# Efficient algorithms for constructing fault-tolerant geometric spanners

Christos Levcopoulos* Giri Narasimhan† Michiel Smid‡

December 19, 1997

**Abstract**

Let $S$ be a set of $n$ points in $\mathbb{R}^d$, and $k$ an integer such that $1 \leq k \leq n-2$. Algorithms are given that construct fault-tolerant spanners for $S$. If in such a spanner at most $k$ edges or vertices are removed, then each pair of points in the remaining graph is still connected by a "short" path. Our results include (i) an algorithm with running time $O(n \log^{d-1} n + kn \log\log n + k^2 n)$ that constructs a spanner with $O(k^2 n)$ edges, that is resilient to $k$ edge faults, (ii) an algorithm with running time $O(n \log n + k^2 n)$ that constructs a spanner with $O(k^2 n)$ edges, that is resilient to $k$ vertex faults, and (iii) an algorithm with running time $O(n \log n + c^k n)$ that constructs a spanner of degree $O(c^k)$, whose total edge length is bounded by $O(c^k)$ times the weight of a minimum spanning tree of $S$, and that is resilient to $k$ edge or vertex faults. Here, $c$ is a constant that is independent of $n$ and $k$.

Our algorithms are based on well-separated pairs, and approximate neighbors of pairs of points. The algorithms themselves are simple, but the correctness proofs are non-trivial.

*Department of Computer Science, Lund University, Sweden. E-mail: `Christos.Levcopoulos@dna.lth.se`.

†Department of Mathematical Sciences, The University of Memphis, Memphis TN 38152. E-mail: `giri@msci.memphis.edu`. Research supported by NSF Grant CCR-940-9752.

‡Department of Computer Science, University of Magdeburg, D-39106 Magdeburg, Germany. E-mail: `michiel@isg.cs.uni-magdeburg.de`.

# 1 Introduction

Spanners have applications in the design of geometric networks. Consider a set $S$ of $n$ points in $\mathbb{R}^d$, where the dimension $d$ is a constant. A geometric network on $S$ can be modeled as an undirected graph $G$ with vertex set $S$ and with edges $e = (a, b)$ of length $|e|$ that is defined as the Euclidean distance $|ab|$ between its two endpoints $a$ and $b$. Let $p$ and $q$ be two points of $S$, and let $P$ be a $pq$-path in $G$, i.e., a path in $G$ between $p$ and $q$. The length $|P|$ of $P$ is defined as the sum of the lengths of the edges of $P$.

Let $t > 1$ be a real number. We say that $G$ is a *$t$-spanner* for $S$, if for each pair of points $p, q \in S$, there exists a $pq$-path in $G$ of length at most $t$ times the Euclidean distance between $p$ and $q$.

The problem of constructing spanners has been investigated by many researchers. Keil and Gutwin [9] showed that for any $t > 1$, and any set $S$ of $n$ points in the plane, a $t$-spanner for $S$ having $O(n)$ edges can be constructed in $O(n \log n)$ time. Salowe [13], Vaidya [14] and Callahan and Kosaraju [4] showed the same result for any fixed dimension $d$. Later work concentrated on constructing spanners that have other properties, such as low degree and low total edge length. For example, Das and Narasimhan [8] gave an $O(n \log^2 n)$-time algorithm that constructs for any set $S$ of $n$ points in $\mathbb{R}^d$, and any constant $t > 1$, a $t$-spanner for $S$ in which the degree of every point is bounded by a constant, and whose total edge length is proportional to the weight of a minimum spanning tree of $S$. The time complexity was later improved by Arya et al. [2] to $O(n \log n)$.

In this paper, we show that it is possible to incorporate *fault-tolerance* into such geometric networks while still maintaining many other desirable properties like small weight and degree. Fault tolerance is intimately related to the graph-theoretic concept of *connectivity*. The edge (vertex) connectivity of a graph is the number of edges (vertices) that can be removed without disconnecting it. Fault-tolerant networks are usually designed by making them highly connected.

