + 2c_3\varepsilon^2/n^2$ longer than its corresponding edge in $C_{\overrightarrow{T}}(p', q')$. Since the path has at most $n - 1$ edges, it holds that

$$|C| \leq |C_{\overrightarrow{T}}(p', q')| + n \cdot (2c_1\varepsilon/n + 2c_3\varepsilon^2/n^2) \leq |C_{\overrightarrow{T}}(p', q')| + 2(c_1 + c_3)\varepsilon.$$

Since C is a closed walk through p and q, it holds that

$$|d_T(p, q)| \leq |C|/2.$$

Therefore, since $|pq| \geq 1$, we have

$$\operatorname{dil}(T) = \frac{|d_T(p, q)|}{|pq|} \leq \frac{|C|}{2|pq|} \leq \frac{|C_{\overrightarrow{T}}(p', q')| + 2(c_1 + c_3)}{2|pq|}.$$

The third point of $\Delta^*(p',q')$ is next to p' or q' in the same triangle of its gadget. Thus, it holds that

$$\begin{aligned}|\Delta^*(p',q')| &\le 2|p'q'| + 2\cdot 2c_3\varepsilon^2/n^2\\ &\le 2(|pq| + 2c_1\varepsilon/n + 2c_3\varepsilon^2/n^2) + 4c_3\varepsilon^2/n^2\\ &\le 2|pq| + (4c_1 + 8c_3)\varepsilon.\end{aligned}$$

The definition gives us

$$\mathrm{odil}(\overrightarrow{T}) \ge \mathrm{odil}_{\overrightarrow{T}}(p',q') = \frac{|C_{\overrightarrow{T}}(p',q')|}{|\Delta^*(p',q')|}.$$

When this is plugged in, we have

$$\begin{aligned}\mathrm{dil}(T) &\le \frac{\mathrm{odil}(\overrightarrow{T})\cdot|\Delta^*(p',q')| + 2(c_1+c_3)\varepsilon}{2|pq|}\\ &\le \frac{\mathrm{odil}(\overrightarrow{T})\cdot(2|pq| + (4c_1+8c_3)\varepsilon) + 2(c_1+c_3)\varepsilon}{2|pq|}\\ &\le \mathrm{odil}(\overrightarrow{T})\cdot\left(1 + \frac{(3c_1+5c_3)\varepsilon}{|pq|}\right) \le (1 + (3c_1+5c_3)\varepsilon)\,\mathrm{odil}(\overrightarrow{T})\\ &\le (1 + c_4\varepsilon)\,\mathrm{odil}(\overrightarrow{T}).\end{aligned}$$

Step ii) $\mathrm{odil}(\overrightarrow{T}_{\min}) \le (1 + c_5\varepsilon)\,\mathrm{dil}(T^*)$

Let p' and q' be a worst-case pair in $\overrightarrow{T}_{\min}$, i.e., $\mathrm{odil}(\overrightarrow{T}_{\min}) = \frac{|C_{\overrightarrow{T}_{\min}}(p',q')|}{|\Delta^*(p',q')|}$. Let p and q be the corresponding points in P. It holds that

$$|\Delta^*(p',q')| \ge 2|p'q'| \ge 2(|pq| - 2c_1\varepsilon/n - 2c_3\varepsilon^2/n^2).$$

Note that $C_{\overrightarrow{T}_{\min}}(p',q')$ is simply the union of a shortest path from p' to q' and a shortest path from q' to p' in $\overrightarrow{T}_{\min}$. Let Π_1 and Π_2 be the corresponding paths in T. By construction holds that

$$|\Pi_1| \le |d_T(p,q)| + n\cdot 2c_1\varepsilon/n + 2c_3\varepsilon^2/n^2 \le |d_T(p,q)| + (2c_1+2c_3)\varepsilon$$

$$\text{and analogously}\quad |\Pi_2| \le |d_T(q,p)| + (2c_1+2c_3)\varepsilon.$$

(Otherwise T contains two shorter paths which belong to a closed walk in $\overrightarrow{T}_{\min}$ that is shorter than $|C_{\overrightarrow{T}_{\min}}(p',q')|$.)

Since $|d_T(p,q)| = |d_T(q,p)|$ for undirected graphs, we have

$$\begin{aligned}\mathrm{odil}(\overrightarrow{T}_{\min}) &= \frac{|C_{\overrightarrow{T}_{\min}}(p',q')|}{|\Delta^*(p',q')|} \le \frac{2|d_T(p,q)| + 2(2c_1+2c_3)\varepsilon}{2(|pq| - 2c_1\varepsilon/n - 2c_3\varepsilon^2/n^2)}\\ &\le \frac{|d_T(p,q)| + (2c_1+2c_3)\varepsilon}{|pq|}\cdot\frac{|pq|}{|pq| - 2c_1\varepsilon/n - 2c_3\varepsilon^2/n^2}\\ &\le \frac{|d_T(p,q)|}{|pq|} + \frac{(2c_1+2c_3)\varepsilon}{|pq|} \le \mathrm{dil}(T^*) + (2c_1+2c_3)\varepsilon\\ &\le (1 + c_5\varepsilon)\,\mathrm{dil}(T^*).\end{aligned}$$

Since $\operatorname{odil}(\overrightarrow{T}) \leq (1+\varepsilon') \cdot \operatorname{odil}(\overrightarrow{T}_{\min})$, combining i) and ii) gives us

$$\begin{aligned}\operatorname{dil}(T) &\leq (1+c_4\varepsilon)\cdot(1+c_5\varepsilon)\cdot(1+\varepsilon')\operatorname{dil}(T^*)\\ &= (1+c_4\varepsilon+c_5\varepsilon+\varepsilon'+c_4\cdot c_5\varepsilon^2+c_4\varepsilon\varepsilon'+c_5\varepsilon\varepsilon'+c_4\cdot c_5\varepsilon^2\varepsilon')\operatorname{dil}(T^*)\\ &\leq (1+2(c_4+c_5+c_4\cdot c_5)\varepsilon+\varepsilon')\operatorname{dil}(T^*).\end{aligned}$$

Note that c_4 and c_5 are positive constants given by c_1, c_2, c_3 and also the gadget size $c_1\varepsilon/n$ is given by c_2 and c_3. By choosing the positive constants c_2 and c_3 arbitrarily small, we achieve

$$\operatorname{dil}(T) \leq (1+\delta+\varepsilon')\operatorname{dil}(T^*)$$

for any arbitrary small $0 < \delta < \varepsilon^2$.

Recall that, when choosing c_2 and c_3, we ensure that the ratio c_3/c_2 is sufficiently small. In particular, it must be chosen such that it does not dominate the oriented dilation of our construction (compare to Equation (6.1)).

We conclude that approximating a minimum oriented triangulation on P' picks the edges that correspond to a minimum dilation triangulation on P up to a factor of ε. Let $T(n, \varepsilon')$ be the runtime to compute a $(1+\varepsilon')$-approximation of the minimum oriented triangulation on a given point set. Then, since $|P'| = 3 \cdot \lceil \frac{4n}{3} \rceil \cdot n$ and $\varepsilon' = \varepsilon - \delta$, our construction is a $(1+\varepsilon)$-approximation for the minimum dilation triangulation with runtime $T(4n^2, \varepsilon - \delta)$. $\square$

Theorem 6.1 shows that if the minimum dilation triangulation cannot be efficiently approximated within a factor α, then the minimum oriented triangulation also cannot be efficiently approximated within a factor $\alpha - \delta$ for arbitrary small $\delta > 0$. Furthermore, a (F)PTAS for the minimum oriented triangulation problem would imply a corresponding algorithm for the minimum dilation triangulation problem.

6.2 The Greedy Triangulation

Given the relative hardness of computing a minimum oriented triangulation, we next investigate whether plane oriented $\mathcal{O}(1)$-spanners exist. Our main result for points in the plane is an orientation of the greedy triangulation that is an $\mathcal{O}(n)$-spanner and for convex point sets (i.e., point sets for which the points lie in convex position) another orientation that is an $\mathcal{O}(1)$-spanner.

Greedy Triangulation. The greedy triangulation of a point set is obtained by considering all pairs of points by increasing distance, and by adding the straight-line edge if it does not intersect any of the edges already added.

Computing the greedy triangulation on a set of n points takes $\mathcal{O}(n \log n)$ time. For a convex point set, if the order of the points along the convex hull is given, this can be done in linear time [67].

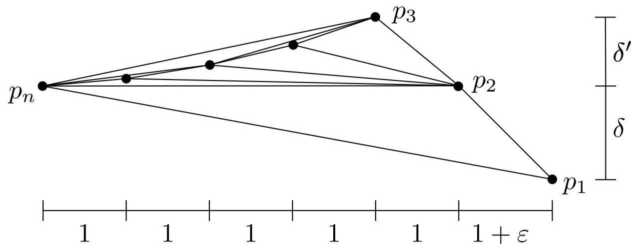


Figure 6.3: Greedy triangulation for a non-convex point set. It is also a minimum weight triangulation.

We first observe that restricting to convex point sets is necessary to obtain constant oriented dilation. Consider the greedy triangulation $T = (P, E)$ on the following non-convex point set $P = \{p_1, \ldots, p_n\}$. As shown in Figure 6.3, we place the points $p_3, \ldots, p_n$ on a very flat parabola. By arbitrarily decreasing the y-distances $\delta, \delta' > 0$, we can reduce the construction to an almost one-dimensional problem. The x-distance of each consecutive pair of points p_i, p_{i+1} is 1 for $2 \leq i < n$. We place p_1 slightly right of p_2 such that the x-distance is $1 + \varepsilon$ for a small $\varepsilon > 0$. Therefore, the x-distance of p_1 to p_i is ε-larger than x-distance of p_2 to p_{i+1} for $3 \leq i < n$. By this, the greedy triangulation added $p_2p_{i+1} \in E$ and discarded $p_1p_i \notin E$ for $3 \leq i < n$. Further, we place p_1 slightly below p_2 so that, due to planarity, p_1 is only adjacent to p_2 and p_n. Since p_1 has degree 2, for any orientation $\overrightarrow{T}$ of T, every shortest closed walk containing p_1 visits p_1, p_2 and p_n. From this construction, for any orientation $\overrightarrow{T}$ and $\delta, \delta', \varepsilon < 1$, it follows that

$$\operatorname{odil}(\overrightarrow{T}) \geq \operatorname{odil}(p_1, p_3) \geq \frac{|p_1p_n|}{|\Delta_{p_1p_2p_3}|} \geq \frac{n - 1 + \varepsilon}{7}.$$

Therefore, every orientation of this greedy triangulation has oriented dilation $\Omega(n)$.

6.2.1 Plane Spanner for Arbitrary Point Sets

We first prove new insights on the undirected greedy triangulation. We bound the perimeter of a face adjacent to an edge pq in the triangulation by a multiple of the minimum perimeter triangle of p and q in P, where the factor is linear in the degree of p and q. We use this result to prove that a certain orientation of the greedy triangulation yields an oriented $\mathcal{O}(n)$-spanner for any point set of size n. These results can be improved for convex point sets (see Section 6.2.2).

Lemma 6.2. *Let P be a point set in $\mathbb{R}^2$. Let $T = (P, E)$ be a greedy triangulation and $pq \in E$. It holds*

$$|\Delta_T(pq)| = \mathcal{O}(\deg_T(p) + \deg_T(q)) \cdot |\Delta^*(p, q)|.$$

Proof. The idea of our proof is to "find" a face in T adjacent to pq and then bound its perimeter.

Let r be a point in $P \setminus \{p, q\}$ such that $|\Delta_{pqr}| = |\Delta^*(p, q)|$. We consider the shortest path $d_T(p, r)$ from p to r in T. Since the greedy triangulation fulfills the α-diamond property [39], it holds $|d_T(p, r)| = \mathcal{O}(1) \cdot |pr|$.

Case 1: $q \notin d_T(p, r)$

Let b_0 be the point, such that pb_0 is the first edge in $d_T(p, r)$. So, b_0 is distinct from p and q ($b_0 = r$ is possible). Since $pb_0 \in d_T(p, r)$, it holds $|pb_0| \leq |d_T(p, r)| = \mathcal{O}(|pr|)$.

W.l.o.g., we assume that p lies left the edge qb_0 (compare to Figure 6.4).

Let $b_0, b_1, \ldots, b_k, b_{k+1}$ be the points adjacent to p in clockwise order with $q = b_{k+1}$. If $k = 0$, then Δ_{pqb_0} is a face of T. Further, for all $1 \leq i \leq k$, let b_i not lie in $\Delta_{pqb_{i-1}}$.

First, we give an intuition for the properties of this sequence, i.e., that b_i is adjacent to p and lies not in $\Delta_{pqb_{i-1}}$: Consider Figure 6.4. Since $pb_{i-1} \in E$, if $qb_{i-1} \in E$, then the triangle $\Delta_{pqb_{i-1}}$ is a face of T and we do not need to consider any further points. Then, by bounding the length of pb_{i-1} (see below), we can bound $|\Delta_T(pq)| \leq |\Delta_{pqb_{i-1}}|$.

Otherwise, if $qb_{i-1} \notin E$, by construction of the greedy triangulation, there is an edge $bb' \in E$ that blocks qb_{i-1}. It holds either $b = p$ and then $b' = b_i$ (properties fulfilled) or b lies inside of $\Delta_{pqb_{i-1}}$. For the latter case, if Δ_{pqb} is a face of T, then we have $|\Delta_T(pq)| \leq |\Delta_{pqb}| \leq |\Delta_{pqb_{i-1}}|$. Otherwise, at least one in $\{pb, qb\}$ is blocked. The edge qb may be blocked by an edge pp', where p' lies right of the edge qb_{i-1}. In this case, we have $p' = b_i$ and the properties are fulfilled. Any other option to block pb or qb requires a point b'' inside of Δ_{pqb}. This point takes the role of b and the argumentation repeats until we found a face of T (whose perimeter is bounded by $|\Delta_{pqb_{i-1}}|$) or there is an edge pb_i, where b_i lies not in $\Delta_{pqb_{i-1}}$.

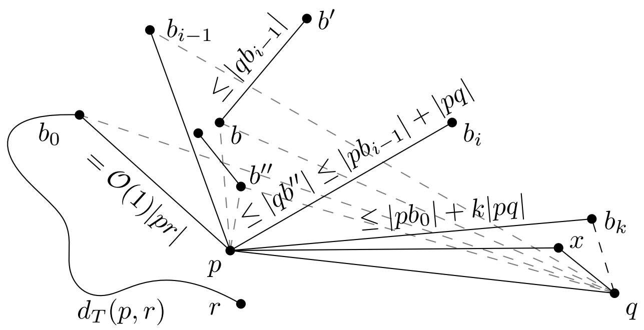


Figure 6.4: Sketch of Case 1 in the proof of Lemma 6.2.

Consider b_k, the last point in our sequence. Either the triangle Δ_{pqb_k} is a face of T, or inside of Δ_{pqb_k} lies a point x such that Δ_{pqx} is a face of T. In both cases, we can bound $|\Delta_T(pq)|$ by $|\Delta_{pqb_k}|$, since it holds that

$$|\Delta_T(pq)| \leq |\Delta_{pqx}| \leq |\Delta_{pqb_k}|.$$

We show that $|\Delta_T(pq)| \leq |\Delta_{pqb_k}| = \mathcal{O}(\deg_T(p)) \cdot |\Delta^*(p, q)|$.

For all $1 \leq i \leq k$, the edge $pb_i \in E$ is the blocker of some edge qq' where q' lies in $\Delta_{pqb_{i-1}}$ ($b_{i-1} = q'$ is possible). Since qq' lies in $\Delta_{pqb_{i-1}}$, we have

$$|pb_i| \leq |qq'| \leq \max(|pq|, |pb_{i-1}|, |qb_{i-1}|) \leq |pq| + |pb_{i-1}|.$$

Applying this repeatedly gives us

$$|pb_i| \leq |pq| + |pb_{i-1}| \leq \ldots \leq i \cdot |pq| + |pb_0| \leq i \cdot |pq| + \mathcal{O}(|pr|) = \mathcal{O}(i) \cdot |\Delta^*(p, q)|.$$

Since $k \leq \deg(p)$, this leads to

$$\begin{aligned}|\Delta_T(pq)| &\leq |\Delta_{pqb_k}| \leq 2|pb_k| + 2|pq| \\ &\leq \mathcal{O}(\deg(p)) \cdot |\Delta^*(p,q)| + |\Delta^*(p,q)| = \mathcal{O}(\deg(p)) \cdot |\Delta^*(p,q)|.\end{aligned}$$

Case 2: $q \in d_T(p,r)$

Since $pq \in E$, the first edge in the shortest path $d_T(p,r)$ is pq. Let b_0 be the point, such that qb_0 is the second edge in $d_T(p,r)$. So, b_0 is distinct from p and q ($b_0 = r$ is possible). Since $qb_0 \in d_T(p,r)$, it holds $|qb_0| \leq |d_T(p,r)| = \mathcal{O}(1) \cdot |pr|$.

We argue analogously to Case 1, but with switched roles for p and q.

W.l.o.g., we assume that q lies left of pb_0. Let $b_0, b_1, \ldots, b_k, b_{k+1}$ be the points adjacent to q in clockwise order with $p = b_{k+1}$. Further, for all $1 \leq i \leq k$, let b_i not lie in $\Delta_{pqb_{i-1}}$. Either Δ_{pqb_k} is a face of T or inside of Δ_{pqb_k} lies a point x such that Δ_{pqx} is a face of T and thus

$$|\Delta_T(pq)| \leq |\Delta_{pqx}| \leq |\Delta_{pqb_k}|.$$

We show that $|\Delta_{pqb_k}| = \mathcal{O}(\deg_T(q)) \cdot |\Delta^*(p,q)|$.

Analogously to Case 1, for all $1 \leq i \leq k$, we have

$$|qb_i| \leq |pq| + |pb_{i-1}| \leq \ldots \leq i \cdot |pq| + |qb_0| \leq i \cdot |pq| + \mathcal{O}(1) \cdot |pr| = \mathcal{O}(i) \cdot |\Delta^*(p,q)|.$$

Since $k \leq \deg(q)$, this leads to

$$\begin{aligned}|\Delta_T(pq)| &\leq |\Delta_{pqb_k}| \leq 2|qb_k| + 2|pq| \\ &\leq \mathcal{O}(\deg(q)) \cdot |\Delta^*(p,q)| + |\Delta^*(p,q)| = \mathcal{O}(\deg(q)) \cdot |\Delta^*(p,q)|.\end{aligned}$$

As both cases are upper bounded by the maximum of $\deg_T(p)$ and $\deg_T(q)$, we conclude

$$|\Delta_T(pq)| = \mathcal{O}(\deg_T(p) + \deg_T(q)) \cdot |\Delta^*(p,q)|. \qquad \square$$

We use Algorithm 4.1 to orient the greedy triangulation. Given the triangular faces of a triangulation $T = (P,E)$ as input, for the resulting orientation $\overrightarrow{T}$ holds $|C_{\overrightarrow{T}}(p,q)| \leq 2|\Delta_T(pq)|$ for any two distinct points p,q with $pq \in E$. Note that this guarantee holds only for pairs with $pq \in E$ (compare to Lemma 4.2). To prove that orienting a greedy triangulation by Algorithm 4.1 yields an oriented $\mathcal{O}(n)$-spanner, it remains to bound $|C_{\overrightarrow{T}}(p,q)|$ for $pq \notin E$.