We construct geometric networks that are more than just resilient to edge or vertex faults. Our networks have the property that after removing at most $k$ edges (or vertices), the remaining graph still contains "short" paths between each pair of points. Before we can define this formally, we have to introduce the following notation.

If $S$ is a set of points, then $K_S$ denotes the complete Euclidean graph on $S$. Let $G = (S, E)$ be a graph, and let $E'$ be a subset of $E$. Then $G \setminus E'$

denotes the graph $(S, E \setminus E')$. Similarly, if $S'$ is a subset of $S$, then $G \setminus S'$ is the graph with vertex set $S \setminus S'$, and edge set the set of all edges of $E$ that have both endpoints in $S \setminus S'$.

**Definition 1** *Let $S$ be a set of $n$ points in $\mathbb{R}^d$, $t > 1$ a real number, and $k$ an integer such that $1 \leq k \leq n - 2$. A graph $G = (S, E)$ is called a* $k$-edge fault-tolerant $t$-spanner *for $S$, or* $(k, t)$-EFTS, *if the following holds for each subset $E'$ of $E$ having size at most $k$:*

- *For each pair $p$ and $q$ of distinct points in $S$, the graph $G \setminus E'$ contains a $pq$-path having length at most $t$ times the length of a shortest $pq$-path in the graph $K_S \setminus E'$.*

*Similarly, $G$ is called a* $k$-vertex fault-tolerant $t$-spanner *for $S$, or* $(k, t)$-VFTS, *if the following holds for each subset $S'$ of $S$ having size at most $k$:*

- *The graph $G \setminus S'$ is a $t$-spanner for the points of $S \setminus S'$.*

Note that in an edge or vertex fault-tolerant $t$-spanner, our definition insists that between every pair of points there is a path whose length is at most $t$ times the best possible path under the circumstances, i.e., the shortest path in the graph $K_S \setminus E'$ or $K_S \setminus S'$.

In the definition of a $(k, t)$-VFTS, we could have required that for each pair $p$ and $q$ of distinct points in $S \setminus S'$, the graph $G \setminus S'$ contains a $pq$-path having length at most $t$ times the length of a shortest $pq$-path $P$ in the graph $K_S \setminus S'$. Since $K_S \setminus S'$ is the complete Euclidean graph on the point set $S \setminus S'$, this shortest path $P$, however, consists of the single edge $(p, q)$. Hence, $G \setminus S'$ is indeed a $t$-spanner for $S \setminus S'$.

To our knowledge, fault-tolerant spanners have not been investigated before.

## 1.1 Our results

We will show in Section 4 that a $(k, t)$-EFTS having $O(k^2 n)$ edges can be constructed in $O(n \log^{d-1} n + kn \log \log n + k^2 n)$ time, where the constant factors only depend on $d$ and $t$.

Our construction is based on the *well-separated pair decomposition* of Callahan and Kosaraju [5]. They show in [4] that a $t$-spanner with $O(n)$

edges can be obtained from this decomposition. We extend this result to fault-tolerant spanners. Our construction is simple, but the correctness proof is quite intricate.

The construction needs the notion of a "neighbor" of a pair of points. More formally, given two distinct points $a$ and $b$ in $\mathbb{R}^d$, a neighbor of $(a, b)$ is a point $c \in S \setminus \{a, b\}$ for which $|ac|+|cb|$ is minimal. This notion generalizes to $k$ neighbors in a straightforward way. The problem is that it is not known how to compute such neighbors in polylogarithmic time while using close to linear space. For our purposes, however, it suffices to find an approximation to these neighbors. For this reason, we develop the notion of *strong* approximate neighbors of a pair of points, which is interesting in its own right. In Section 3, we show how to compute $k$ strong approximate neighbors for an off-line batch of $m$ pairs $(a, b)$, in $O((n+m)(\log(n+m) + \log^{d-1}) + km(\log\log n + \log k))$ total time. Our algorithm is based on range trees, and is similar to that of computing the $\Theta$-graph (see [7, 9, 12, 3]), which is in fact a $t$-spanner. Again, the algorithm is simple, but the correctness proof is non-trivial.