Theorem 6.3. *Let P be a set of n points in $\mathbb{R}^2$. A plane oriented $\mathcal{O}(n)$-spanner on P can be computed in $\mathcal{O}(n \log n)$ time.*

Proof. Let $T = (P,E)$ be the greedy triangulation of P and $\overrightarrow{T}$ its orientation computed by Algorithm 4.1, given the faces of T as input triangles. For two distinct points $p,q \in P$, we will show that $\mathrm{odil}(p,q) = \mathcal{O}(n)$.

For $pq \in E$, as the degree of any point is in $\mathcal{O}(n)$, combining Lemmas 6.2 and 4.2 guarantees

$$|C_{\overrightarrow{T}}(p,q)| \leq 2|\Delta_T(pq)| = \mathcal{O}(n) \cdot |\Delta^*(p,q)|$$

and therefore $\mathrm{odil}(p,q) = \mathcal{O}(n)$.

For $pq \notin E$, we consider a shortest (undirected) path $d_T(p,q)$ from p to q in T. Let $p_0, p_1, \ldots, p_k$ be the points along this path with $p_0 = p$ and $p_k = q$. Due to Lemma 4.2, for each edge $p_i p_{i+1} \in E$ along this path there exists a shortest closed walk in $\overrightarrow{T}$ of length $|C_{\overrightarrow{T}}(p_i, p_{i+1})| \leq 2|\Delta_T(p_i p_{i+1})|$. The union of all these closed walks along the path $d_T(p,q)$ is a closed walk containing p and q (compare to Figure 6.5). Applying Lemma 6.2 bounds

$$\begin{aligned}|C_{\overrightarrow{T}}(p,q)| &\leq \sum_{p_i p_{i+1} \in d_T(p,q)} |C_{\overrightarrow{T}}(p_i, p_{i+1})| \leq \sum_{p_i p_{i+1} \in d_T(p,q)} 2|\Delta_T(p_i p_{i+1})| \\ &= \sum_{p_i p_{i+1} \in d_T(p,q)} \mathcal{O}(\deg(p_i) + \deg(p_{i+1})) \cdot |\Delta^*(p_i, p_{i+1})|.\end{aligned}$$

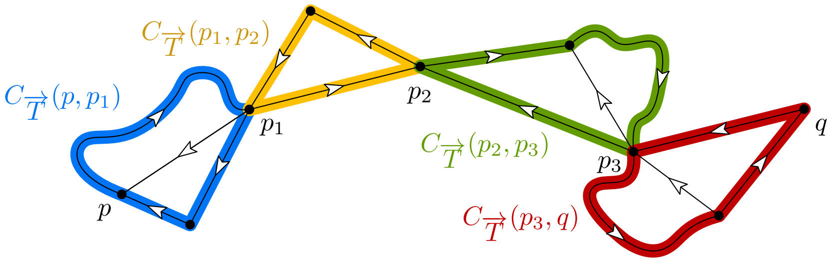


Figure 6.5: Sketch of the proof of Theorem 6.3. We orient T such that for each undirected triangular face of T there exists a shortest closed walk in the orientation $\overrightarrow{T}$ of bounded length. The union of these closed walks along the undirected path $d_T(p,q)$ is a closed walk through p and q.

For all $1 \leq i \leq k-1$, we have

$$|\Delta^*(p_i, p_{i+1})| \leq |\Delta_{p_{i-1} p_i p_{i+1}}| \leq 2|p_{i-1} p_i| + 2|p_i p_{i+1}| \leq 2|d_T(p,q)|.$$

Since the greedy triangulation fulfills the α-diamond property [39], it holds that $|d_T(p,q)| = \mathcal{O}(1)|pq|$. We summarize

$$\begin{aligned}|C_{\overrightarrow{T}}(p,q)| &\leq \sum_{p_i p_{i+1} \in d_T(p,q)} \mathcal{O}(\deg(p_i) + \deg(p_{i+1}))|\Delta^*(p_i, p_{i+1})| \\ &\leq \sum_{p_i p_{i+1} \in d_T(p,q)} \mathcal{O}(\deg(p_i) + \deg(p_{i+1})) \cdot 2|d_T(p,q)| \\ &\leq \sum_{p_i \in d_T(p,q)} \mathcal{O}(\deg(p_i)) \cdot |pq| \leq |pq| \cdot \sum_{p' \in P} \mathcal{O}(\deg(p')) = \mathcal{O}(|E|) \cdot |pq|,\end{aligned}$$

where the last step uses the handshaking lemma. As $|E| = \mathcal{O}(n)$ for triangulations, we have

$$\operatorname{odil}(p,q) = \frac{|C_{\overrightarrow{T}}(p,q)|}{|\Delta^*(p,q)|} \leq \frac{\mathcal{O}(n) \cdot |pq|}{|\Delta^*(p,q)|} = \mathcal{O}(n).$$

The greedy triangulation on a given set of n points can be computed in $\mathcal{O}(n \log n)$ time. Due to Lemma 4.2, orienting its $\mathcal{O}(n)$ faces takes $\mathcal{O}(n \log n)$ time. $\square$

6.2.2 Plane Spanner for Convex Point Sets

In contrast, for convex point sets, we present an orientation of the greedy triangulation that yields an $\mathcal{O}(1)$-spanner.

Lemma 6.2 bounds the perimeter of $\Delta_T(pq)$ by a multiple of the minimum perimeter triangle through p and q. As this factor depends on the degree of p and q, it can be linear in the number of points. We show a constant bound for convex point sets. The proof works similarly to the proof of the dilation of the greedy one-page plane spanner (see Theorem 5.9).

Lemma 6.4. *Let P be a convex point set in $\mathbb{R}^2$. Let $T = (P, E)$ be a greedy triangulation and $pq \in E$. It holds $|\Delta_T(pq)| = \mathcal{O}(|pq|)$.*

Proof. Let r be a point in $P \setminus \{p, q\}$ such that $|\Delta_{pqr}| = |\Delta^*(p, q)|$. We consider the shortest path $d_T(p, r)$ from p to r in T. Analogously to the proof of Lemma 6.2, we distinguish cases based on $q \notin d_T(p, r)$ or $q \in d_T(p, r)$. Since the latter case is straightforward by just exchanging the roles for p and q (compare to Lemma 6.2), we only prove $q \notin d_T(p, r)$.

Let b_0 be the point, such that pb_0 is the first edge in $d_T(p, r)$. For $q \notin d_T(p, r)$, b_0 is distinct from p and q ($b_0 = r$ is possible).

Analogously to Lemma 6.2, we consider a "sequence of blockers". Due to convexity, all these blockers are adjacent to p. Compare to Figure 6.6. Let the points $b_1, \dots, b_k$ be ordered such that $pb_i \in E$ blocks qb_{i-1}. Since pb_k is the last edge in this sequence and due to the convexity of P and the maximality of E, it holds $qb_k \in E$. So, the triangle Δ_{pqb_k} is a face of T. In the following, we show that $|\Delta_{pqb_k}| = \mathcal{O}(|pq|)$.

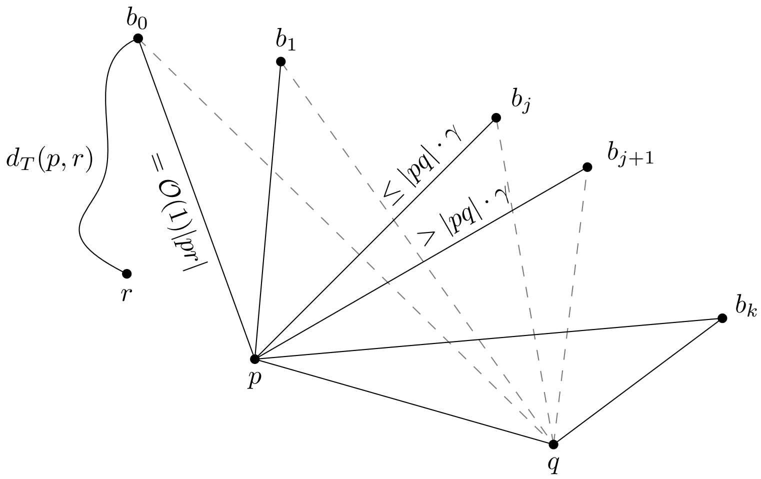


Figure 6.6: "Sequence of blockers": $b_1, \dots, b_k$ where $pb_i \in E$ is the blocker of qb_{i-1}. Δ_{pqb_k} is a face of T.

Note that, since we define the blocker of an edge as the shortest edge that intersects it, for all $1 \leq i \leq k-1$, we have

$$|pb_i| \leq |pb_{i+1}| \tag{6.2}$$

(pb_i is the blocker of qb_{i-1} and thus pb_{i+1} is a longer edge that intersects qb_{i-1}).

Analogously to Lemma 6.2, we can show

$$|pb_i| \leq |qb_{i-1}| \leq |pq| + |pb_{i-1}| \leq \ldots \leq i \cdot |pq| + |pb_0| \leq i \cdot |pq| + \mathcal{O}(|pr|),$$

where the last step follows from the fact that the greedy triangulation fulfills the α-diamond property [39] and thus $|pb_0| \leq |d_T(p,r)| = \mathcal{O}(1) \cdot |pr|$. So, the length of a blocker is increasing linearly in the number of blockers. However, we will show that this sequence contains only a constant number of "long blockers".

Let $\gamma > \pi$ be an arbitrary constant. We call an edge *long* if its length is larger than $|pq| \cdot \gamma$. Let b_j with $1 \leq j \leq k$ be the point such that pb_j is the *last short* blocker, i.e.,

$$|pb_j| \leq |pq| \cdot \gamma \text{ and}$$
$$|pb_{j+1}| > |pq| \cdot \gamma.$$

With this, we bound the length of the last blocker in the sequence by

$$|pb_k| \leq |qb_{k-1}| \leq |pq| + |pb_{k-1}| \leq \ldots \leq (k-j) \cdot |pq| + |pb_j| \leq (k-j+\gamma) \cdot |pq|. \quad (6.3)$$

It is possible that already the first blocker pb_1 is long. Therefore, it remains to prove that the number of long blockers, which is $k-j$, is constant.

Due to convexity and planarity, $pb_{i+1} \in E$ implies $b_ib_{i+2} \notin E$. This means the greedy algorithm added pb_{i+1} and discarded b_ib_{i+2} in later iteration. Therefore, for $1 \leq i \leq k-2$, we have

$$|pb_{i+1}| \leq |b_ib_{i+2}|. \quad (6.4)$$

We lower and upper bound the sum of the pairwise distances $|b_ib_{i+2}|$ for $j \leq i \leq k-2$. Combining these bounds will lead to an upper bound on $k-j$, which is the number of long blockers.

Consider Figure 6.7. By convexity, the points $b_j, \ldots, b_k$ are contained within a semicircle centered at p (in particular, the entire set $\{b_0, b_1, \ldots, b_k, q\}$ lies inside this semicircle). As the lengths of the blockers increase (compare to Lemma 6.2), pb_k is the longest edge in $\{pb_j, \ldots, pb_k\}$. Thus, the radius of this semicircle is $|pb_k|$.

As the segments b_ib_{i+2} for $j \leq i \leq k-2$ overlap, we partition them into two separate sums, one over even indices and one over odd indices. Each of these partial sums is bounded above by the circumference of the semicircle. This gives us

$$\sum_{i=j}^{k-2} |b_ib_{i+2}| \leq 2 \cdot \pi \cdot |pb_k| \leq 2\pi \cdot (k-j+\gamma) \cdot |pq|.$$

Applying Equations (6.2) and (6.4), leads to a lower bound of

$$\sum_{i=j}^{k-2} |b_ib_{i+2}| \geq \sum_{i=j}^{k-2} |pb_{i+1}| \geq \sum_{i=j}^{k-2} |pb_{j+1}| > (k-1-j) \cdot |pq| \cdot \gamma,$$

where the last step uses that pb_{j+1} is the first long blocker.

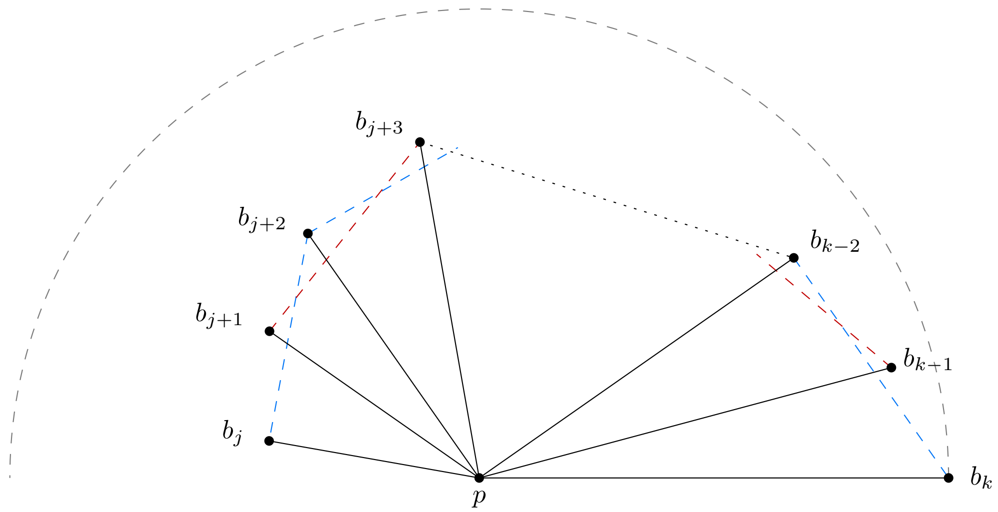


Figure 6.7: The points $b_j, \ldots, b_k$ lie in a semicircle with center p and radius $|pb_k|$. We sum up all pairwise distances $b_i b_{i+2}$ for $j \leq i \leq k-2$ separately for even and odd i (red and blue).

Combining this upper and the lower bound, it follows that the number $k - j$ of long blockers is bounded by a function dependent on γ:

$$(k-1-j) \cdot |pq| \cdot \gamma < \sum_{i=j}^{k-2} |b_i b_{i+2}| \leq 2\pi \cdot (k - j + \gamma) \cdot |pq|$$
$$\iff k - j < \frac{2\pi\gamma}{\gamma - 2\pi} + 1 \quad \text{for } \gamma > 2\pi.$$

For $\gamma = \mathcal{O}(1)$, it holds that $k - j = \mathcal{O}(1)$. Thus, the length of the last blocker pb_k is bounded by a constant times $|pq|$.

Inserting this in Equation (6.3), we conclude

$$|\Delta_T(pq)| \leq |\Delta_{pqb_k}| \leq 2(|pq| + |pb_k|) \leq 2|pq| + 2(k - j + \gamma)|pq| = \mathcal{O}(|pq|). \qquad \square$$

Now, we consider how to orient the greedy triangulation such that the oriented dilation is constant in the undirected dilation. Since the dual graph of a triangulation on a convex point set is a tree, we can orient it such that each face in the orientation is confined by an oriented cycle. We call this a *consistent* orientation. If the neighborhood of each point is sorted and accessible in constant time, such an orientation can be computed in linear time.

Theorem 6.5. *Let P be a convex point set in $\mathbb{R}^2$ of size n, where the points are sorted along the convex hull. A plane oriented $\mathcal{O}(1)$-spanner on P can be computed in $\mathcal{O}(n)$ time.*

Proof. Let $T = (P, E)$ be the greedy triangulation of P and $\overrightarrow{T} = (P, \overrightarrow{E})$ its consistent orientation (which is unique up to reversing all edges).

Note that T is an undirected graph, whereas $\overrightarrow{T}$ is oriented. For two distinct points $p, q \in P$, we will show $|C_{\overrightarrow{T}}(p,q)| = \mathcal{O}(|pq|)$. Since $|\Delta^*(p,q)| \geq 2|pq|$, this gives us $\operatorname{odil}(p,q) = \mathcal{O}(1)$.

For $pq \in E$, Lemma 6.4 guarantees

$$|C_{\overrightarrow{T}}(p,q)| = |\Delta_T(pq)| = \mathcal{O}(|pq|).$$

For $pq \notin E$, we consider the shortest (undirected) path $d_T(p,q)$ from p to q in T. Let $p_0, p_1, \dots, p_k$ be the points along this path with $p_0 = p$ and $p_k = q$. Since the greedy triangulation fulfills the α-diamond property [39], it holds that $|d_T(p,q)| = \mathcal{O}(1) \cdot |pq|$.

Due to the consistent orientation, for each edge $p_ip_{i+1} \in d_T(p,q)$ the adjacent minimum perimeter face $\Delta_T(p_ip_{i+1})$ is oriented consistently. Due to Lemma 6.4, the length of this shortest closed walk through p_i and p_{i+1} in $\overrightarrow{T}$ is bounded by $\mathcal{O}(1)|p_ip_{i+1}|$. The union of all these closed walks along the path $d_T(p,q)$ is a closed walk containing p and q. This gives us

$$|C_{\overrightarrow{T}}(p,q)| \leq \sum_{p_ip_{i+1} \in d_T(p,q)} \mathcal{O}(|p_ip_{i+1}|) = \mathcal{O}(1)|d_T(p,q)| = \mathcal{O}(1) \cdot |pq|.$$

Given the order of the n points along the convex hull, the greedy triangulation can be computed in $\mathcal{O}(n)$ time [67]. If the neighborhood of each point is sorted and accessible in $\mathcal{O}(1)$ time, a consistent orientation can be computed in $\mathcal{O}(n)$ time. □

The aim of this proof was to show that the consistent orientation of the greedy triangulation provides a plane constant dilation spanner at all. Let t_g be the smallest upper bound on the undirected dilation of a greedy triangulation. The proven constant factor is minimized as

$$\min_{\gamma > 2\pi} \frac{2 + 2(k - j + \gamma)}{2} t_g < \min_{\gamma > 2\pi} \left(1 + \frac{2\pi\gamma}{\gamma - 2\pi} + 1 + \gamma\right) \cdot t_g = (2 + 8\pi) t_g < 27.14 \cdot t_g,$$

(for details compare to Lemma 6.4 and Theorem 6.5).