In Section 5, we consider tolerance to vertex faults and show that a $(k, t)$-VFTS having $O(k^2 n)$ edges can be constructed in $O(n \log n + k^2 n)$ time. Again, the constant factors only depend on $d$ and $t$. This construction is also based on well-separated pairs, but is much simpler than that for edge fault-tolerant spanners.

In Section 6, we consider fault-tolerant spanners having low total edge length. The main result is a simple construction that transforms any $t$-spanner $G_0$ into a $(k, t)$-EFTS $G$. In fact, the resulting graph $G$ is also a $(k, t)$-VFTS. If the degree of each vertex in $G_0$ is bounded by $D$, then each vertex of $G$ has degree $O(D^{k+1})$. Moreover, the total edge length of $G$ is proportional to $k \cdot D^k$ times that of $G_0$. The running time of the algorithm that transforms $G_0$ into $G$ is bounded by $O(D^{k+1} n)$.

Arya et al.[2] show how to compute in $O(n \log n)$ time a $t$-spanner $G_0$ whose total edge length is proportional to the weight of a minimum spanning tree of $S$, and in which each point has a degree that is bounded by a constant $D$, that only depends on $d$ and $t$. Combining this with our transformation of Section 6 gives an algorithm that constructs in $O(n \log n + D^{k+1} n)$ time, a $(k, t)$-EFTS (or $(k, t)$-VFTS) whose total edge length is proportional to $k \cdot D^k$ times the weight of a minimum spanning tree of $S$, and in which each point has degree $O(D^{k+1})$. If $k$ is a constant, then this result is optimal. The optimality of the running time follows from Chen et al.[6], who proved that computing any $t$-spanner takes $\Omega(n \log n)$ time in the algebraic computation tree model.

In Section 2, we define and recall some basic notions that are used throughout the rest of the paper. In particular, we define well-separated pairs and strong approximate neighbors. We conclude the paper in Section 7 by mentioning some open problems.

# 2 Preliminaries

## 2.1 Well-separated pairs

Our construction of fault-tolerant spanners is based on the notion of *well-separated pairs*, which is due to Callahan and Kosaraju [5]. Before we can define this notion, we have to introduce the following notation.

If $X$ is a bounded subset of $\mathbb{R}^d$, then we denote by $R(X)$ the smallest axes-parallel $d$-dimensional rectangle that contains $X$. We call $R(X)$ the *bounding rectangle* of $X$.

**Definition 2** *Let $s > 0$ be a real number, and let $A$ and $B$ be two finite sets of points in $\mathbb{R}^d$. We say that $A$ and $B$ are* well-separated w.r.t. $s$, *if there are two disjoint d-dimensional balls $C_A$ and $C_B$, having the same radius, such that (i) $C_A$ contains the bounding rectangle $R(A)$ of $A$, (ii) $C_B$ contains $R(B)$, and (iii) the distance between $C_A$ and $C_B$ is at least equal to $s$ times the radius of $C_A$.*

See Figure 1 for an illustration. In this paper, $s$ will always be a constant, called the *separation constant.*

**Definition 3 ([5])** *Let $S$ be a set of $n$ points in $\mathbb{R}^d$, and $s > 0$ a real number. A* well-separated pair decomposition *(WSPD) for $S$ (w.r.t. $s$) is a sequence of pairs of non-empty subsets of $S$,*

$$\{A_1, B_1\}, \{A_2, B_2\}, \ldots, \{A_m, B_m\},$$

*such that*

1. *$A_i \cap B_i = \emptyset$, for all $i = 1, 2, \ldots, m$,*

2. *for each unordered pair $\{p, q\}$ of distinct points of $S$, there exists exactly one pair $\{A_i, B_i\}$ in the sequence, such that $p \in A_i$ and $q \in B_i$, and*