Corollary 6.6. *Let P be a convex point set in $\mathbb{R}^2$. Let t_g be the undirected dilation of a greedy triangulation T on P. By orienting T consistently, we get a plane oriented t-spanner on P with $t < 27.14 \cdot t_g$.*

6.3 Other Triangulations

As the greedy triangulation leads to a plane oriented $\mathcal{O}(n)$-spanner for n points and $\mathcal{O}(1)$-spanner for convex point sets, the question arises whether the α-diamond property implies the existence of oriented spanners.

Minimum Weight Triangulation. The minimum weight triangulation has the α-diamond property [43] but does not yield an $\mathcal{O}(1)$-spanner in general. This can be seen by the example that was presented for the greedy triangulation in Figure 6.3. By essentially the same argument, the minimum weight triangulation has the same edges incident to p_1, which results in an oriented dilation of $\Omega(n)$. Whether the minimum weight triangulation is an oriented $\mathcal{O}(n)$-spanner or $\mathcal{O}(1)$-spanner for convex point sets, we leave as an open problem.

Delaunay Triangulation. For the Delaunay triangulation the situation is even worse. The Delaunay triangulation has the α-diamond property [66] and is the basis for many undirected plane spanner constructions [26]. However, for $n \geq 4$ its oriented dilation can be arbitrary large, no matter how we orient its edges, even for convex point sets.

Figure 6.8 shows the Delaunay triangulation T of a convex point set of size 4. For any orientation of T, the shortest closed walk through p_2 and p_4 contains p_1. We place p_2, p_3, p_4 arbitrarily close to each other without decreasing the radius of the circle through them. By placing p_1 in the circle and sufficiently far away from p_2, p_3 and p_4, the oriented dilation $\operatorname{odil}(p_2, p_4) \geq \frac{|p_2p_1|+|p_1p_4|}{|\Delta_{p_2p_3p_4}|}$ can be made arbitrarily large. The example can be modified to include more points without changing $\operatorname{odil}(p_2, p_4)$.

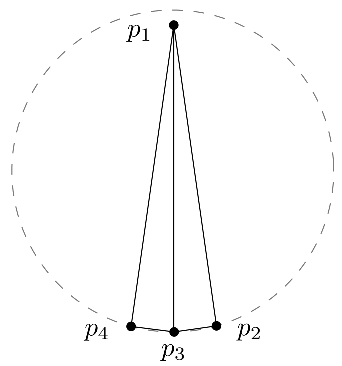


Figure 6.8: Delaunay triangulation for a convex point set. It is also a minimum dilation triangulation.

Minimum Dilation Triangulation. Essentially the same example as for the Delaunay triangulation shows that there are convex point sets for which any orientation has arbitrarily large oriented dilation. In Figure 6.8, consider the case in which we have fixed all positions except for the y-coordinates of p_2 and p_4. Let y_0 be the y-coordinate of p_3, and let $y_0 + \varepsilon$ be the y-coordinate of p_2 and p_4. By decreasing $\varepsilon > 0$, we can make sure that the minimum dilation triangulation chooses the same diagonal as the Delaunay triangulation, since the undirected dilation between p_2 and p_4 by the path through p_3 becomes arbitrarily close to 1. However, as in the Delaunay triangulation, the oriented dilation between p_2 and p_4 can be made arbitrarily large.

Considering the Delaunay and minimum dilation triangulation, we conclude that the α-diamond property is not sufficient to find plane oriented spanners.

Greedy Triangles Triangulation. Algorithm 4.1 computes an oriented spanner by sorting the input triangles by perimeter and orienting them in this order consistently whenever this is still possible. A natural question is whether a similar greedy strategy could produce a triangulation with small oriented dilation.

Given a set of n points, we construct an undirected triangulation as follows. Sort all $\binom{n}{3}$ triangles by perimeter and then add the edges of the triangle to the current edge set E if none of them intersects with an edge in E.

At first glance, this approach appears promising. However, the triangulation presented in Figure 6.3 is returned by this algorithm. Therefore, this construction is also an $\Omega(n)$-spanner. In contrast to the minimum weight triangulation, which is difficult to analyze due to its global optimization objective, it might be promising to prove also an upper bound of $\mathcal{O}(n)$ on the oriented dilation of the *greedy triangles triangulation* on non-convex point sets.

6.4 Relation to Fault-Tolerant Spanners

In communication as well as road networks, the system should be resilient to edge (or vertex) failures, e.g., a connection is turned off or a road is closed for construction. Despite the failure of one or more vertices or edges, the spanner should retain its key properties – such as short paths between pairs of points.

Formally, given a point set P and parameters t and k, a *k-edge fault-tolerant t-spanner* $G = (P, E)$ is a subgraph of the complete undirected graph on P such that for every edge set $E' \subset E$ of size $|E'| = k$, the shortest path in $G \setminus E'$ between any pair of points is at most a factor t longer than their shortest path in $K(P) \setminus E'$. The latter describes the shortest path "under the circumstances" [74].

Definition (Fault-tolerant spanner)**.** *Given $k \in \mathbb{N}$, an undirected graph $G = (P, E)$ is called a k-edge fault-tolerant t-spanner if*

$$t = \max_{\substack{E' \subseteq E,\ |E'|=k \\ p,q \in P}} \frac{|d_{G\setminus E'}(p,q)|}{|d_{K(P)\setminus E'}(p,q)|}.$$

Fault-tolerant spanners were introduced in [69] and have since gained a lot of interest [1, 17, 77]. Lukovszki [72] proved that every spanner that is resilient to vertex faults is also resilient to edge faults. There is a simple algorithm to transform any t-spanner into a k-fault-tolerant t-spanner [74]. Further, for a fixed k, we can compute a sparse k-fault-tolerant spanner [69, 70]. However, none of these constructions guarantees a plane graph.

In the following, we consider 1-edge fault-tolerant spanners. In this case, the dilation simplifies to

$$\max_{pq \in E} \left(\frac{|d_{G\setminus pq}(p,q)|}{\min\limits_{r \in P\setminus\{p,q\}} (|pr| + |rq|) = |\Delta^*(p,q)| - |pq|} \, , \max_{p',q' \in P\setminus\{p,q\}} \frac{|d_{G\setminus pq}(p',q')|}{|p'q'|} \right).$$

We observed that oriented and 1-edge fault-tolerant spanners are quite related models: Both, the dilation of a spanner under the failure of an edge pq and the oriented dilation of two points p, q in a given oriented graph, depend on the perimeter of the smallest triangle in P containing p and q (see also Lemma 6.8).

However, undirecting an oriented spanner with constant oriented dilation does not lead to a fault-tolerant spanner with bounded dilation (compare to Figure 6.9). But, in the following, we prove that orienting a suitable 1-edge fault-tolerant spanner, namely the greedy triangulation, yields an oriented spanner with (up to a constant factor) the same dilation (see Theorems 6.9 and 6.10).

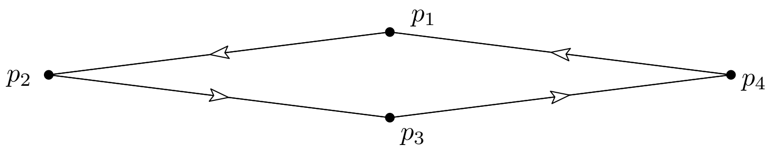


Figure 6.9: An oriented 2-spanner. If p_1 and p_3 are very close to each other and p_2 and p_4 arbitrarily far away, the underlying undirected graph is not fault-tolerant.

Plane 1-Edge Fault-Tolerant Spanners

Using the same gadgets as in Theorem 6.1, we can prove that computing a minimum plane 1-edge fault-tolerant spanner is at least as hard as computing the minimum dilation triangulation. Recall Figures 6.1 and 6.2. As each edge in the minimum dilation triangulation corresponds to two (now undirected) edges, the failure of one such edge does not change the dilation. The dilation of points in the same gadget under the failure of one edge is less than $1 + \varepsilon$.

Corollary 6.7. *An FPTAS for computing a minimum dilation* 1*-edge fault-tolerant triangulation would imply an FPTAS for minimum dilation triangulation.*

To analyze the 1-edge fault-tolerance of a triangulation, we use the following:

Lemma 6.8. *Let $T = (P, E)$ be a triangulation that is an undirected constant dilation spanner, i.e., $\frac{|d_T(p,q)|}{|pq|} = \mathcal{O}(1)$ for all $p, q \in P$. The dilation t of T under the failure of an edge $pq \in E$ is $t = \mathcal{O}\left(\frac{|\Delta_T(pq)|}{|\Delta^*(p,q)|}\right)$.*

Proof. Let $\varphi = \frac{|\Delta_T(pq)|}{|\Delta^*(p,q)|}$, so $|\Delta_T(pq)| = \varphi|\Delta^*(p,q)|$. Note that $\varphi \geq 1$.

The dilation of the pair pq under the failure of pq is

$$\frac{|d_{T\setminus pq}(p,q)|}{|d_{K(P))\setminus pq}(p,q)|} = \frac{|\Delta_T(pq)| - |pq|}{\min\limits_{r \in P\setminus\{p,q\}} |pr| + |rq|} \leq \frac{|\Delta_T(pq)|}{|\Delta^*(p,q)| - |pq|} = \frac{\varphi|\Delta^*(p,q)|}{|\Delta^*(p,q)| - |pq|} \leq 2\varphi.$$

Now, we bound the dilation of a pair $p'q' \neq pq$ under the failure of pq, which is $\frac{|d_{T\setminus pq}(p',q')|}{|p'q'|}$. We show $|d_{T\setminus pq}(p',q')| = \varphi \cdot \mathcal{O}(|p'q'|)$ to complete the proof.

Let $d_T(p',q')$ be the shortest path from p' to q'. Since T is an undirected $\mathcal{O}(1)$-spanner, it holds $|d_T(p',q')| = \mathcal{O}(1) \cdot |p'q'|$.

If $pq \notin d_T(p',q')$, then we have $d_{T\setminus pq}(p',q') = d_T(p',q')$. It holds that

$$|d_T(p',q')| = \mathcal{O}(|p'q'|) \leq \varphi \cdot \mathcal{O}(|p'q'|).$$

If $pq \in d_T(p',q')$, we may assume, w.l.o.g., that this path from p' to q' visits p before q. So, we have

$$|d_T(p',q')| = |d_T(p',p)| + |pq| + |d_T(q,q')|.$$

The length of the shortest path from p' to q' in $T \setminus pq$ is bounded by

$$|d_{T\setminus pq}(p',q')| \leq |d_T(p',p)| + |\Delta_T(pq)| - |pq| + |d_T(q,q')| \leq |d_T(p',q')| + \varphi|\Delta^*(p,q)|.$$

Since $\Delta^*(p,q)$ is the minimum perimeter triangle of p and q, it holds that

$$|\Delta^*(p,q)| \leq |\Delta_{pqq'}| \leq 2(|pq| + |qq'|).$$

Further, since $pq \in d_T(p',q')$, it holds that

$$|pq| + |qq'| \leq |pq| + |d_G(q,q')| \leq |d_T(p',q')| = \mathcal{O}(|p'q'|).$$

Thus, the dilation of the pair $p'q'$ under the failure of pq is

$$\frac{|d_{T\setminus pq}(p',q')|}{|p'q'|} \leq \frac{|d_T(p',q')| + \varphi|\Delta^*(p,q)|}{|p'q'|} = \frac{\mathcal{O}(|p'q'|) + \varphi \cdot \mathcal{O}(|p'q'|)}{|p'q'|} = \mathcal{O}(1) \cdot \varphi. \quad \square$$

Due to Lemma 6.8, for a given triangulation $T = (P,E)$, an upper bound on t such that T is a 1-edge fault-tolerant t-spanner can be computed as

$$t = \mathcal{O}\left(\max_{pq \in E} \frac{|\Delta_T(pq)|}{|\Delta^*(p,q)|}\right).$$

Hence, if an asymptotic bound suffices, the dilation under the failure of exact one edge can be evaluated efficiently: For each edge $pq \in E$, it is sufficient to compare the (at most two) triangles of T that are adjacent to pq with a minimum perimeter triangle in P that contains p and q. This takes $\mathcal{O}(n)$ time per edge.

In contrast, the best known approach to compute the exact dilation requires computing the undirected dilation of the graph $T \subset pq$ for every edge $pq \in E$. No nontrivial lower bound for computing the dilation of a given graph is known, but the problem is conjectured to be APSP-hard [55]. Since Dijkstra's algorithm runs in $\mathcal{O}(n)$ time on planar graphs [62], this leads to a runtime of $\mathcal{O}(n^2)$ for solving APSP on triangulations. Consequently, if an asymptotic bound suffices, the above approach saves up to a factor of $\mathcal{O}(n)$ in runtime.

Lemma 6.8 applies for any triangulation that is an undirected constant dilation spanner. Since every triangulation that fulfills the α-diamond property has constant dilation [39], we consider several popular triangulation with this property.

Greedy Triangulation. As for oriented spanners, the lower bound on the fault-tolerance of the greedy triangulation is $\Omega(n)$ for non-convex point sets: Consider the greedy triangulation T presented in Figure 6.3. The dilation of $T \setminus p_1p_2$ is

$$t \geq \frac{|p_1p_n| + |p_np_3|}{|p_1p_3|} \geq \frac{n-1+\varepsilon+n-2}{2+\varepsilon+\delta+\delta'} = \Omega(n).$$

Thus, it is a 1-edge fault-tolerant $\Omega(n)$-spanner on the shown point set.

As the maximum degree is $n-1$, Lemma 6.2 gives us $|\Delta_T(pq)| \leq \mathcal{O}(n)|\Delta^*(p,q)|$ for any $pq \in E$. Applying Lemma 6.8 leads to an upper bound for any point set.

Theorem 6.9. *Let P be a point set in $\mathbb{R}^2$ of size n. The greedy triangulation on P is a 1-edge fault-tolerant $\mathcal{O}(n)$-spanner on P.*

Combining Lemmas 6.4 and 6.8 leads to a constant dilation for convex point sets.

Theorem 6.10. *Let P be a convex point set in $\mathbb{R}^2$. The greedy triangulation on P is a 1-edge fault-tolerant $\mathcal{O}(1)$-spanner on P.*

Minimum Weight Triangulation. The same example as for the greedy triangulation in Figure 6.3 is also a minimum weight triangulation. By essentially the same argument, the minimum weight triangulation is a 1-edge fault-tolerant $\Omega(n)$-spanner. Whether the minimum weight triangulation is a 1-edge fault-tolerant $\mathcal{O}(n)$-spanner, we leave as an open problem.

Delaunay Triangulation. We consider the convex point set and its Delaunay triangulation T presented in Figure 6.8. If the edge p_2p_3 fails, the shortest path from p_2 to p_4 visits p_1. The dilation t of T under the failure of the edge p_2p_3 is $t \geq \frac{|p_1p_2|+|p_1p_4|}{|p_2p_4|}$. We can place p_2, p_3, p_4 arbitrarily close to each other without decreasing the radius of the circle through them. If we do so, t can be made arbitrarily large. The example can be extended with more points without changing this lower bound. Note that in this example, the value $\frac{|\Delta_T(p_2p_3)|}{|\Delta^*(p_2,p_3)|}$ can be arbitrarily large. So, there are convex point sets for which the Delaunay triangulation is not fault-tolerant.

Minimum Dilation Triangulation. As for oriented spanners, we can argue that the triangulation of the convex point set presented in Figure 6.8 is also a minimum dilation triangulation. As for the Delaunay triangulation, the dilation under the failure of the edge p_2p_3 can be made arbitrarily large. So, there are convex point sets for which the minimum dilation triangulation is not fault-tolerant.

We conclude that a triangulation that fulfills the α-diamond property does not imply a plane fault-tolerant spanner.

Chapter 7
Sparse Spanners

In this chapter, we construct oriented spanners with a linear number of edges for point set in higher-dimensional spaces. The well-separated pair decomposition is a key tool of our constructions.

In Section 7.1, we show that, given t and m, it is NP-hard to decide if there is an oriented t-spanner on a given point set with at most m edges. Next, we explore constructions by sparsifying a dense graph while (almost) preserving its dilation. In Section 7.2, we present an algorithm that extracts a sparse $(t+\varepsilon)$-spanner out of a given t-tournament. Combined with our results on tournaments (see Section 4.3), we derive in Section 7.3 the formal result: a sparse $(5/3+\varepsilon)$-spanner computable in $\mathcal{O}(n^3)$ time for n points in any metric space that admits a WSPD. Since this method depends on computing tournaments, its runtime and space complexity are relatively high. Our main result is a $(2+\varepsilon)$-spanner with $\mathcal{O}(n)$ edges in $\mathcal{O}(n \log n)$ time for point sets in a Euclidean space of constant dimension. In Section 7.4, we move beyond worst-case dilation and construct a sparse oriented spanner with average dilation $1+\varepsilon$ for sufficiently large Euclidean point sets.

7.1 Hardness

We show that computing a minimum sparse spanner is NP-hard.

Theorem 7.1. *Given a point set P in $\mathbb{R}^d$ with $d \geq 2$ and the parameters t and m, it is NP-hard to decide if there is an oriented t-spanner on P with at most m edges.*

Proof. We prove the problem is NP-hard even for a point set in the plane $\mathbb{R}^2$, which implies the result for all $d \geq 2$. We reduce from the NP-complete Hamilton circuit problem for grid graphs [63]. We start with a polynomial-size node-induced subgraph G of a grid whose cells have side length one. Let P be its point set and set $n = |P|$. We choose $m = n$ and $t = n \cdot c$, where c is a rational number such that

$$\frac{1}{2+\sqrt{2}} \leq c < \frac{1}{2+\sqrt{2}} + \frac{0.1}{n} < \frac{1}{2+\sqrt{2}} + \frac{1}{n} \cdot \frac{\sqrt{2}-1}{2+\sqrt{2}} .$$

To choose t, it is sufficient to compute the first $\mathcal{O}(\log(n))$ digits of c, which is possible in polynomial time.

We will show that there is an oriented t-spanner on P with at most m edges if and only if G has a Hamiltonian cycle.

First, we show that if G has a Hamiltonian cycle, then there is such a spanner. Let E_{HC} be the edges of a Hamiltonian cycle in G. We bound the dilation of the oriented graph $\overrightarrow{G'} = (P, \overrightarrow{E_{HC}})$ with $|\overrightarrow{E_{HC}}| = n$ edges. Since the edges of $\overrightarrow{G'}$ form a Hamiltonian cycle, a shortest closed walk of any point tuple is this cycle. Further, all its edges have length 1.

As the points in P are embedded on a grid whose cells have side length one, w.l.o.g., we may assume the existence of an "L-shaped" point triple $p_1, p_2, p_3 \in P$ with $|p_1p_2| = |p_2p_3| = 1$ and $|p_1p_3| = \sqrt{2}$. If no such triple exists, G is a one-dimensional path and the Hamilton circuit problem for this graph class is trivial. Therefore, we have

$$\min_{p,q \in P} |\Delta^*(p,q)| = 2 + \sqrt{2}.$$

Thus, the dilation t of $\overrightarrow{G'}$ is

$$\max_{p,q \in P} \frac{|C_{\overrightarrow{G'}}(p,q)|}{|\Delta^*(p,q)|} \le \frac{n \cdot 1}{2 + \sqrt{2}} \le t.$$

Now, we show that if G has no Hamiltonian cycle, then there is no oriented t-spanner on P with at most m edges. Any strongly connected oriented graph on P with $m = n$ edges must be a cycle. Again, a shortest closed walk of any pair of points is this cycle C. So, the dilation t of the graph $\overrightarrow{G'} = (P, \overrightarrow{C})$ is

$$\max_{p,q \in P} \frac{|C_{\overrightarrow{G'}}(p,q)|}{|\Delta^*(p,q)|} = \frac{\sum_{e \in \overrightarrow{C}} |e|}{\min_{p,q \in P} |\Delta^*(p,q)|}\,.$$

Since G has no Hamiltonian cycle, $\overrightarrow{G'}$ must contain an edge whose length is at least $\sqrt{2}$. Therefore, the dilation t of $\overrightarrow{G'}$ is at least

$$\sum_{e \in \overrightarrow{C}} \frac{|e|}{2 + \sqrt{2}} \ge \frac{n - 1 + \sqrt{2}}{2 + \sqrt{2}} > t. \qquad \square$$

7.2 Sparsification of Tournaments

In this section, we show how to compute a sparse subgraph of a given tournament whose dilation is negligibly larger. Algorithms to orient complete graphs such that the oriented dilation is small are presented in Section 4.3.

Given a tournament on a point set P that contains for every two distinct points $p, q \in P$ an oriented cycle with $\mathcal{O}(1)$ edges whose length is at most $t|\Delta^*(p,q)|$, we compute a sparse $(t + \varepsilon)$-spanner. Thus, we reduce the number of edges from $\binom{n}{2}$ to $\mathcal{O}(n)$ while increasing the dilation only by a given parameter $0 < \varepsilon < 2$. The idea is to extract a linear number of cycles in the tournament, each containing $\mathcal{O}(1)$ edges, such that for each pair of points exists a sufficiently short closed walk passing through them.

The algorithm that computes such an orientation is given in Algorithm 7.1. It first computes the WSPD for the given point set. For each well-separated pair, we

Algorithm 7.1 Sparse $(t+\varepsilon)$-Spanner.

Require: Set P of n points, constant $c \in \mathbb{N}$, positive real numbers $\varepsilon < 2, t \geq 1$, s, and a tournament $\overrightarrow{K}$ on P such that for any pair $p, q \in P$ there is an oriented cycle through p and q with at most c edges and length at most $t|\Delta^*(p,q)|$
Ensure: $(t+\varepsilon)$-spanner for P with $\mathcal{O}(n)$ edges

```
Compute an s-WSPD {A_i, B_i} for 1 ≤ i ≤ m = O(n)
E⃗ = ∅
for i = 1 to m do
    𝒜_i = Pick min(|A_i|, 2) arbitrary points of A_i
    ℬ_i = Pick min(|B_i|, 2) arbitrary points of B_i
    for each ab ∈ 𝒜_i × ℬ_i do
        Regarding (a, b) or (b, a) ∈ K⃗, compute the shortest path Π in K⃗ with at
        most c − 1 edges from b to a (or resp. from a to b)
        Add all edges of Π and (a, b) (resp. (b, a)) to E⃗
return oriented graph G⃗ = (P, E⃗)
```

add up to four cycles with at most c edges. Therefore, we run the Bellman-Ford algorithm for only $c-1$ iterations, to compute the shortest path in $\overrightarrow{K}$ with at most $c-1$ edges between two given points. Note that the returned graph $\overrightarrow{G}$ is a subgraph of $\overrightarrow{K}$. Since we add at most four cycles with $c = \mathcal{O}(1)$ edges for m well-separated pairs, $\overrightarrow{G}$ has $\mathcal{O}(m)$ edges.

We pick up to two points of A and up to two points of B for each well-separated pair $\{A, B\}$. In particular, picking two points is only necessary, if a set has exactly size 2. Choosing two points for any set that contains at least two points, causes a minor computational overhead, but improves the readability of the proof.

Now, we show that Algorithm 7.1 constructs a sparse oriented $(t+\varepsilon)$-spanner. As the WSPD is the only limiting component in this algorithm, Algorithm 7.1 works for metric spaces that admit a WSPD.

Theorem 7.2. *Let $(P, |\cdot|)$ be a metric space with n points that admits a WSPD with m pairs constructible in $T(n)$ time. Let $c \in \mathbb{N}$ be a constant and $t \geq 1$ and $0 < \varepsilon < 2$ be real numbers. Let $\overrightarrow{K}$ be a tournament on P such that for any $p, q \in P$ there is an oriented cycle through p and q with at most c edges and length at most $t|\Delta^*(p,q)|$. An oriented $(t+\varepsilon)$-spanner for P with $\mathcal{O}(cm)$ edges can be constructed in $\mathcal{O}(mcn^2 + T(n))$ time using $\mathcal{O}(n^2)$ space.*

Proof. Let $\overrightarrow{G} = (P, \overrightarrow{E})$ be the graph returned by Algorithm 7.1. Let $s = 10t/\varepsilon + 6$. For any two distinct points $p, q \in P$, we will prove $\mathrm{odil}(p,q) \leq t + \varepsilon$. The proof is by induction on the rank of $|pq|$ in the sorted sequence of all $\binom{n}{2}$ pairwise distances.

If pq is a closest pair, then the WSPD contains the pair $\{\{p\}, \{q\}\}$. Therefore, the algorithm added an oriented cycle through p and q whose length is at most $t|\Delta^*(p,q)|$ to $\overrightarrow{E}$. Hence, we have $\mathrm{odil}(p,q) \leq t \leq t + \varepsilon$.

Assume that pq is not a closest pair. Let $\{A, B\}$ be the well-separated pair with $p \in A$ and $q \in B$. Let $\mathcal{A} \subseteq A$ and $\mathcal{B} \subseteq B$ be the sets selected by Algorithm 7.1 during processing the pair $\{A, B\}$. There are three cases.

Case 1: $p \notin \mathcal{A}$ and $q \notin \mathcal{B}$

Since p is not picked for A, it holds $|A| \geq 3$. Analogously, it holds $|B| \geq 3$. Let $a, a' \in A \setminus \{p\}$ be the points picked for A and $b, b' \in B \setminus \{q\}$ for B. We will prove

$$|C_{\overrightarrow{G}}(p,q)| \leq |C_{\overrightarrow{G}}(p,a)| + |C_{\overrightarrow{G}}(a,b)| + |C_{\overrightarrow{G}}(b,q)| \leq (t+\varepsilon) \cdot |\Delta^*(p,q)|.$$

Due to WSPD-Property 1, it holds that $|pa| < |pq|$ and $|bq| < |pq|$. Thus, we apply induction to bound $|C_{\overrightarrow{G}}(p,a)|$ and $|C_{\overrightarrow{G}}(b,q)|$. Since $a \in \mathcal{A}$ and $b \in \mathcal{B}$, the returned graph $\overrightarrow{G}$ contains an oriented cycle through a and b whose length is at most $t|\Delta^*(a,b)|$. This gives us

$$|C_{\overrightarrow{G}}(p,q)| \leq (t+\varepsilon)|\Delta^*(p,a)| + t|\Delta^*(a,b)| + (t+\varepsilon)|\Delta^*(b,q)|.$$

Using WSPD-Property 1, it holds that

$$|\Delta^*(p,a)| \leq |\Delta_{paa'}| \leq 3 \cdot 2/s \cdot |pq| \leq 3/s|\Delta^*(p,q)|,$$

where the last inequality $|\Delta^*(p,q)| \geq 2|pq|$ follows from the triangle inequality. Analogously, $|\Delta^*(b,q)|$ is bounded by $3/s|\Delta^*(p,q)|$. By Lemma 3.3, we bound $|\Delta^*(a,b)| \leq (1+4/s)|\Delta^*(p,q)|$.

Since $s = \frac{10t+6\varepsilon}{\varepsilon}$, we achieve in total

$$\begin{aligned}|C_{\overrightarrow{G}}(p,q)| &\leq 2(t+\varepsilon)3/s|\Delta^*(p,q)| + t(1+4/s)|\Delta^*(p,q)| = \left(t + \frac{10t+6\varepsilon}{s}\right)|\Delta^*(p,q)| \\ &\leq (t+\varepsilon)|\Delta^*(p,q)|\end{aligned}$$

and therefore $\operatorname{odil}(p,q) \leq t+\varepsilon$.

Case 2: $p \in \mathcal{A}$ and $q \notin \mathcal{B}$ or $p \notin \mathcal{A}$ and $q \in \mathcal{B}$

We show the scenario in which $p \in \mathcal{A}$ and $q \notin \mathcal{B}$. The other case is analogous. Since q is not selected for B, it holds $|B| \geq 3$. Analogously to Case 1, we show that

$$\begin{aligned}|C_{\overrightarrow{G}}(p,q)| &\leq |C_{\overrightarrow{G}}(p,b)| + |C_{\overrightarrow{G}}(b,q)| \\ &\leq t(1+4/s)|\Delta^*(p,q)| + (t+\varepsilon)3/s|\Delta^*(p,q)| = \left(t + \frac{7t+3\varepsilon}{s}\right)|\Delta^*(p,q)| \\ &\leq (t+\varepsilon) \cdot |\Delta^*(p,q)|\end{aligned}$$

for $s = \frac{10t+6\varepsilon}{\varepsilon}$ and therefore $\operatorname{odil}(p,q) \leq t+\varepsilon$.

Case 3: $p \in \mathcal{A}$ and $q \in \mathcal{B}$

Since p and q are picked points, the algorithm adds an oriented cycle through p and q whose length is at most $t|\Delta^*(p,q)|$. Therefore, we have $\operatorname{odil}(p,q) \leq t \leq t+\varepsilon$.

For each of the m well-separated pairs, we consider up to four pairs of points. For each, we compute their shortest path with at most $c-1$ edges. By using the Bellman-Ford algorithm, this takes $\mathcal{O}(c(|P|+|E|))$ time, where $|E| = \mathcal{O}(n^2)$ is the number of edges. The input, a tournament on n points, requires $\mathcal{O}(n^2)$ space. Hence, a $(t+\varepsilon)$-spanner can be computed in $\mathcal{O}(mcn^2 + T(n))$ time and $\mathcal{O}(n^2)$ space. Each well-separated pair contributes $\mathcal{O}(c)$ edges resulting in $\mathcal{O}(cm)$ edges in total. $\square$

Note that the runtime of Algorithm 7.1 may depend on the construction time and size of the WSPD. For all metrics where a subquadratic-size WSPD is known, it can be constructed in $o(n^2)$ time. While linear-size WSPDs are known for Euclidean [30] and doubling metrics [84], it is open whether the unit-disk graph metric admits a WSPD with $\mathcal{O}(n)$ pairs [61].

7.3 Sparse Constant Dilation Spanners

In Section 4.3, we present an orientation of the complete graph on a given point set P such that for any $p, q \in P$ there is an oriented cycle through p and q whose length is at most $5/3|\Delta^*(p,q)|$ and which has at most four edges. Therefore, combining Theorems 4.5 and 7.2 leads to a sparse $(5/3+\varepsilon)$-spanner.

Corollary 7.3. *Let $(P, |\cdot|)$ be a metric space with n points that admits a WSPD with $\mathcal{O}(n)$ pairs constructible in $\mathcal{O}(n^3)$ time. Given a real number $0 < \varepsilon < 2$, an oriented $(5/3+\varepsilon)$-spanner for P with $\mathcal{O}(n)$ edges can be constructed in $\mathcal{O}(n^3)$ time and $\mathcal{O}(n^2)$ space.*

However, this algorithm is computationally expensive, as it relies on shortest-path queries in a complete graph. In the following, we present a spanner construction that achieves a significant reduction in runtime while increasing the dilation by at most 1/3. We construct a sparse oriented $(2+\varepsilon)$-spanner in $\mathcal{O}(n \log n)$ time by the following steps:

1. Compute the WSPD on the given point set.
2. For two points of each well-separated pair, approximate a minimum perimeter triangle containing those points with our algorithm based on approximate nearest neighbor queries that is presented in Section 3.2.
3. Orient the undirected graph, whose edge set consists of edges of approximated triangles, using the greedy algorithm presented in Section 4.2.

In Step 3, we use the greedy orientation algorithm, which provides a 2-approximation with respect to the input triangles (see Algorithm 4.1 and Lemma 4.2). The input consists of the $(1+\varepsilon)$-approximated triangles obtained in Step 2. Note that in this case, Lemma 4.2 guarantees $\mathrm{odil}_{\overrightarrow{G}}(p,q) \leq 2+2\varepsilon$ only for two points p, q whose minimum perimeter triangle is approximated in Step 2. For such a triangle Δ_{pqr}, the dilation $\mathrm{odil}_{\overrightarrow{G}}(p,r)$ and $\mathrm{odil}_{\overrightarrow{G}}(q,r)$ can be arbitrarily large.

Algorithm 7.2 is a detailed version of the above described spanner construction.

The algorithm and its analysis work similar to our $(t+\varepsilon)$-spanner construction (see Section 7.2), as both use the WSPD to obtain a suitable set of pairs of points to consider. The key difference is that Algorithm 7.1 requires a tournament as input and then extracts a sparse subgraph, while Algorithm 7.2 first constructs a sparse undirected graph and then orients it.

Now, we show that Algorithm 7.2 constructs a sparse oriented $(2+\varepsilon)$-spanner for point sets in a Euclidean space $\mathbb{R}^d$:

Algorithm 7.2 Sparse $(2+\varepsilon)$-Spanner.

Require: Set P of n points in $\mathbb{R}^d$, positive reals $\varepsilon < 2$, ε_1, s
Ensure: $(2+\varepsilon)$-spanner for P with $\mathcal{O}(n)$ edges
  Compute an s-WSPD $\{A_i, B_i\}$ for $1 \leq i \leq m = \mathcal{O}(n)$
  Compute the data structure on P from [10] to approximate nearest neighbors
  $L = \emptyset$
  for $i = 1$ **to** m **do**
    $\mathcal{A}_i$ = Pick $\min(|A_i|, 2)$ arbitrary points in A_i
    $\mathcal{B}_i$ = Pick $\min(|B_i|, 2)$ arbitrary points in B_i
    for each $ab \in \mathcal{A}_i \times \mathcal{B}_i$ **do**
      $\Delta_{abr'} = (1+\varepsilon_1)$-approximation of $\Delta^*(a,b)$ computed by Algorithm 3.1
      Add $\Delta_{abr'}$ to L
  return oriented graph generated by Algorithm 4.1

Theorem 7.4. *Given a set P of n points in $\mathbb{R}^d$ and a real number $0 < \varepsilon < 2$, an oriented $(2+\varepsilon)$-spanner for P with $\mathcal{O}(1/\varepsilon^d\ n)$ edges can be constructed in $\mathcal{O}(d/\varepsilon^d\ n \log n)$ time and $\mathcal{O}(1/\varepsilon^d\ n)$ space.*

Proof. Let the constants in Algorithm 7.2 be equal to $\varepsilon_1 = \varepsilon/4$ and $s = 72/\varepsilon$. Let $\overrightarrow{G}$ be the returned graph. We will prove that $|C_{\overrightarrow{G}}(p,q)| \leq (2+\varepsilon) \cdot |\Delta^*(p,q)|$ for any two distinct points $p, q \in P$. The proof is by induction on the rank of $|pq|$ in the sorted sequence of all $\binom{n}{2}$ pairwise distances.

If p, q is a closest pair, then $\{\{p\},\{q\}\}$ is a well-separated pair. Therefore, a $(1+\varepsilon_1)$-approximation of $\Delta^*(p,q)$ is added to the list L of triangles, that define the edge set of $\overrightarrow{G}$, and, due to Lemma 4.2 and $\varepsilon_1 = \varepsilon/4$, we have

$$|C_{\overrightarrow{G}}(p,q)| \leq 2 \cdot (1+\varepsilon_1)|\Delta^*(p,q)| \leq (2+\varepsilon)|\Delta^*(p,q)|.$$

Assume that p, q is not a closest pair. Let $\{A, B\}$ be the well-separated pair with $p \in A$ and $q \in B$. Let $\mathcal{A} \subseteq A$ and $\mathcal{B} \subseteq B$ be the sets selected by Algorithm 7.2 during processing the pair $\{A, B\}$. We distinguish cases based on whether p and/or q are picked.

Case 1: $p \notin \mathcal{A}$ and $q \notin \mathcal{B}$
This implies $|A| \geq 3$ and $|B| \geq 3$. Let $a \in \mathcal{A}$ and $b \in \mathcal{B}$ be picked. We will prove

$$|C_{\overrightarrow{G}}(p,q)| \leq |C_{\overrightarrow{G}}(p,a)| + |C_{\overrightarrow{G}}(a,b)| + |C_{\overrightarrow{G}}(b,q)| \leq (2+\varepsilon) \cdot |\Delta^*(p,q)|.$$

A sketch of this proof is visualized in Figure 7.1.

Due to WSPD-Property 1, it holds that $|pa| < |pq|$ and $|bq| < |pq|$. Thus, we apply induction to bound $|C_{\overrightarrow{G}}(p,a)|$ and $|C_{\overrightarrow{G}}(b,q)|$. Since a and b are picked points, due Lemma 4.2, $\overrightarrow{G}$ contains a closed walk through a and b whose length is at most $(2+2\varepsilon_1)|\Delta^*(a,b)|$. This gives us

$$|C_{\overrightarrow{G}}(p,q)| \leq (2+\varepsilon)|\Delta^*(p,a)| + (2+2\varepsilon_1)|\Delta^*(a,b)| + (2+\varepsilon)|\Delta^*(b,q)|.$$

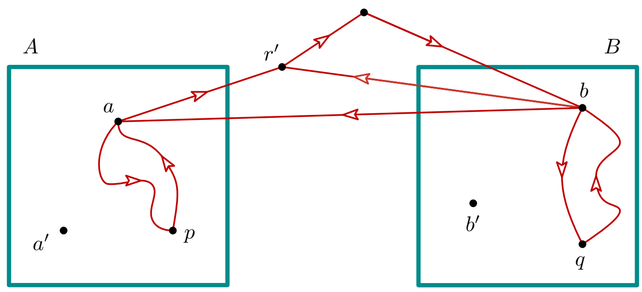


Figure 7.1: Visualization of Case 1 in the proof of Theorem 7.4: Let $\Delta_{abr'}$ be a $(1+\varepsilon_1)$-approximation of $\Delta^*(a,b)$.