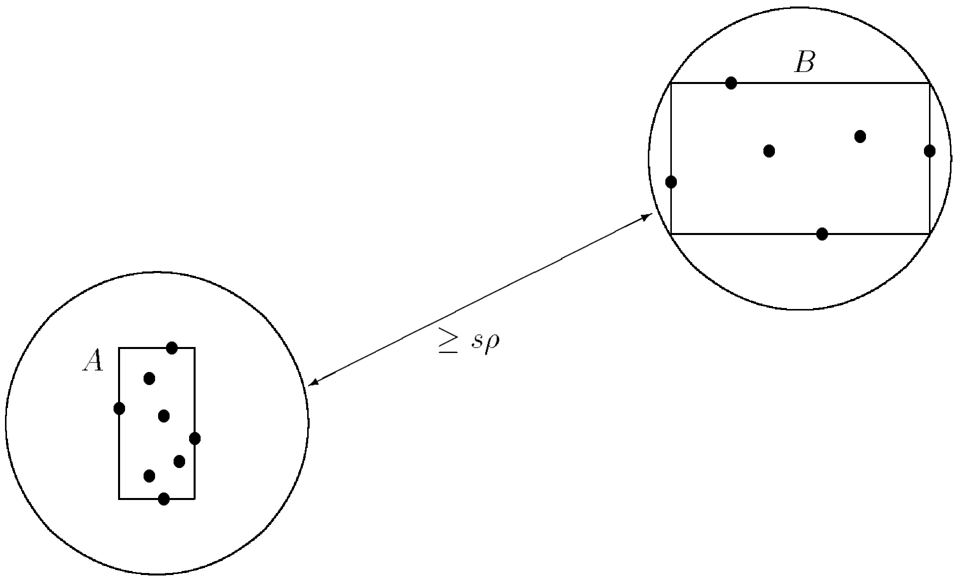


Figure 1: *Two planar point sets $A$ and $B$ that are well-separated w.r.t. $s$. Both circles have radius $\rho$; their distance is at least $s\rho$.*

---

*3. $A_i$ and $B_i$ are well-separated w.r.t. $s$, for all $i = 1, 2, \ldots, m$.*

*The integer $m$ is called the* size *of the WSPD.*

**Lemma 1** *Let $S$ be a set of $n$ points in $\mathbb{R}^d$, and $p, q$ a closest pair in $S$. Consider a WSPD*

$$\{A_1, B_1\}, \{A_2, B_2\}, \ldots, \{A_m, B_m\}$$

*for $S$, with separation constant $s > 2$. Let $i$ be the index such that $p \in A_i$ and $q \in B_i$. Then $A_i = \{p\}$ and $B_i = \{q\}$.*

**Proof.** The proof follows immediately from Definition 3. ∎

**Theorem 1 ([5])** *Let $S$ be a set of $n$ points in $\mathbb{R}^d$, and $s > 0$ a separation constant. In $O(n \log n + \alpha_{ds} n)$ time, we can compute a WSPD for $S$ of size at most $\alpha_{ds} n$. The constant in this Big-Oh bound does not depend on $d$ and $s$. Moreover, for a large separation constant $s$, the value of $\alpha_{ds}$ is proportional to $((c+1)s)^d$ for some constant $c$.*

The following lemma shows that all edges connecting points from the two sets of a well-separated pair are approximately of the same length.

**Lemma 2** *Let $s > 0$ be a real number, and let $A$ and $B$ be two non-empty sets of points in $\mathbb{R}^d$ that are well-separated w.r.t. $s$. Let $a$ and $a'$ be two points in $A$, $b$ and $b'$ two points in $B$, and $v$ an arbitrary point in $\mathbb{R}^d$. Then*

$$|av| + |vb| \leq (1 + 4/s)\,(|a'v| + |vb'|).$$

**Proof.** Let $C_A$ and $C_B$ be two disjoint balls having the same radius, say $\rho$, such that $A \subseteq C_A$, $B \subseteq C_B$, and the distance between them is at least $s\rho$. Then, $|aa'| \leq 2\rho$ and $|bb'| \leq 2\rho$. The triangle inequality implies that

$$|av| + |vb| \leq |aa'| + |a'v| + |vb'| + |bb'| \leq 4\rho + |a'v| + |vb'|.$$

Since $|a'v| + |vb'| \geq |a'b'| \geq s\rho$, it follows that $4\rho \leq (4/s)\,(|a'v| + |vb'|)$. ∎