Since $|A| \geq 3$, $|\Delta^*(p,a)|$ is at most the perimeter of any triangle with vertices p, a, and one other point in A. Each of the three edges of such a triangle can be bounded using WSPD-Property 1. It holds that

$$|\Delta^*(p,a)| \leq 3 \cdot 2/s \cdot |pq| \leq 3/s \cdot |\Delta^*(p,q)|, \tag{7.1}$$

where the last inequality $|\Delta^*(p,q)| \geq 2|pq|$ follows from the triangle inequality. Analogously, we bound $|\Delta^*(b,q)|$ by $3/s|\Delta^*(p,q)|$. By Lemma 3.3, we bound $|\Delta^*(a,b)| \leq (1+4/s)|\Delta^*(p,q)|$ and achieve in total

$$|C_{\overrightarrow{G}}(p,q)| \leq 2(2+\varepsilon) \cdot 3/s|\Delta^*(p,q)| + (2+2\varepsilon_1)(1+4/s)|\Delta^*(p,q)|.$$

Since $\varepsilon_1 = \varepsilon/4$, $\varepsilon < 2$ and $s = 72/\varepsilon$, we conclude that

$$\begin{aligned}|C_{\overrightarrow{G}}(p,q)| &\leq \left(2 + \frac{\varepsilon}{2} + \frac{20+8\varepsilon}{s}\right)|\Delta^*(p,q)| \\ &\leq (2+\varepsilon/2+36/s)\,|\Delta^*(p,q)| = (2+\varepsilon)|\Delta^*(p,q)|.\end{aligned}$$

Case 2: $p \in \mathcal{A}$ and $q \notin \mathcal{B}$ or $p \notin \mathcal{A}$ and $q \in \mathcal{B}$

We show the scenario in which $p \in \mathcal{A}$ and $q \notin \mathcal{B}$. The other case is analogous. W.l.o.g., let p be picked for A and a distinct point b be picked for B.

Analogously to Case 1, for $\varepsilon_1 = \varepsilon/4$, $\varepsilon < 2$ and $s = 72/\varepsilon$, we have

$$\begin{aligned}|C_{\overrightarrow{G}}(p,q)| &\leq |C_{\overrightarrow{G}}(p,b)| + |C_{\overrightarrow{G}}(b,q)| \\ &\leq (2+2\varepsilon_1)|\Delta^*(p,b)| + (2+\varepsilon)|\Delta^*(b,q)| \\ &\leq (2+2\varepsilon_1)\,(1+4/s)\,|\Delta^*(p,q)| + (2+\varepsilon)3/s|\Delta^*(p,q)| \\ &\leq (2+\varepsilon)|\Delta^*(p,q)|.\end{aligned}$$

Case 3: $p \in \mathcal{A}$ and $q \in \mathcal{B}$

An approximation of $\Delta^*(p,q)$ is added to L and then, due to Lemma 4.2 and $\varepsilon_1 = \varepsilon/4$, we have

$$|C_{\overrightarrow{G}}(p,q)| \leq 2 \cdot (1+\varepsilon_1)|\Delta^*(p,q)| \leq (2+\varepsilon)|\Delta^*(p,q)|.$$

The initialization is dominated by computing a WSPD in $\mathcal{O}(n\log n + s^d n)$ time and $\mathcal{O}(s^d n)$ space [30] (WSPD-Property 2). We use the data structure of Arya, Mount, Netanyahu, Silverman and Wu [10] to approximate nearest neighbors. Its construction takes $\mathcal{O}(dn \log n)$ time and it uses $\mathcal{O}(dn)$ space (see Lemma 2.1). For each of the $\mathcal{O}(s^d n)$ pairs, we consider at most four pairs of points and approximate their minimum perimeter triangles. Therefore, computing the list L of $\mathcal{O}(s^d n)$ triples costs $\mathcal{O}(s^d dn \log n)$ time (see Lemma 3.4) and orienting those costs $\mathcal{O}(s^d n \log n)$ time (Lemma 4.2). Therefore, a $(2+\varepsilon)$-spanner can be computed in $\mathcal{O}(s^d dn \log n)$ time and $\mathcal{O}(s^d n)$ space. □

7.4 Average Dilation Spanners

This section presents a construction of a sparse oriented spanner with average dilation $1+\varepsilon$ for sufficiently large point sets in a Euclidean space. Our algorithm consists of the following steps:

1. Compute the WSPD on the given point set.
2. For each singleton-singleton pair $\{\{p\},\{q\}\}$, approximate $\Delta^*(p,q)$ by our algorithm using approximate nearest neighbor queries presented in Section 3.2.
3. Orient the undirected graph whose edge set consists of all edges of the approximated triangles, using the greedy algorithm presented in Section 4.2.
4. Sort all well-separated pairs ascending by the distance of two representatives per pair.
5. For each non-singleton-singleton pair, we add at most five oriented edges.

Note that Algorithm 3.1 returns a $(1+\varepsilon)$-approximation of the minimum perimeter triangle of two query points. However, for this spanner construction a 2-approximation is sufficient and therefore used for simplification.

While Steps 1–3 follow Algorithm 7.2, the essential difference is that Step 2 was applied uniformly to all well-separated pairs, resulting in a worst-case (and average) dilation of $2+\varepsilon$. In contrast, the proposed algorithm treats singleton-singleton and non-singleton-singleton pairs separately.

For singleton-singleton pairs, a dilation of $1+\varepsilon$ is impossible once their minimum perimeter triangles overlap and their consistent orientations conflict. Even when using a $(1+\varepsilon)$-approximation for each triangle, such conflicts may occur. However, since they are only a small portion of all well-separated pairs, their effect on the average dilation is negligible, as long as their dilation is constant.

For non-singleton-singleton pairs, the edges oriented in Step 5 form a closed walk through points from both sets, ensuring a worst-case dilation of $1+\varepsilon$. The main challenge is preserving previously constructed closed walks: orienting an edge for one pair must not alter any walk belonging to previous pairs. This is ensured by our processing order (Step 4) together with structural properties of the WSPD.

Algorithm 7.3 provides a detailed version of the described construction.

To show that a spanner with average dilation $1+\varepsilon$ is returned, first we bound the worst-case dilation of any two distinct points p,q. Let $\{A_i,B_i\}$ be the unique pair such that $p \in A_i$ and $q \in B_i$ or $p \in B_i$ and $q \in A_i$. We bound the dilation of p

Algorithm 7.3 Sparse Average Dilation Spanner.

Require: Set P of n points in $\mathbb{R}^d$, sufficiently large real number s
Ensure: Oriented spanner on P with average dilation $1 + \mathcal{O}(1/s + s^d/n)$ and $\mathcal{O}(s^d n)$ edges
  Compute an s-WSPD $\{A_i, B_i\}$ for $1 \leq i \leq m$; assume $|A_i| \leq |B_i|$ for all i
  Compute the data structure on P from [10] to approximate nearest neighbors
  $L = \emptyset$
  for each singleton-singleton pairs $\{\{p\}, \{q\}\}$ **do**
    Δ_{pqr} = 2-approximation of $\Delta^*(p, q)$ by Algorithm 3.1
    Add the point triple $(p, q; r)$ to L
  Orient the edges of the triangles in L by the greedy orientation Algorithm 4.1
  Sort all well-separated pairs $\{A_i, B_i\}$ by ascending distance of any $a_i \in A_i$ and $b_i \in B_i$
  for each $\{A_i, B_i\}$ with $|B_i| \geq 2$ (in sorted order) **do**
    Choose two arbitrary points of B_i. Let b, b' be those points.
    Choose $\min(|A_i|, 2)$ arbitrary points of A_i. Let a (and a', if $|A_i| > 1$) be those points.
    Note: The following may change orientations of edges, set by the greedy algorithm.
    if $|B_i| = 2$ **then**
      if bb' is not oriented yet **then** Orient bb' arbitrarily
      Orient $\Delta_{abb'}$ consistently preserving the orientation of bb' (and adjusting the orientation of ab and ab' if necessary)
      if $|A_i| = 2$ **then** Orient $\Delta_{a'bb'}$ consistently preserving the orientation of bb'
    else
      Set the following orientations: (a, b), (b, a'), (a', b'), (b', a)

and q according to whether $\{A_i, B_i\}$ is a singleton-singleton pair or not. Later, we sum up the dilation for both types to bound the average.

Lemma 7.5. *Let p and q be two distinct points in P. Let $\overrightarrow{G}$ be the graph returned by Algorithm 7.3. For a sufficiently large number $s \geq 30$, it holds*

- $\operatorname{odil}(p, q) \leq 16$, *if $\{\{p\}, \{q\}\}$ is a well-separated pair, and*
- $\operatorname{odil}(p, q) = 1 + \mathcal{O}(1/s)$, *otherwise.*

Proof. Let $\{A_i, B_i\}$ be the well-separated pair with $p \in A_i$ and $q \in B_i$ or $p \in B_i$ and $q \in A_i$. W.l.o.g., let $p \in A_i$ and $q \in B_i$ and $|A_i| \leq |B_i|$. Our proof uses induction on the sorted order of the well-separated pairs.

Base case. The pair $\{A_1, B_1\}$ is a closest pair in P. Therefore, $\{A_1, B_1\}$ is the singleton-singleton pair $\{\{p\}, \{q\}\}$.

Let $\overrightarrow{G'} = (P, \overrightarrow{E'})$ be the oriented graph returned by Algorithm 4.1 (ran after the first for-loop). So, $\overrightarrow{G'}$ is an orientation of the undirected graph whose edge set consists of all edges of the approximated triangles of all singleton-singleton pairs of the WSPD. By construction, the shortest closed walk in $\overrightarrow{G'}$ containing p and q has at most four edges (due to Lemma 4.2). W.l.o.g., let

$$C_{\overrightarrow{G'}}(p,q) = p \to q \to x \to y \to p,$$

for $x, y \in P \setminus \{p,q\}$. Further, due to Lemma 4.2, we have

$$|C_{\overrightarrow{G'}}(p,q)| = |pq| + |qx| + |xy| + |yp| \leq 4|\Delta^*(p,q)|.$$

Since there is a unique pair $\{A_i, B_i\}$ such that $p \in A_i$ and $q \in B_i$, the edge pq is oriented by the first loop and is not changed later. But the orientation of edges qx, xy, and py may flip (at most once) in the second loop (see Figure 7.2).

Assume the orientation of xy is flipped during the second for-loop, i.e., $(x,y) \in \overrightarrow{E'}$ and $(y,x) \in \overrightarrow{E}$. We will show that $\overrightarrow{G}$ contains a path from x to y whose length is at most $(3 + 8/s)|xy|$.

Let $\{A_j, B_j\}$ be the well-separated pair such that $y \in A_j$ and $x \in B_j$ or $x \in A_j$ and $y \in B_j$. We assume $|A_j| \leq |B_j|$. If the orientation of xy is flipped, then both x and y must have been chosen while processing the pair $\{A_j, B_j\}$. Moreover, this can only happen during the second loop, implying that $|B_j| \geq 2$. W.l.o.g., let y be chosen for A_j, say $y = a$, and x be chosen for B_j, say $x = b$. Let b' be the other point chosen for B_j and a' for A_j, if $|A_j| > 1$.

If $|B_j| = 2$, then the triangle $\Delta_{xyb'}$ is consistently oriented in $\overrightarrow{G}$. Applying WSPD-Property 1, there is a path from x to y in $\overrightarrow{G}$ of length

$$|xb'| + |b'y| \leq 2/s|xy| + (1 + 4/s)|xy| = (1 + 6/s)|xy| \leq (3 + 8/s)|xy|.$$

If $|B_j| > 2$, then $(y,x), (x,a'), (a',b'), (b',y) \in \overrightarrow{E}$. Applying WSPD-Property 1, $\overrightarrow{G}$ contains a path from x to y of length

$$|xa'| + |a'b'| + |b'y| \leq (1 + 2/s)|xy| + (1 + 4/s)|xy| + (1 + 2/s)|xy| = (3 + 8/s)|xy|.$$

We conclude that, if $(x,y) \in \overrightarrow{E'}$ and $(y,x) \in \overrightarrow{E}$, then $\overrightarrow{G}$ contains a path from x to y whose length is at most $(3 + 8/s)|xy|$.

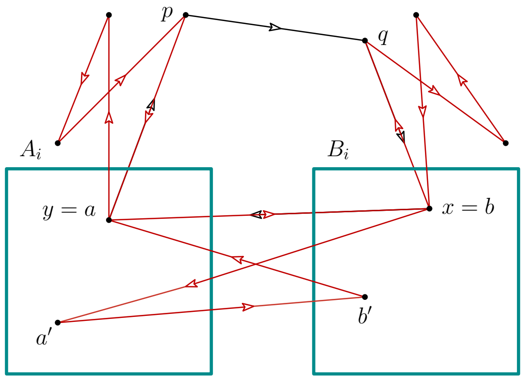


Figure 7.2: Illustration of the base case of Lemma 7.5.

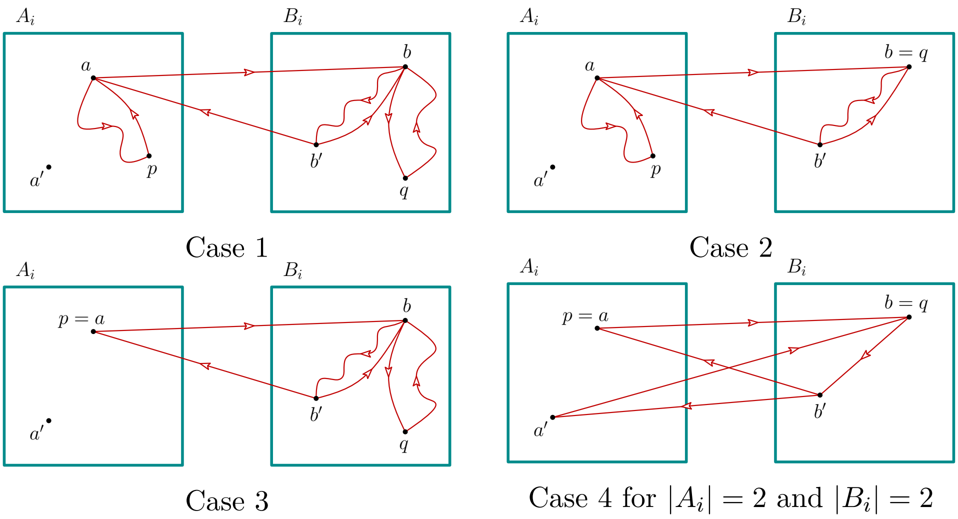


Figure 7.3: Illustration of cases in the inductions step of Lemma 7.5.

We proved the statement for the edge xy. The same argumentation applies for the edges qx and yp. Thus, a shortest closed walk in $\overrightarrow{G}$ through p and q has length

$$\begin{aligned}|C_{\overrightarrow{G}}(p,q)| &\le |pq| + (3+8/s)\cdot(|qx|+|xy|+|yp|) \le (3+8/s)|C_{\overrightarrow{G'}}(p,q)| \\ &\le (3+8/s)\cdot 4|\Delta^*(p,q)|.\end{aligned}$$

For $s \ge 8$, we have

$$\mathrm{odil}(p,q) \le (3+8/s)\cdot 4 \le 4\cdot 4 = 16.$$

Induction step. Let b (and b', if $|B_i| > 1$) be chosen for B_i and let a (and a', if $|A_i| > 1$) be chosen for A_i. Note that for $|B_i| \ge 2$, it holds either $(a,b),(b',a) \in \overrightarrow{E}$ or $(b,a),(a,b') \in \overrightarrow{E}$. (The latter is only possible for $|A_i| \le 2$.) Note that, since there is a unique well-separated pair $\{A_i, B_i\}$ with $p \in A_i$ and $q \in B_i$, the orientations of ab and $b'a$ are not changed in any later iteration. We will show that these edges lead to a closed walk in $\overrightarrow{G}$ through p and q. In particular, we show $|C_{\overrightarrow{G}}| \le (1+148/s)|\Delta^*(p,q)|$, which gives us $\mathrm{odil}(p,q) \le 1 + \mathcal{O}(1/s)$.
We distinguish cases based on whether p and/or q are chosen by the algorithm while processing the pair $\{A_i, B_i\}$ (compare to Figure 7.3).

Case 1: $p \notin \{a, a'\}$ and $q \notin \{b, b'\}$
In this case, it holds that $|A_i| \ge 3$ and $|B_i| \ge 3$. We will show

$$|C_{\overrightarrow{G}}(p,q)| \le |C_{\overrightarrow{G}}(p,a)| + |ab| + |C_{\overrightarrow{G}}(b,q)| + |C_{\overrightarrow{G}}(b,b')| + |b'a| \le (1+148/s)|\Delta^*(p,q)|.$$

Due to the order of the pairs (see WSPD-Property 3), we apply induction to bound $|C_{\overrightarrow{G}}(p,a)|$. Since $1 + 148/s \le 16$ for $s \ge 30$, we bound

$$|C_{\overrightarrow{G}}(p,a)| \le 16|\Delta^*(p,a)|$$

regardless whether p, a is a singleton-singleton pair or not. Analogously, we bound $|C_{\overrightarrow{G}}(b,q)|$ and $|C_{\overrightarrow{G}}(b,b')|$. We apply WSPD-Property 1 to bound $|ab|$ and $|b'a|$. So, we have

$$|C_{\overrightarrow{G}}(p,q)| \leq 16|\Delta^*(p,a)| + (1+4/s)|pq| + 16|\Delta^*(b,q)| + 16|\Delta^*(b,b')| + (1+4/s)|pq|.$$

Since $|A_i| \geq 3$, $|\Delta^*(p,a)|$ is at most the perimeter of any triangle with vertices p, a, and one other point in A_i. Each of the three edges of such a triangle can be bounded using WSPD-Property 1. It holds that

$$|\Delta^*(p,a)| \leq 3 \cdot 2/s \cdot |pq| \leq 3/s \cdot |\Delta^*(p,q)|,$$

where the last inequality $|\Delta^*(p,q)| \geq 2|pq|$ follows from the triangle inequality. Analogously, we bound $|\Delta^*(b,q)|$ and $|\Delta^*(b,b')|$ each by $3/s|\Delta^*(p,q)|$. We conclude

$$|C_{\overrightarrow{G}}(p,q)| \leq 3 \cdot 16 \cdot 3/s \cdot |\Delta^*(p,q)| + (1+4/s)|\Delta^*(p,q)| = (1+148/s)|\Delta^*(p,q)|.$$

Case 2: $p \notin \{a,a'\}$ and $q \in \{b,b'\}$

In this case, it holds that $|A_i| \geq 3$ and therefore $|B_i| \geq 3$. W.l.o.g., let $q = b$. We will show

$$|C_{\overrightarrow{G}}(p,q)| \leq |C_{\overrightarrow{G}}(p,a)| + |ab| + |C_{\overrightarrow{G}}(b,b')| + |b'a| \leq (1+148/s)|\Delta^*(p,q)|.$$

Since $|A_i| \geq 3$ and $|B_i| \geq 3$, we are in the set up as in Case 1. Therefore, length of $|C_{\overrightarrow{G}}(p,q)|$ in Case 1 is an upper bound for this case.