## 2.2 Strong approximate neighbors of a pair of points

Let $S$ be a set of $n$ points in $\mathbb{R}^d$, and let $a$ and $b$ be two distinct points in $\mathbb{R}^d$. A point $c \in S \setminus \{a, b\}$ is called a *neighbor of the pair* $(a, b)$, if $|ac| + |cb| = \min\{|ax| + |xb| : x \in S \setminus \{a, b\}\}$. More generally, let $k$ be an integer such that $1 \leq k \leq n - 2$. A sequence $c_1, c_2, \ldots, c_k$ of pairwise distinct points of $S \setminus \{a, b\}$ is called a sequence of $k$ *neighbors of the pair* $(a, b)$, if the values $|ac_i| + |c_ib|$, $1 \leq i \leq k$, are the $k$ smallest elements in the set $\{|ax| + |xb| : x \in S \setminus \{a, b\}\}$. We will always assume that the $c_i$'s are numbered in such a way that $|ac_i| + |c_ib| \leq |ac_{i+1}| + |c_{i+1}b|$ for all $1 \leq i < k$.

There is a data structure for the set $S$, based on partition trees, linearization and parametric search, such that a neighbor of any query pair $(a, b)$ can be computed in sublinear time. (This follows from general results in [1].) For example, if $d = 2$, then for any $m$, $n \leq m \leq n^2$, there is a data structure having size $O(m)$ and query time close to $O(n/\sqrt{m})$ (ignoring polylogarithmic factors). It is not known if there is a data structure for this problem having polylogarithmic query time and close to linear size. Fortunately, it suffices for our application to compute (strong) approximate neighbors, which are defined as follows. (The third condition is the reason for the use of the word "strong".)

**Definition 4** *Let $c_1, c_2, \ldots, c_k$ be a sequence of $k$ neighbors of the pair $(a, b)$, and $t > 1$ a real number. A sequence $c'_1, c'_2, \ldots, c'_k$ of pairwise distinct points of $S \setminus \{a, b\}$ is called a sequence of* strong $t$-approximate neighbors of the pair $(a, b)$, *if*

1. $|ac'_i| + |c'_i b| \leq |ac'_{i+1}| + |c'_{i+1} b|$ *for all* $1 \leq i < k$,
2. $|ac_i| + |c_i b| \leq |ac'_i| + |c'_i b| \leq t\,(|ac_i| + |c_i b|)$ *for all* $1 \leq i \leq k$, *and*
3. *for all* $1 \leq i < j \leq k$, *if* $|ac'_j| + |c'_j b| > t\,(|ac'_i| + |c'_i b|)$, *then* $c_i \in \{c'_1, c'_2, \ldots, c'_k\}$.

In Section 3, we will show how these strong approximate neighbors can be computed efficiently.

# 3 Computing strong approximate neighbors

As mentioned earlier, the notion of strong approximate neighbors of a pair of points was introduced because they can be computed more efficiently than exact neighbors of a pair of points. Our algorithm uses the notion of cones. A *(simplicial) cone* is the intersection of $d$ halfspaces in $\mathbb{R}^d$. The hyperplanes that bound these halfspaces are assumed to be in general position, in the sense that their intersection is a point, called the *apex* of the cone. In two-dimensional space, a cone having its apex at the point $a$ is a wedge bounded by two rays emanating from $a$.

Let $\theta$ be a fixed angle such that $0 < \theta \leq \pi$. Let $\mathcal{C}$ be a collection of cones such that (i) each cone has its apex at the origin, (ii) each cone has *angular diameter* at most $\theta$, i.e., for any two points $p$ and $q$ inside a cone, the vectors $\vec{p}$ and $\vec{q}$ make an angle between zero and $\theta$, and (iii) all cones cover $\mathbb{R}^d$.