Case 3: $p \in \{a,a'\}$ and $q \notin \{b,b'\}$

In this case, it holds that $|A_i| \geq 1$ and $|B_i| \geq 3$. W.l.o.g., let $p = a$. We show

$$|C_{\overrightarrow{G}}(p,q)| \leq |ab| + |C_{\overrightarrow{G}}(b,q)| + |C_{\overrightarrow{G}}(b,b')| + |b'a| \leq (1+148/s)|\Delta^*(p,q)|.$$

Since $q \notin \{b,b'\}$, it holds $|B_i| \geq 3$. So, analogously to Case 1, we bound $|C_{\overrightarrow{G}}(b,q)|$ and $|C_{\overrightarrow{G}}(b,b')|$ each by $16 \cdot 3/s|\Delta^*(p,q)|$. We conclude

$$|C_{\overrightarrow{G}}(p,q)| \leq 2 \cdot 16 \cdot 3/s|\Delta^*(p,q)| + (1+4/s)|\Delta^*(p,q)| \leq (1+148/s)|\Delta^*(p,q)|.$$

Case 4: $p \in \{a,a'\}$ and $q \in \{b,b'\}$

In this case, it holds that $|A_i| \geq 1$ and $|B_i| \geq 1$. W.l.o.g., let $p = a$ and $q = b$.

If $|B_i| = 1$ and thus $|A_i| = 1$, then $\{\{p\},\{q\}\}$ is a singleton-singleton pair. Note that for the proof of the base case we only used the fact that $\{\{p\},\{q\}\}$ is a singleton-singleton pair (not that pq is the closest pair). Therefore, the same argumentation as in the base case applies here. Hence, it holds $\text{odil}(p,q) \leq 16$.

For $|B_i| = 2$, the triangle $\Delta_{abb'}$ is oriented consistently when processing the pair $\{A_i, B_i\}$. Note that, if bb' is oriented already, this does not change its orientation. Let $\{A_j, B_j\}$ be the pair such that $b \in A_j$ and $b' \in B_j$ or $b' \in A_j$ and $b \in B_j$. Due to the order of the pairs, it holds $j < i$ (WSPD-Property 3). Therefore, the orientation of bb' is not changed in any later iteration and thus $\Delta_{abb'}$, consistently oriented in $\overrightarrow{G}$, is a closed walk in $\overrightarrow{G}$ containing p and q. Applying WSPD-Property 1, we have

$$\begin{aligned} |C_{\overrightarrow{G}}(p,q)| &\le |\Delta_{abb'}| = |ab| + |bb'| + |b'a| \\ &\le (1+4/s)|pq| + 2/s|pq| + (1+2/s)|pq| = (2+8/s)|pq| \\ &\le (1+4/s)|\Delta^*(p,q)| \le (1+148/s)|\Delta^*(p,q)|. \end{aligned}$$

If $|B_i| \ge 3$, then, analogously to Case 3, we bound

$$\begin{aligned} |C_{\overrightarrow{G}}(p,q)| &\le |ab| + |C_{\overrightarrow{G}}(b,b')| + |b'a| \\ &\le 16 \cdot 3/s|\Delta^*(p,q)| + (1+4/s)|\Delta^*(p,q)| \le (1+148/s)|\Delta^*(p,q)|. \quad \square \end{aligned}$$

Theorem 7.6. *Given a set P of n points in $\mathbb{R}^d$, where d is a constant, and a sufficiently large real number $s \ge 30$, an oriented graph for P with average dilation $1 + \mathcal{O}(1/s + s^d/n)$ and $\mathcal{O}(s^d n)$ edges can be constructed in $\mathcal{O}(s^d n \log n)$ time using $\mathcal{O}(s^d n)$ space.*

Proof. Let $\overrightarrow{G}$ be the graph returned by Algorithm 7.3. Due to Lemma 7.5, we have $\mathrm{odil}_{\overrightarrow{G}}(p,q) \le 16$ for a singleton-singleton pair and $\mathrm{odil}_{\overrightarrow{G}}(p,q) \le 1 + \mathcal{O}(1/s)$ otherwise. Let $m = \mathcal{O}(s^d n)$ be the number of well-separated pairs. Let $m' \le m$ be the number of singleton-singleton pairs. It holds

$$\begin{aligned} \sum_{p \ne q} \mathrm{odil}(p,q) &\le 16m' + \left(\binom{n}{2} - m'\right)(1 + \mathcal{O}(1/s)) \le 16m + \binom{n}{2}(1 + \mathcal{O}(1/s)) \\ &= \mathcal{O}(s^d n) + (1 + \mathcal{O}(1/s))\binom{n}{2}. \end{aligned}$$

Therefore, the average dilation of $\overrightarrow{G}$ is

$$\frac{1}{\binom{n}{2}} \sum_{p \ne q} \mathrm{odil}(p,q) \le 1 + \mathcal{O}(1/s + s^d/n).$$

Computing a WSPD costs $\mathcal{O}(n \log n + s^d n)$ time and $\mathcal{O}(s^d n)$ space [30]. We use the data structure of Arya, Mount, Netanyahu, Silverman and Wu [10] to approximate nearest neighbors. Its construction takes $\mathcal{O}(dn \log n)$ time and it uses $\mathcal{O}(dn)$ space (see Lemma 2.1). For each of the $\mathcal{O}(s^d n)$ singleton-singleton pairs, we approximate $\Delta^*(p,q)$ in $\mathcal{O}(d \log n)$ time. Therefore, computing the list L of $\mathcal{O}(s^d n)$ triples costs $\mathcal{O}(s^d n \log n)$ time (Lemma 3.4) and orienting those costs $\mathcal{O}(s^d n \log n)$ time (Lemma 4.2). We sort the $\mathcal{O}(s^d n)$ pairs of the WSPD in $\mathcal{O}(s^d n \log n)$ time. For each of the $\mathcal{O}(s^d n)$ non-singleton-singleton pairs, we add oriented edges in $\mathcal{O}(1)$ time. Therefore, Algorithm 7.3 takes $\mathcal{O}(s^d n \log n)$ time and $\mathcal{O}(s^d n)$ space. $\square$

Depending on the choice of s, Theorem 7.6 offers different guarantees. For example, if ε is a constant and $s = \Theta(1/\varepsilon)$, the average dilation is $1 + \varepsilon + \mathcal{O}(1/n)$ and the number of edges is $\mathcal{O}(n)$. If $s = n^{1/(d+1)}$, the average dilation is $1 + \mathcal{O}(1/n^{1/(d+1)})$ and the number of edges is $\mathcal{O}(n^{(2d+1)/(d+1)}) = o(n^2)$. If $s = (\log \log n)^{1/d}$, the average dilation is $1 + \mathcal{O}(1/(\log \log n)^{1/d}) = 1 + o(n)$ and the number of edges is $\mathcal{O}(n \log \log n)$.

Chapter 8
Discussion

Our work initiated the study of oriented spanners. We established a new research direction and obtained insightful initial results, but several questions remain open. In this chapter, we outline directions for future research and highlight challenges for different problem variants.

Other dilation models. While undirected dilation compares the shortest path between two points p and q in a graph with their distance, oriented dilation compares the shortest closed walk through p and q in an oriented graph to the shortest cycle in the complete undirected graph K_n. An analogous measure for an undirected graph G, called *cyclic dilation*, would compare the shortest cycle in G to a shortest cycle in K_n. Can we efficiently compute sparse graphs of low cyclic dilation?

The dilation of an undirected spanner under the failure of k edges is similar to oriented dilation for $k = 1$. Our results on plane oriented spanners also yielded new insights into computing plane 1-edge fault-tolerant spanners. Also our reduction from minimum (undirected) dilation triangulation to minimum oriented triangulation carried over directly. It would be interesting to investigate whether other results on oriented spanners transfer to fault-tolerant spanners.

Another measure of interest would be *detour dilation*, which is close to fault-tolerance. It compares the shortest path in $G \setminus pq$ with the minimum perimeter triangle of p and q. Low detour dilation is a necessary condition for low oriented dilation as well as low 1-edge fault-tolerance, and actually at the core of our analysis of the greedy triangulation. Is there a triangulation with constant detour dilation? Can it be oriented to obtain constant oriented dilation?

Computing the dilation. We prove that computing the oriented dilation is APSP-hard for oriented graphs in a metric space. We complement this by a subcubic approximation algorithm whose runtime and approximation factor depend on approximating the length of a shortest path between two given points. Improving such approximations would also benefit our sparsification algorithm that yields a sparse $(5/3+\varepsilon)$-spanner. Hence, developing faster methods to approximate shortest paths in directed or oriented graphs is an interesting open problem.

Computing the oriented dilation of metric graphs seems to be dominated by computing the minimum perimeter triangle. In contrast, for Euclidean graphs, the minimum perimeter triangle of two given points can be computed in $o(n)$ time.

This raises the question: Is computing the oriented dilation of a Euclidean graph easier than for general metric graphs?

The hardness of computing the undirected dilation remains a long-standing open problem. Our hardness proof for oriented dilation is a reduction from MINIMUM-TRIANGLE, i.e., finding a minimum-weight cycle (which is a triangle) in an undirected graph. Since this reduction depends on the specific definition of oriented dilation, it cannot be adapted to undirected dilation, but the proof may extend to other dilation measures discussed above.

Bounding the dilation. We bounded the worst-case dilation t of the minimum spanner on any metric point set by $3/2 \leq t \leq 5/3$. Metric point sets of size 4 have a tight bound of 3/2. For points in a Euclidean space, the best known bounds are $2\sqrt{3}-2 \leq t \leq 5/3$. Also this lower bound follows from a point set of size 4; whether this bound is tight remains open. A first step to prove tightness would be showing that our 3/2-tournament is optimal for any metric point set of size 4. For both Euclidean and metric space, the question arises whether a larger point set leads to a minimum spanner with higher dilation.

We also analyzed the dilation of the minimum spanner for specific graph classes, such as graphs on one-dimensional point sets with a one- or two-page book embedding. We bounded the worst-case dilation t of the minimum one-page plane spanner by $1 + \Phi \leq t \leq 5$, where Φ denotes the golden ratio. For two-page plane spanners with a baseline, we have $\sqrt{2} \leq t \leq 2$.

Closing these gaps is work in progress. The search for worst-case examples using randomly generated point sets appears to be exhausted (see [AK5] for point sets in $\mathbb{R}$; moreover, an undergraduate thesis under our supervision analyzed minimum dilation tournaments on small point sets embedded in $\mathbb{R}^2$ and $\mathbb{R}^3$).

Plane spanners. We presented how to orient the greedy triangulation such that it yields an oriented $\mathcal{O}(n)$-spanner for arbitrary point sets in $\mathbb{R}^2$ and an oriented $\mathcal{O}(1)$-spanner for points in convex position. Furthermore, we observed a strong connection between oriented spanners and 1-edge fault-tolerant spanners. We proved that the greedy triangulation is a 1-edge fault-tolerant $\mathcal{O}(n)$-spanner for arbitrary and $\mathcal{O}(1)$-spanner for convex point sets. For both oriented and fault-tolerant spanners, an interesting question arises: Do other triangulations satisfying the α-diamond property also have linear dilation? We analyzed other prominent triangulations, such as the Delaunay and minimum dilation triangulation, and concluded that the α-diamond property is not sufficient for plane oriented and fault-tolerant spanners.

This raises the question of whether plane oriented or fault-tolerant spanners with constant dilation even exist on non-convex point sets. If constant dilation cannot be attained in general, what is the smallest dilation that can be guaranteed?

Sparse spanners. We constructed a sparse oriented $(2+\varepsilon)$-spanner. Our algorithm computes such a spanner efficiently in $\mathcal{O}(n \log n)$ time and $\mathcal{O}(n)$ space for n points in a Euclidean space of constant dimension. To obtain sparse oriented spanners with dilation less than 2, we showed how to sparsify a t-tournament while increasing the dilation to at most $t+\varepsilon$. The result is a $(5/3+\varepsilon)$-spanner in $\mathcal{O}(n^3)$ time using $\mathcal{O}(n^2)$ space. This makes computing minimum dilation tournaments even more relevant.

However, relying on tournaments inherently leads to algorithms that require quadratic space and at least quadratic running time. A natural question is whether this can be avoided. Can we compute a sparse oriented spanner with dilation less than 2 using subquadratic space? Or even using subquadratic (or for now, merely subcubic) running time?

Average dilation spanners. We presented an algorithm that computes a sparse oriented spanner with average dilation $1+\varepsilon$ for sufficiently large Euclidean point sets. Despite its practical relevance, average dilation remains largely unexplored compared to the traditional worst-case measure. We hope that our results will inspire further investigation into algorithms and constructions that achieve low average dilation, as well as a deeper theoretical understanding of how average performance relates to other spanner properties.

Besides computing oriented spanners with small average dilation, there are other intriguing problems related to average dilation. For example, as done in [33] for undirected trees, cycles, and plane graphs, an interesting open problem is to compute or approximate the average dilation of a given (oriented) graph.

Orienting undirected graphs. Several of our constructions, namely a 2-tournament, a plane $\mathcal{O}(n)$-spanner, a sparse $(2+\varepsilon)$-spanner and an oriented spanner with average dilation $1+\varepsilon$, rely on first computing a suitable undirected graph and then orienting it. For those applications, we presented an orientation algorithm that returns an oriented graph, whose edge set consists of a given set of triangles, contains for each input triangle a closed walk of length at most twice its perimeter.

All those applications can be improved by optimally orienting the underlying undirected graph. However, we dismiss this approach by proving that, given an undirected graph, finding its minimum dilation orientation is NP-hard, even for Euclidean graphs. The complexity of this problem when restricting the graph class remains open. In particular: Is the problem NP-hard for planar or complete graphs?

A promising direction for future research is to develop orientation algorithms that guarantee improved approximation factors. Our 5/3-tournament does not extend to these settings, because our current proof of its dilation critically depends on the orientation of a complete graph.

Generalization to metric spaces. Our $(5/3+\varepsilon)$-spanner construction works for point sets in any metric space admitting a WSPD. Such metrics include the Euclidean space [30], the unit-disk graph metric [50], and metric spaces with constant doubling dimension [84]. Our construction of a sparse $(2+\varepsilon)$-spanner and a spanner with oriented average dilation $1+\varepsilon$ only apply for points in the Euclidean space, since both rely on an approximation of the minimum perimeter triangle that is specifically designed for the Euclidean setting. Developing such an approximation for broader classes of metric spaces would improve both constructions[1].

[1]Since the submission of this thesis, we have designed such an approximation for spaces with constant doubling dimension, extending both constructions accordingly. See [AK1] for details.

Bounded-degree spanners. In communication networks, the number of connections per node might be limited. This motivates the study of *bounded-degree spanners*. For undirected spanners in general, a tight degree bound of 3 is known [38, 74]. For bounded-degree plane spanners, closing the gap between the best known upper bound of 4 [64] and the lower bound of 3 [16, 38] is an open problem (see [7] for an overview on constructions of bounded-degree plane spanners).

Similarly, one may study oriented spanners with bounded degree. Most known constructions of bounded-degree spanners are based on the Delaunay triangulation (see [19, 23, 24, 64] for examples). However, we presented a point set for which any orientation of its Delaunay triangulation has unbounded oriented dilation. As the Delaunay triangulation is equivalent to a variant of the half-Θ_6-graph [18], a natural starting point for constructing oriented bounded-degree spanners could be directed Θ-graphs [5, 20, 21]. This raises several questions: Which additional edges must be inserted into a directed Θ-graph with constant dilation to achieve small oriented dilation? How can bi-directed edges in such constructions be replaced?

Flexible constructions. Low dilation and a small number of edges are conflicting optimization objectives. Moreover, computing low or even minimum dilation spanners is computationally expensive. Thus, for undirected spanners, there are efficient algorithms providing a trade-off between the dilation t and the number of edges. The most prominent construction is the greedy spanner, which returns a sparse t-spanner for any given $t \geq 1$ [74]. In contrast, for oriented spanners only a few constructions are known, each guaranteeing a specific dilation. Therefore, it would be interesting to design constructions that provide a trade-off similar to the undirected case.

Light spanners. Another optimization goal is to compute small dilation spanners whose summed edge lengths, i.e., the *weight* of the spanner, is low. For undirected spanners, the weight is evaluated relative to the weight of the minimum spanning tree. Given a parameter t, computing the minimum weight t-spanner on a given Euclidean point set is NP-hard [31]. Among the various constructions of undirected light spanners [56, 57, 65, 68], the lightness of the greedy spanner stands out as the most well-established result [40, 74].

The equivalent for oriented spanners is to compare the weight of the spanner with the minimum length Hamiltonian cycle. The problem of computing the shortest cycle that visits all given points is the traveling salesperson problem, which is known to be NP-hard, even in the Euclidean plane [76]. This already makes the evaluation of the weight of an oriented spanner hard. However, since the length of the minimum Hamiltonian cycle is at most twice the weight of a minimum spanning tree [80], the same notion of lightness can be justified as in the undirected setting. With this definition, analogous to our approach for constructing plane spanners, a natural strategy is to first compute an undirected graph with low weight and subsequently orient its edges.

Dynamic spanners. An interesting extension of our work concerns dynamic models, where edges or points may be inserted or deleted over time.

For maintaining bounded dilation of undirected spanners under point updates, several fully dynamic spanner constructions are known [13, 14, 49, 54].