In Yao [15], it is shown how such a collection $\mathcal{C}$, consisting of $(c/\theta)^{d-1}$ cones, for some constant $c$, can be constructed. In two-dimensional space, and for $\theta = 2\pi/m$, we can obtain $\mathcal{C}$ by rotating the positive $x$-axis over angles $i \cdot \theta$, $0 \leq i < m$. This gives $m$ rays. Each wedge between two successive rays defines a cone of $\mathcal{C}$.

For each cone $C \in \mathcal{C}$, let $l_C$ be a fixed ray that emanates from the origin and that is contained in $C$.

Let $C$ be any cone of $\mathcal{C}$ and let $a$ be any point in $\mathbb{R}^d$. We define $C_a := C + a := \{x + a : x \in C\}$, i.e., $C_a$ is the cone obtained by translating $C$ such that its apex is at $a$. Similarly, we define $l_{C,a} := l_C + a$. Hence, $l_{C,a}$ is a ray that emanates from $a$ and that is contained in the translated cone $C_a$.

The proof of the following lemma can be found in [3].

**Lemma 3** **([3])** *Assume that $0 < \theta < \pi/4$. Let $a$ and $b$ be two distinct points in $\mathbb{R}^d$, and let $C$ be the cone of $\mathcal{C}$ such that $b \in C_a$. Let $c$ be any point in $C_a$ such that the projection of $c$ onto the ray $l_{C,a}$ is at least as close to $a$ as the projection of $b$ onto $l_{C,a}$. Then*

$$|cb| \leq |ab| - (\cos\theta - \sin\theta)|ac|.$$

### 3.1 Definition and correctness proof

Let $S$ be a set of $n$ points in $\mathbb{R}^d$, $a$ and $b$ two distinct points in $\mathbb{R}^d$, $t > 1$ a real constant, and $k$ an integer such that $1 \leq k \leq n-2$. Let $\theta$ be an angle such that $0 < \theta < \pi/4$ and $t \geq 1/(\cos\theta - \sin\theta)$.

For each cone $C$ of $\mathcal{C}$, let $S_{C,a}$ be the set of all points of $S \setminus \{a,b\}$ that are contained in the translated cone $C_a$. Let $Y_{C,a}$ be the set consisting of the $\min(k, |S_{C,a}|)$ points of $S_{C,a}$ whose orthogonal projections onto $l_{C,a}$ are closest to $a$.

Let $Z_a := \cup_C Y_{C,a}$. Finally, let $c'_1, c'_2, \ldots, c'_k$ be $k$ pairwise distinct points of $Z_a$ such that

1. $|ac'_i| + |c'_ib| \leq |ac'_{i+1}| + |c'_{i+1}b|$ for all $1 \leq i < k$, and
2. the values $|ac'_i| + |c'_ib|$, $1 \leq i \leq k$, are the $k$ smallest elements in the set $\{|ac| + |cb| : c \in Z_a\}$.

**Lemma 4** *The points $c'_1, c'_2, \ldots, c'_k$ form a sequence of strong $t$-approximate neighbors of the pair $(a,b)$.*

**Proof.** Let $c_1, c_2, \ldots, c_k$ be the sequence of $k$ exact neighbors of the pair $(a,b)$. Recall that these points are numbered such that $|ac_i| + |c_ib| \leq |ac_{i+1}| + |c_{i+1}b|$ for all $1 \leq i < k$. We will prove that the points $c'_1, c'_2, \ldots, c'_k$ satisfy the three conditions in Definition 4. The first condition obviously holds.

Since the points $c_i$, $1 \leq i \leq k$, are the exact neighbors of $(a,b)$, and the points $c'_i$, $1 \leq i \leq k$, belong to $S \setminus \{a,b\}$, it is clear that $|ac_i| + |c_ib| \leq |ac'_i| + |c'_ib|$ for all $1 \leq i \leq k$. To prove the second inequality in the second condition of Definition 4, we define a sequence $c''_1, c''_2, \ldots, c''_k$ of points, as follows.

Let $C$ be any cone of $\mathcal{C}$, and let $m$ be the number of points of the set $\{c_1, c_2, \ldots, c_k\}$ that are contained in the translated cone $C_a$. For any point $c \in C_a$, let $d'(a,c)$ denote the Euclidean distance between $a$ and the projection of $c$ onto the ray $l_{C,a}$. Let $i_1, i_2, \ldots, i_m$ be the indices such that $c_{i_1}, c_{i_2}, \ldots, c_{i_m} \in C_a$, and $d'(a, c_{i_1}) \leq d'(a, c_{i_2}) \leq \ldots \leq d'(a, c_{i_m})$. Note that the indices $i_j$ are, in general, not sorted. For $1 \leq j \leq m$, let $c''_{i_j}$ be the point in $S_{C,a}$ such that $d'(a, c''_{i_j})$ is the $j$-th smallest element in the set $\{d'(a,c) : c \in S_{C,a}\}$.

Repeating this for all cones $C$ of $\mathcal{C}$ gives us $k$ pairwise distinct points $c''_1, c''_2, \ldots, c''_k$ which are all contained in the set $Z_a$. Our first claim is that

$$d'(a, c''_i) \leq d'(a, c_i), \tag{1}$$

for all $1 \leq i \leq k$. To prove this, let $C$ be the cone such that $c_i \in C_a$. Let $j$ be the index such that $d'(a, c_i)$ is the $j$-th smallest element in the set $\{d'(a,c) : c \in \{c_1, c_2, \ldots, c_k\} \cap C_a\}$. We have chosen $c''_i$ in such a way that $d'(a, c''_i)$ is the $j$-th smallest element in the set $\{d'(a,c) : c \in S_{C,a}\}$. This implies that (1) holds.

Let $1 \leq j \leq k$. We claim that

$$|ac''_j| + |c''_j b| \leq t\,(|ac_j| + |c_j b|). \tag{2}$$

To prove this, note that Lemma 3 and (1) imply that

$$|c''_j c_j| \leq |ac_j| - (\cos\theta - \sin\theta)|ac''_j|.$$

Therefore,

$$\begin{aligned}
|ac''_j| + |c''_j b| &\leq |ac''_j| + |c''_j c_j| + |c_j b| \\
&\leq \frac{|ac_j| - |c''_j c_j|}{\cos\theta - \sin\theta} + |c''_j c_j| + |c_j b| \\
&\leq t|ac_j| + \left(1 - \frac{1}{\cos\theta - \sin\theta}\right)|c''_j c_j| + |c_j b| \\
&\leq t|ac_j| + |c_j b| \\
&\leq t\,(|ac_j| + |c_j b|),
\end{aligned}$$

which proves (2).

Now let $1 \leq j \leq i$. Then

$$|ac''_j| + |c''_j b| \leq t\,(|ac_j| + |c_j b|) \leq t\,(|ac_i| + |c_i b|).$$

Hence, the set $Z_a$ contains at least $i$ points $c$ such that

$$|ac| + |cb| \leq t\,(|ac_i| + |c_i b|).$$

Therefore, we must have

$$|ac'_i| + |c'_i b| \leq t\,(|ac_i| + |c_i b|).$$

Hence, we have shown that the second condition in Definition 4 is satisfied.

It remains to prove the third condition. Let $1 \leq i < j \leq k$ be indices, and assume that

$$|ac'_j| + |c'_j b| > t\,(|ac'_i| + |c'_i b|).$$

We will show that $c_i \in \{c'_1, c'_2, \ldots, c'_k\}$.

Let $C$ be the cone such that $c_i \in C_a$. We claim that $d'(a, c_i)$ belongs to the $\min(k, |S_{C,a}|)$ smallest elements in the set $\{d'(a,c) : c \in S_{C,a}\}$. If this is true, then $c_i$ is clearly contained in $Z_a$. Hence, since $c_i$ is the $i$-th exact neighbor of $(p,q)$, this point must be an element of the set $\{c'_1, c'_2, \ldots, c'_k\}$.