Related is edge augmentation, where one seeks a set of k edges whose addition minimizes the dilation of the resulting graph. For small k, several algorithms are known [47, 59, 87]. Edge deletion has been far less explored, existing results cover only restricted classes of undirected graphs [4]. Deleting edges naturally relates to fault-tolerant spanners, as both aim to preserve low dilation under structural loss.

Since adding k edges to an empty graph or deleting k edges from the complete graph is NP-hard for both undirected and oriented spanners, the augmentation and reduction problems are also NP-hard. Exploring their parameterized complexity appears promising.

Dynamic oriented spanners introduce additional challenges beyond those encountered in undirected settings: Updates affect not only the edge set but also edge orientations. Designing algorithms that maintain an oriented spanner during point or edge updates therefore requires efficient handling of both structural and directional changes. It remains open whether techniques from dynamic undirected spanners can be adapted to this setting or whether fundamentally new approaches are needed.

Bi-directed edges. In many applications, it is desirable to reduce the number of bi-directional edges, although avoiding them entirely may not be necessary. This opens up a range of questions concerning the trade-off between dilation and the number of bi-directed edges.

One natural problem is as follows: Given a parameter t, compute the directed t-spanner that minimizes the number of bi-directed edges (without restricting the number of oriented edges). A related parameterized variant allows at most k bi-directed edges. A natural approach would be to first construct an oriented spanner and then selectively bi-direct k edges. Does this lead to small dilation spanners with such restrictions?

Similarly, given an oriented spanner, how many edges must be made bi-directional to decrease its oriented dilation? Since not every point set admits an oriented 1-spanner, it would also be insightful to ask how many edges of a tournament need to be bi-directed to achieve a 1-spanner. Proving upper and lower bounds for these trade-offs could provide valuable theoretical insights.

A further variant motivated by practical settings considers graphs with certain one-way edges fixed in advance: Given a graph where some edges are marked as "one-way", compute an orientation of these edges that minimizes the dilation while bi-directing all other edges. For which graph classes can this be done efficiently? For which is this problem NP-hard?

Bibliography

Own Publications

[AK1] S. Bhore, A. Biniaz, K. Buchin, A. Kalb, A. Maheshwari, S. Odak, M. Smid, C. Rehs, and J.-L. De Carufel. “Sparse Oriented Spanners in Metric Spaces”. In: *Proc. 34th Annu. European Sympos. Algorithms (ESA)*. Proceedings not published yet. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2026.

[AK2] P. Bose, K. Buchin, and A. Kalb. “Fault-Tolerance and Oriented Dilation of the Greedy Triangulation”. In: *Proc. 42nd European Workshop on Comput. Geom. (EuroCG)*. 2026.

[AK3] K. Buchin, J. Conradi, S. Har-Peled, A. Kalb, A. Lahiri, L. Plätz, C. Rehs, and S. Wong. “On Small Pair Decompositions for Point Sets”. In: *Proc. 34th Annu. European Sympos. Algorithms (ESA)*. Proceedings not published yet. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2026. arXiv: `2601.22728 [cs.CG]`.

[AK4] K. Buchin, J. Gudmundsson, A. Kalb, A. Popov, C. Rehs, A. van Renssen, and S. Wong. “Oriented Spanners”. In: *Algorithmica* 88.1 (2026), 14:1–14:30. DOI: `10.1007/S00453-025-01342-8`. Also at ESA 2023.

[AK5] K. Buchin, A. Kalb, G. Li, and C. Rehs. “Experimental analysis of oriented spanners on one-dimensional point sets”. In: *Proc. 36th Canad. Conf. Comput. Geom. (CCCG)*. 2024, pp. 223–231.

[AK6] K. Buchin, A. Kalb, A. Maheshwari, S. Odak, C. Rehs, M. Smid, and S. Wong. “Computing Oriented Spanners and their Dilation”. In: *Proc. 41st Internat. Sympos. Comput. Geom. (SOCG)*. LIPIcs. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025, 27:1–27:17. DOI: `10.4230/LIPIcs.SoCG.2025.27`.

[AK7] K. Buchin, A. Kalb, C. Rehs, and A. Schulz. “Oriented dilation of undirected graphs”. In: *Proc. 30th European Workshop on Comput. Geom. (EuroCG)*. 2024, 65:1–65:8.

[AK8] S. Cabello, D. Garijo, A. Kalb, F. Klute, I. Parada, and R. I. Silveira. “Algorithms for Distance Problems in Continuous Graphs”. In: *Proc. 19th Internat. Sympos. Algorithms and Data Struct. (WADS)*. Vol. 349. LIPIcs. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025, 13:1–13:14. DOI: `10.4230/LIPIcs.WADS.2025.13`.

References

[1] M. A. Abam, M. de Berg, M. Farshi, and J. Gudmundsson. "Region-Fault Tolerant Geometric Spanners". In: *Discrete Comput. Geom.* 41.4 (2009), pp. 556–582. DOI: 10.1007/S00454-009-9137-7. Also at SODA 2007.

[2] M. A. Abam, M. de Berg, and M. J. R. Seraji. "Geodesic Spanners for Points on a Polyhedral Terrain". In: *SIAM J. Comput.* 48.6 (2019), pp. 1796–1810. DOI: 10.1137/18M119358X. Also at SODA 2017.

[3] P. K. Agarwal and J. Matoušek. "On Range Searching with Semialgebraic Sets". In: *Discrete Comput. Geom.* 11 (1994), pp. 393–418. DOI: 10.1007/BF02574015.

[4] H. Ahn, M. Farshi, C. Knauer, M. Smid, and Y. Wang. "Dilation-Optimal Edge Deletion in Polygonal Cycles". In: *Internat. J. Comput. Geom. Appl.* 20.1 (2010), pp. 69–87. DOI: 10.1142/S0218195910003207. Also at ISAAC 2007.

[5] H. A. Akitaya, A. Biniaz, and P. Bose. "On the spanning and routing ratios of the directed Θ_6-graph". In: *Comput. Geom. Theory Appl.* 105-106 (2022). article no. 101881. DOI: 10.1016/J.COMGEO.2022.101881. Also at WADS 2021.

[6] I. Althöfer, G. Das, D. P. Dobkin, D. Joseph, and J. Soares. "On sparse spanners of weighted graphs". In: *Discrete Comput. Geom.* 9.1 (1993), pp. 81–100. DOI: 10.1007/BF02189308. Also at SWAT 1990.

[7] F. Anderson, A. Ghosh, M. Graham, L. Mougeot, and D. Wisnosky. "Bounded-Degree Plane Geometric Spanners in Practice". In: *ACM J. Exp. Algorithmics* 28 (2023), 1:1–1:36. DOI: 10.1145/3582497.

[8] S. R. Arikati, D. Z. Chen, L. P. Chew, G. Das, M. Smid, and C. D. Zaroliagis. "Planar Spanners and Approximate Shortest Path Queries among Obstacles in the Plane". In: *Proc. 4th Annu. European Sympos. Algorithms (ESA).* Vol. 1136. Lecture Notes in Computer Science. Springer-Verlag, 1996, pp. 514–528. DOI: 10.1007/3-540-61680-2_79.

[9] B. Aronov, K. Buchin, M. Buchin, B. M. P. Jansen, T. de Jong, M. J. van Kreveld, M. Löffler, J. Luo, R. I. Silveira, and B. Speckmann. "Connect the dot: Computing feed-links for network extension". In: *J. Spatial Information Science* 3 (2011), pp. 3–31. DOI: 10.5311/JOSIS.2011.3.47. Also at SIGSPATIAL 2008, WADS 2009.

[10] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu. "An optimal algorithm for approximate nearest neighbor searching in fixed dimensions". In: *J. ACM* 45.6 (1998), pp. 891–923. DOI: 10.1145/293347.293348. Also at SODA 1994.

[11] M. A. Bekos, M. Gronemann, and C. N. Raftopoulou. "Two-Page Book Embeddings of 4-Planar Graphs". In: *Algorithmica* 75.1 (2016), pp. 158–185. DOI: 10.1007/s00453-015-0016-8. Also at STACS 2014.

[12] S. de Berg, F. Staals, and M. J. van Kreveld. "The complexity of geodesic spanners". In: *J. Comput. Geom.* 15.1 (2024), pp. 21–65. DOI: 10.20382/JOCG.V15I1A2. Also at SOCG 2023.

[13] S. Bhore, A. Filtser, H. Khodabandeh, and C. D. Tóth. "Online Spanners in Metric Spaces". In: *SIAM J. Discrete Math.* 38.1 (2024), pp. 1030–1056. DOI: 10.1137/22M1534572. Also at ESA 2022.

[14] S. Bhore and C. D. Tóth. "Online Euclidean Spanners". In: *ACM Trans. Algorithms* 21.1 (2025), 5:1–5:22. DOI: 10.1145/3681790. Also at ESA 2021.

[15] A. Biniaz, M. Amani, A. Maheshwari, M. Smid, P. Bose, and J. D. Carufel. "A plane 1.88-spanner for points in convex position". In: *J. Comput. Geom.* 7.1 (2016), pp. 520–539. Also at SWAT 2016.

[16] A. Biniaz, P. Bose, J.-L. De Carufel, C. Gavoille, A. Maheshwari, and M. Smid. "Towards plane spanners of degree 3". In: *J. Comput. Geom.* 8.1 (2017), pp. 11–31. DOI: 10.20382/JOCG.V8I1A2. Also at ISAAC 2016.

[17] G. Bodwin, B. Haeupler, and M. Parter. "Fault-Tolerant Spanners against Bounded-Degree Edge Failures: Linearly More Faults, Almost For Free". In: *Annu. ACM-SIAM Sympos. Discrete Algorithms (SODA).* 2024, pp. 2609–2642. DOI: 10.1137/1.9781611977912.93.

[18] N. Bonichon, C. Gavoille, N. Hanusse, and D. Ilcinkas. "Connections between Theta-Graphs, Delaunay Triangulations, and Orthogonal Surfaces". In: *Proc. of the 36th Int. Conference on Graph-Theoretic Concepts in Computer Science (WG).* 2010, pp. 266–278. DOI: 10.1007/978-3-642-16926-7_25.

[19] N. Bonichon, C. Gavoille, N. Hanusse, and L. Perkovic. "Plane Spanners of Maximum Degree Six". In: *Proc. 37th Internat. Colloq. Automata Lang. Program. (ICALP).* Vol. 6198. Lecture Notes in Computer Science. Springer-Verlag, 2010, pp. 19–30. DOI: 10.1007/978-3-642-14165-2_3.

[20] P. Bose, J. D. Carufel, D. Hill, and M. Smid. "On the Spanning and Routing Ratio of the Directed Theta-Four Graph". In: *Discrete Comput. Geom.* 71.3 (2024), pp. 872–892. DOI: 10.1007/S00454-023-00597-8. Also at SODA 2019.

[21] P. Bose, J.-L. De Carufel, P. Morin, A. van Renssen, and S. Verdonschot. "Towards tight bounds on theta-graphs: More is not always better". In: *Theoret. Comput. Sci.* 616 (2016), pp. 70–93. DOI: 10.1016/j.tcs.2015.12.017.

[22] P. Bose, V. Dujmović, P. Morin, and M. H. M. Smid. "Robust Geometric Spanners". In: *SIAM J. Comput.* 42.4 (2013), pp. 1720–1736. DOI: 10.1137/120874473. SOCG 2013.

[23] P. Bose, J. Gudmundsson, and M. H. M. Smid. "Constructing Plane Spanners of Bounded Degree and Low Weight". In: *Algorithmica* 42.3–4 (2005), pp. 249–264. DOI: 10.1007/s00453-005-1168-8. Also at ESA 2002.

[24] P. Bose, D. Hill, and M. Smid. "Improved Spanning Ratio for Low Degree Plane Spanners". In: *Algorithmica* 80.3 (2018), pp. 935–976. DOI: 10.1007/S00453-017-0305-5. Also at LATIN 2016.

[25] P. Bose, A. Lee, and M. Smid. "On Generalized Diamond Spanners". In: *Proc. 10th Internat. Sympos. Algorithms and Data Struct. (WADS).* Vol. 4619. Lecture Notes in Computer Science. Springer-Verlag, 2007, pp. 325–336. DOI: 10.1007/978-3-540-73951-7_29.

[26] P. Bose and M. Smid. "On Plane Geometric Spanners: A Survey and Open Problems". In: *Comput. Geom. Theory Appl.* 46.7 (2013), pp. 818–830. DOI: 10.1016/j.comgeo.2013.04.002.

[27] A. F. Brandt, M. M. Gaiowski, C. C. de Souza, and P. J. de Rezende. "Minimum Dilation Triangulation: Reaching Optimality Efficiently". In: *Proc. 26th Canad. Conf. Comput. Geom. (CCCG).* 2014.

[28] K. Buchin, S. Har-Peled, and D. Oláh. "A Spanner for the Day After". In: *Discrete Comput. Geom.* 64.4 (2020), pp. 1167–1191. DOI: 10.1007/s00454-020-00228-6. Also at SOCG 2019.

[29] M. Burkhart, P. Von Rickenbach, R. Wattenhofer, and A. Zollinger. "Does topology control reduce interference?" In: *Proc. 5th ACM Internat. Sympos. Mobile Ad Hoc Networking and Computing (MobiHoc).* 2004, pp. 9–19. DOI: 10.1145/989459.989462.

[30] P. B. Callahan and S. R. Kosaraju. "A decomposition of multidimensional point sets with applications to k-nearest-neighbors and n-body potential fields". In: *J. ACM* 42.1 (1995), pp. 67–90. DOI: 10.1145/200836.200853.

[31] P. Carmi and L. Chaitman-Yerushalmi. "Minimum weight Euclidean t-spanner is NP-hard". In: *J. Discrete Algorithms* 22 (2013), pp. 30–42. DOI: 10.1016/j.jda.2013.06.010.

[32] T. M. Chan. "Optimal Partition Trees". In: *Discrete Comput. Geom.* 47.4 (2012), pp. 661–690. DOI: 10.1007/S00454-012-9410-Z. Also at SOCG 2010.

[33] S. Cheng, C. Knauer, S. Langerman, and M. Smid. "Approximating the average stretch factor of geometric graphs". In: *J. Comput. Geom.* 3.1 (2012), pp. 132–153. DOI: 10.20382/JOCG.V3I1A7. Also at ISAAC 2010.

[34] O. Cheong, H. Haverkort, and M. Lee. "Computing a minimum-dilation spanning tree is NP-hard". In: *Comput. Geom. Theory Appl.* 41.3 (2008), pp. 188–205. DOI: 10.1016/j.comgeo.2007.12.001. Also at CATS 2007.

[35] F. R. K. Chung, F. T. Leighton, and A. L. Rosenberg. "Embedding Graphs in Books: A Layout Problem with Applications to VLSI Design". In: *SIAM J. Algebraic Discrete Methods* 8.1 (1987), pp. 33–58. DOI: 10.1137/0608002.

[36] L. J. Cowen and C. G. Wagner. "Compact Roundtrip Routing for Digraphs". In: *Proc. 10th Annu. ACM-SIAM Sympos. Discrete Algorithms (SODA).* SIAM, 1999, pp. 885–886. DOI: 10.5555/314500.315068.

[37] L. J. Cowen and C. G. Wagner. “Compact roundtrip routing in directed networks”. In: *J. Algorithms* 50.1 (2004), pp. 79–95. DOI: 10.1016/j.jalgor.2003.08.001.

[38] G. Das and P. J. Heffernan. “Constructing Degree-3 Spanners with Other Sparseness Properties”. In: *Int. J. Found. Comput. Sci.* 7.2 (1996), pp. 121–136. DOI: 10.1142/S0129054196000105. Also at ISAAC 1993.

[39] G. Das and D. Joseph. “Which Triangulations Approximate the Complete Graph?” In: *Internat. Sympos. on Optimal Algorithms (OA)*. Vol. 401. Lecture Notes in Computer Science. Springer-Verlag, 1989, pp. 168–192. DOI: 10.1007/3-540-51859-2_15.

[40] G. Das and G. Narasimhan. “A Fast Algorithm for Constructing Sparse Euclidean Spanners”. In: *Internat. J. Comput. Geom. Appl.* 7.4 (1997), pp. 297–315. DOI: 10.1142/S0218195997000193. Also at SOCG 1994.

[41] A. Dobson and K. E. Bekris. “Sparse roadmap spanners for asymptotically near-optimal motion planning”. In: *Internat. J. Robotics Research* 33.1 (2014), pp. 18–47. DOI: 10.1177/0278364913498.

[42] M. Dory, S. Forster, Y. Kirkpatrick, Y. Nazari, V. Vassilevska Williams, and T. de Vos. “Fast 2-Approximate All-Pairs Shortest Paths”. In: *Annu. ACM-SIAM Sympos. Discrete Algorithms (SODA)*. SIAM, 2024, pp. 4728–4757. DOI: 10.1137/1.9781611977912.169.

[43] R. Drysdale III, S. A. McElfresh, and J. Snoeyink. “On exclusion regions for optimal triangulations”. In: *Discrete Appl. Math.* 109.1-2 (2001), pp. 49–65. DOI: 10.1016/S0166-218X(00)00236-5.

[44] V. Dujmović, P. Morin, and M. Smid. “Average Stretch Factor: How Low Does It Go?” In: *Discrete Comput. Geom.* 53.2 (2015), pp. 296–326. DOI: 10.1007/s00454-015-9663-4.

[45] V. Dujmović and D. R. Wood. “On Linear Layouts of Graphs”. In: *Discrete Math. Theor. Comput. Sci.* 6.2 (2004), pp. 339–358. DOI: 10.46298/dmtcs.317.

[46] D. Eppstein. “Spanning Trees and Spanners”. In: *Handbook of Computational Geometry*. Elsevier, 2000, pp. 425–461. DOI: 10.1016/B978-044482537-7/50010-3.

[47] M. Farshi, P. Giannopoulos, and J. Gudmundsson. “Improving the Stretch Factor of a Geometric Network by Edge Augmentation”. In: *SIAM J. Comput.* 38.1 (2008), pp. 226–240. DOI: 10.1137/050635675.

[48] R. W. Floyd. “Algorithm 97: Shortest path”. In: *Communications of the ACM* 5 (1962), pp. 345–345. DOI: 10.1145/367766.368168.

[49] J. Gao, L. J. Guibas, and A. T. Nguyen. “Deformable spanners and applications”. In: *Comput. Geom. Theory Appl.* 35.1-2 (2006), pp. 2–19. DOI: 10.1016/J.COMGEO.2005.10.001. Also at SOCG 2004.

[50] J. Gao and L. Zhang. “Well-Separated Pair Decomposition for the Unit-Disk Graph Metric and Its Applications”. In: *SIAM J. Comput.* 35.1 (2005), pp. 151–169. DOI: 10.1137/S0097539703436357. Also at STOC 2003.

[51] P. Giannopoulos, R. Klein, C. Knauer, M. Kutz, and D. Marx. “Computing Geometric Minimum-Dilation Graphs is NP-Hard”. In: *Internat. J. Comput. Geom. Appl.* 20.2 (2010), pp. 147–173. DOI: 10.1142/S0218195910003244. Also at GD 2006.

[52] M. Godau. “On the complexity of measuring the similarity between geometric objects in higher dimensions”. Dissertation. Free University Berlin, 1999. DOI: 10.17169/refubium-7780.

[53] A. V. Goldberg and C. Harrelson. “Computing the shortest path: A search meets graph theory”. In: *Annu. ACM-SIAM Sympos. Discrete Algorithms (SODA).* 2005, pp. 156–165. DOI: 10.5555/1070432.1070455.

[54] L. Gottlieb and L. Roditty. “An Optimal Dynamic Spanner for Doubling Metric Spaces”. In: *Proc. 16th Annu. European Sympos. Algorithms (ESA).* Vol. 5193. Lecture Notes in Computer Science. Springer-Verlag, 2008, pp. 478–489. DOI: 10.1007/978-3-540-87744-8_40.

[55] J. Gudmundsson and C. Knauer. “Dilation and detours in geometric networks”. In: *Handbook of Approximation Algorithms and Metaheuristics.* Chapman and Hall/CRC, 2018, pp. 53–69. DOI: 10.1201/9781420010749-62.

[56] J. Gudmundsson, C. Levcopoulos, and G. Narasimhan. “Fast Greedy Algorithms for Constructing Sparse Geometric Spanners”. In: *SIAM J. Comput.* 31.5 (2002), pp. 1479–1500. DOI: 10.1137/S0097539700382947.

[57] J. Gudmundsson, C. Levcopoulos, and G. Narasimhan. “Improved Greedy Algorithms for Constructing Sparse Geometric Spanners”. In: *Proc. 7th Scandinavian Sympos. on Algorithm Theory (SWAT).* Lecture Notes in Computer Science. Springer-Verlag, 2000, pp. 314–328.

[58] J. Gudmundsson and M. Smid. “On Spanners of Geometric Graphs”. In: *Int. J. Found. Comput. Sci.* 20.1 (2009), pp. 135–149. DOI: 10.1142/S0129054109006486. Also at SWAT 2006.

[59] J. Gudmundsson and S. Wong. “Improving the Dilation of a Metric Graph by Adding Edges”. In: *ACM Trans. Algorithms* 18.3 (2022), 20:1–20:20. DOI: 10.1145/3517807. Also at SODA 2021.

[60] S. Har-Peled and M. Mendel. “Fast Construction of Nets in Low-Dimensional Metrics and Their Applications”. In: *SIAM J. Comput.* 35.5 (2006), pp. 1148–1184. DOI: 10.1137/S0097539704446281. Also at SOCG 2005.

[61] S. Har-Peled, B. Raichel, and E. W. Robson. “Well-Separated Pairs Decomposition Revisited”. In: *CoRR* 2509.05997 (2025). arXiv: 2509.05997 [cs.CG].

[62] M. R. Henzinger, P. Klein, S. Rao, and S. Subramanian. “Faster Shortest-Path Algorithms for Planar Graphs”. In: *J. Comput. System Sci.* 55.1 (1997), pp. 3–23. DOI: https://doi.org/10.1006/jcss.1997.1493. Also at STOC 1994.

[63] A. Itai, C. H. Papadimitriou, and J. L. Szwarcfiter. "Hamilton Paths in Grid Graphs". In: *SIAM J. Comput.* 11.4 (1982), pp. 676–686. DOI: 10.1137/0211056.

[64] I. A. Kanj, L. Perkovic, and D. Türkoglu. "Degree four plane spanners: Simpler and better". In: *J. Comput. Geom.* 8.2 (2017), pp. 3–31. DOI: 10.20382/JOCG.V8I2A2. Also at SOCG 2016.

[65] I. A. Kanj, L. Perkovic, and G. Xia. "On Spanners and Lightweight Spanners of Geometric Graphs". In: *SIAM J. Comput.* 39.6 (2010), pp. 2132–2161. DOI: 10.1137/080737708.

[66] J. M. Keil and C. A. Gutwin. "Classes of Graphs Which Approximate the Complete Euclidean Graph". In: *Discrete Comput. Geom.* 7 (1992), pp. 13–28. DOI: 10.1007/BF02187821.

[67] C. Levcopoulos and A. Lingas. "Fast Algorithms for Greedy Triangulation". In: *BIT Numerical Mathematics* 32.2 (1992), pp. 280–296. DOI: 10.1007/BF01994882. Also at SWAT 1990.

[68] C. Levcopoulos and A. Lingas. "There Are Planar Graphs Almost as Good as the Complete Graphs and Almost as Cheap as Minimum Spanning Trees". In: *Algorithmica* 8.3 (1992), pp. 251–256. DOI: 10.1007/BF01758846.

[69] C. Levcopoulos, G. Narasimhan, and M. Smid. "Efficient Algorithms for Constructing Fault-Tolerant Geometric Spanners". In: *Proc. 30th Annu. ACM Sympos. Theory Comput. (STOC)*. 1998, pp. 186–195. DOI: 10.1145/276698.276734.

[70] C. Levcopoulos, G. Narasimhan, and M. Smid. "Improved Algorithms for Constructing Fault-Tolerant Spanners". In: *Algorithmica* 32.1 (2002), pp. 144–156. DOI: 10.1007/S00453-001-0075-X.

[71] D. Lichtenstein. "Planar Formulae and Their Uses". In: *SIAM J. Comput.* 11.2 (1982), pp. 329–343. DOI: 10.1137/0211025.

[72] T. Lukovszki. "New Results of Fault Tolerant Geometric Spanners". In: *Proc. 6th Internat. Sympos. Algorithms and Data Struct. (WADS)*. Vol. 1663. Lecture Notes in Computer Science. Springer-Verlag, 1999, pp. 193–204. DOI: 10.1007/3-540-48447-7_20.

[73] N. Megiddo. "Applying Parallel Computation Algorithms in the Design of Serial Algorithms". In: *J. ACM* 30.4 (1983), pp. 852–865. DOI: 10.1145/2157.322410. Also at FOCS 1981.

[74] G. Narasimhan and M. Smid. *Geometric Spanner Networks*. Cambridge University Press, 2007. DOI: 10.1017/CBO9780511546884.

[75] J. O'Rourke. "Advanced Problem 6369". In: *Amer. Math. Monthly* 88.10 (1981), p. 769.

[76] C. H. Papadimitriou. "The Euclidean travelling salesman problem is NP-complete". In: *Theoret. Comput. Sci.* 4.3 (1977), pp. 237–244. DOI: 10.1016/0304-3975(77)90012-3.

[77] M. Parter, A. Petruschk, S. Sapir, and E. Tzalik. “Parks and Recreation: Color Fault-Tolerant Spanners Made Local”. In: *Annu. ACM-SIAM Sympos. Discrete Algorithms (SODA)*. 2025, pp. 4061–4094. DOI: 10.1137/1.9781611978322.139.

[78] D. Peleg and A. Schäffer. “Graph Spanners”. In: *J. Graph Theory* 13.1 (1989), pp. 99–116. DOI: 10.1002/jgt.3190130114.

[79] K. R. U. K. Reddy. “A survey of the all-pairs shortest paths problem and its variants in graphs”. In: *Acta Univ. Sapientiae Inform* 8 (2016), pp. 16–40. DOI: 10.1515/ausi-2016-0002.

[80] D. J. Rosenkrantz, R. E. Stearns, and P. M. Lewis, II. “An Analysis of Several Heuristics for the Traveling Salesman Problem”. In: *SIAM J. Comput.* 6.3 (1977), pp. 563–581. DOI: 10.1137/0206041.

[81] B. Saha and C. Ye. “Faster Approximate All Pairs Shortest Paths”. In: *Annu. ACM-SIAM Sympos. Discrete Algorithms (SODA)*. 2024, pp. 4758–4827. DOI: 10.1137/1.9781611977912.169.

[82] C. Schindelhauer, K. Volbert, and M. Ziegler. “Geometric spanners with applications in wireless networks”. In: *Comput. Geom. Theory Appl.* 36.3 (2007), pp. 197–214. DOI: 10.1016/j.comgeo.2006.02.001. Also at ISAAC 2004.

[83] M. Smid. “The Well-Separated Pair Decomposition and Its Applications”. In: *Handbook of Approximation Algorithms and Metaheuristics*. Chapman and Hall/CRC, 2018.

[84] K. Talwar. “Bypassing the embedding: algorithms for low dimensional metrics”. In: *Proc. 36th Annu. ACM Sympos. Theory Comput. (STOC)*. ACM, 2004, pp. 281–290. DOI: 10.1145/1007352.1007399.

[85] V. Vassilevska Williams and R. R. Williams. “Subcubic Equivalences Between Path, Matrix, and Triangle Problems”. In: *J. ACM* 65.5 (2018). article no. 27. DOI: 10.1145/3186893. Also at FOCS 2010.

[86] S. Warshall. “A Theorem on Boolean Matrices”. In: *J. ACM* 9.1 (1962), pp. 11–12. DOI: 10.1145/321105.321107.

[87] C. Wulff-Nilsen. “Computing the dilation of edge-augmented graphs in metric spaces”. In: *Comput. Geom. Theory Appl.* 43.2 (2010), pp. 68–72. DOI: 10.1016/J.COMGEO.2009.03.008.

[88] G. Xia. “The Stretch Factor of the Delaunay Triangulation Is Less than 1.998”. In: *SIAM J. Comput.* 42.4 (2013), pp. 1620–1659. DOI: 10.1137/110832458. Also at SOCG 2011.

[89] M. Yannakakis. “Embedding Planar Graphs in Four Pages”. In: *J. Comput. Syst. Sci.* 38.1 (1989), pp. 36–67. DOI: 10.1016/0022-0000(89)90032-9.

[90] U. Zwick. “All pairs shortest paths using bridging sets and rectangular matrix multiplication”. In: *J. ACM* 49.3 (2002), pp. 289–317. DOI: 10.1145/567112.567114. Also at FOCS 1998.

Appendix

A Subcubic Equivalences Between Path, Matrix, and Triangle Problems

In [85], Vassilevska Williams and Williams prove for a list of various problems that either all of them have subcubic algorithms, or none of them do. This list includes

- computing a shortest path for each pair of points on weighted graphs (known as APSP),
- finding a minimum weight cycle in an undirected graph with non-negative weights (called MinimumCycle), and
- detecting if a weighted graph has a triangle of negative edge weight (called NegativeTriangle).

Theorem (NegativeTriangle $\leq_3$ MinimumCycle [85])**.** *Let G be an undirected graph with weights in $[1, M]$ with $M \in \mathbb{Z}^+ \cup \infty$. Finding a minimum weight cycle in an undirected graph with non-negative weights is at least as hard as deciding if a weighted graph contains a triangle of negative weight.*

Proof. The following proof is restated from [85]. Let $G = (V, E)$ be a given undirected graph with weights $w : E \to [-M, M]$. We define an undirected graph $G' = (V, E)$, which is just G, but with weights $w' : E \to [7M, 9M]$ defined as $w'(e) = w(e)+8M$. Any cycle C in G with k edges has length $w'(C) = 8Mk+w(C)$, and it holds that $7Mk \leq w'(C) \leq 9Mk$. Therefore, every cycle C with at least four edges has length $w'(C) \geq 28M$ and any triangle Δ has length $w'(\Delta) \leq 27M < 28M$. Thus, a minimum weight cycle in G' is a minimum weight triangle in G. $\square$

Since the minimum weight cycle in the constructed graph G' in the proof above is a triangle, we immediately obtain the following theorem.

Theorem 3.1 (APSP $\leq_3$ MinimumTriangle)**.** *Let G be an undirected graph with weights in $[7M, 9M]$ with $M \in \mathbb{Z}^+$ whose minimum weight cycle is a triangle. It is APSP-hard to find a minimum weight triangle in G.*

B Loop Variant for Algorithm 4.2 (5/3-Tournament)

The following result is a loop invariant for the outer loop of Algorithm 4.2. It is not used to show that Algorithm 4.2 returns a 5/3-tournament, however, we present this result here to give more insights on this algorithm.

In the following, we number the iterations of the outer loop as $1, 2, \ldots, m$. For each $i = 1, 2, \ldots, m+1$, define

- $P_i = \{p_k \mid 1 \leq k < i\} \cup \{q_k \mid 1 \leq k < i\}$, and
- $\overrightarrow{E}_i$ to be the set of edges that have an orientation at the beginning of the i-th iteration of the outer loop. If $i = m+1$ then $\overrightarrow{E}_i$ is the set of all m oriented edges of the complete graph, as computed by the algorithm.

Lemma. *For each $i = 1, 2, \ldots, m+1$, the following holds. Let a and b be any two distinct points in $P \setminus P_i$, and let c be any point in P_i. Then, it holds that*

- *$(a, c) \in \overrightarrow{E}_i$ if and only if $(b, c) \in \overrightarrow{E}_i$, and*
- *$(c, a) \in \overrightarrow{E}_i$ if and only if $(c, b) \in \overrightarrow{E}_i$.*

Proof. The proof is by induction on i. If $i = 1$, then both P_i and $\overrightarrow{E}_i$ are empty, and the statement vacuously holds.

Let i be such that $1 \leq i \leq m$ and assume that the statement in the lemma holds for $1, 2, \ldots, i$ (i.e., up to the beginning of the i-th iteration of the outer loop). We show that the statement also holds for $i+1$ (i.e., at the end of the i-th iteration of the outer loop).

Let a and b be two distinct points in $P \setminus P_{i+1}$, and let c be a point in P_{i+1}. We prove that, if (a, c) is contained in $\overrightarrow{E}_{i+1}$, then (b, c) is also contained in $\overrightarrow{E}_{i+1}$. The other claims in the lemma can be proved in a symmetric way.

Assume that $(a, c) \in \overrightarrow{E}_{i+1}$. Let k be the smallest integer such that $(a, c) \in \overrightarrow{E}_{k+1}$. Then, $1 \leq k \leq i$ and the pair ac is oriented as (a, c) in the k-th iteration of the outer loop.

Consider the k-th iteration of the outer loop. We may assume, w.l.o.g., that the pair $p_k q_k$ is oriented as (p_k, q_k). It follows from the algorithm that $\{a, c\} \cap \{p_k, q_k\} \neq \emptyset$. Thus, since $a \notin P_{k+1}$, the point c is in $\{p_k, q_k\}$.

We claim that $c = p_k$. If this is not the case, then $c = q_k$. Since $p_k q_k$ is oriented as (p_k, q_k), it then follows from the algorithm that the pair ac is oriented as (c, a). This is a contradiction.

First assume that $k < i$. Since both a and b are in $P \setminus P_{k+1}$, $c \in P_{k+1}$, and $(a, c) \in \overrightarrow{E}_{k+1}$, it follows from the induction hypothesis that $(b, c) \in \overrightarrow{E}_{k+1}$. Since $\overrightarrow{E}_{k+1} \subseteq \overrightarrow{E}_{i+1}$, we conclude that $(b, c) \in \overrightarrow{E}_{i+1}$.

It remains to consider the case when $k = i$. Thus, the pair ac is oriented during the i-th iteration of the outer loop and $c = p_i$. The edges of the triangle (a, c, q_i) are oriented as $a \to c \to q_i \to a$.

Let ℓ be an integer with $1 \leq \ell < i$. We show that the pair bc is not oriented during the ℓ-th iteration of the outer loop. Assume that this pair is oriented during this iteration. It follows from the algorithm that $\{b, c\} \cap \{p_\ell, q_\ell\} \neq \emptyset$. Since $b \notin P_{i+1}$, the point b is not in $\{p_\ell, q_\ell\}$. Thus, $c \in \{p_\ell, q_\ell\}$. First assume that bc is oriented as (b, c). Then $(b, c) \in \overrightarrow{E}_{\ell+1}$. Since both a and b are in $P \setminus P_{\ell+1}$ and c is in $P_{\ell+1}$, the induction hypothesis implies that $(a, c) \in \overrightarrow{E}_{\ell+1}$, which contradicts our choice of k (which is i). The second case is when bc is oriented as (c, b). Then, again by the induction hypothesis, $(c, a) \in \overrightarrow{E}_{\ell+1}$, which is a contradiction, because ac is oriented as (a, c).

Consider the i-th iteration of the outer loop. If we can show that, at the end of this iteration, bc is oriented as (b, c), then $(b, c) \in \overrightarrow{E}_{i+1}$.

If, at the beginning of the i-th iteration, the pair $q_i b$ is not oriented, or it is oriented as (q_i, b), then it follows from the algorithm that bc is oriented as (b, c).

It remains to consider the case when, at the start of the i-th iteration of the outer loop, the pair $q_i b$ is oriented as (b, q_i). We show that this cannot happen. Let j be the iteration of the outer loop during which $q_i b$ is oriented. Then $1 \leq j < i$ and $\{b, q_i\} \cap \{p_j, q_j\} \neq \emptyset$. Since $b \notin P_{j+1}$, the point q_i is in $\{p_j, q_j\}$.

If $p_j q_j$ is oriented as (p_j, q_j), then $q_i = p_j$. Since $p_j \in P_j$ and neither a nor b is in P_{j+1}, it follows from the induction hypothesis that the pair ap_j is oriented as (a, p_j). This is a contradiction, because $p_j = q_i$.

If $p_j q_j$ is oriented as (q_j, p_j), then $q_i = q_j$. Since $q_j \in P_j$ and neither a nor b is in P_{j+1}, it follows from the induction hypothesis that the pair aq_j is oriented as (a, q_j). This is a contradiction, because $q_j = q_i$. $\square$

Statement of Originality

The content of this thesis is based on six papers listed in Section 1.2. For each of these papers, I contributed to the conceptualization and development of the technical details and was the main contributor to the writing of the manuscripts.

For the writing of this thesis, I used LLMs exclusively for grammar checks and language refinement. All suggestions were reviewed and thoroughly checked.

I affirm that the intellectual content of this thesis is a product of my own work.

Eigenständigkeitserklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit selbstständig verfasst habe und keine anderen außer die angegebenen Quellen und Hilfsmittel verwendet sowie Zitate kenntlich gemacht habe.

Dortmund, den 20. August 2026

Antonia Kalb