We prove the claim by contradiction. If the claim is not true, then the set $S_{C,a}$ contains more than $k$ elements, and $d'(a, c_i)$ has rank larger than $k$ in the set $\{d'(a,c) : c \in S_{C,a}\}$. Let $c$ be any point in $S_{C,a}$ such that $d'(a,c)$ has rank at most $k$ in the set $\{d'(a,c) : c \in S_{C,a}\}$. Then, $c$ is contained in the set $Z_a$, and $d'(a,c) \leq d'(a,c_i)$. The latter inequality implies that $|ac| + |cb| \leq t\,(|ac_i| + |c_i b|)$. (The proof is the same as that of (2).) Hence,

$$\begin{aligned} |ac| + |cb| &\leq t\,(|ac_i| + |c_i b|) \\ &\leq t\,(|ac'_i| + |c'_i b|) \\ &< |ac'_j| + |c'_j b| \\ &\leq |ac'_k| + |c'_k b|. \end{aligned}$$

This shows that the set $Z_a$ contains at least $k$ points $c$ such that $|ac| + |cb| < |ac'_k| + |c'_k b|$, which is clearly a contradiction. ∎

## 3.2 The algorithm

In this section, we show how strong approximate neighbors can be computed efficiently. To be more precise, we give an algorithm that solves the following problem: Given a set $S$ of $n$ points in $\mathbb{R}^d$, and an *off-line* sequence $F$ of $m$ pairs of distinct points in $\mathbb{R}^d$, compute for each pair $(a,b)$ in $F$, the sequence $c'_1, c'_2, \ldots, c'_k$ of strong $t$-approximate neighbors of Lemma 4.

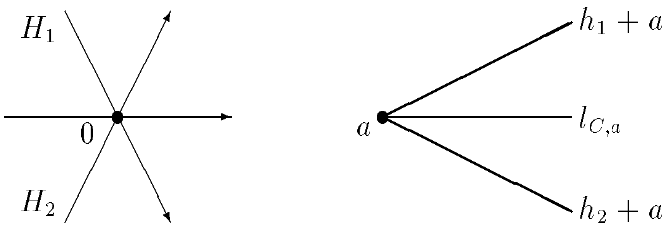


Figure 2: *The directed lines $H_1$ and $H_2$, and the translated cone $C_a$.*

If $k = 1$, then this problem is basically the same as that of computing the so-called $\Theta$-graph. In [7, 9, 12, 3], efficient algorithms are given for constructing this graph. Our algorithm is a direct generalization of the algorithm in [3], and we will describe it for the case when the dimension $d$ is equal to two. The generalization to any dimension $d$ is straightforward.

Consider our collection $\mathcal{C}$ of cones with angular diameter $\theta$. Let $C$ be a fixed cone of $\mathcal{C}$. Recall that $l_C$ is a ray emanating from the origin that is contained in $C$. We introduce the following notation. Refer to Figure 2. Let $h_1$ and $h_2$ be the two lines that contain the bounding rays of $C$. Let $H_1$ and $H_2$ be the lines through the origin that are orthogonal to $h_1$ and $h_2$, respectively. We assume for simplicity that $l_C$ coincides with the positive $x$-axis, and that $h_1$ and $h_2$ contain the upper and lower bounding rays of $C$, respectively. We give the lines $H_1$, $H_2$, and the $x$-axis directions, as indicated in Figure 2.

Consider a pair $(a, b)$ in $F$. Let $S_{C,a}$ be the set of points of $S \setminus \{a, b\}$ that are contained in the translated cone $C_a$. We have to compute the $k' := \min(k, |S_{C,a}|)$ points of $S_{C,a}$ whose projections onto $l_{C,a}$ are closest to $a$. That is, among all points of $S \setminus \{a, b\}$, we have to find $k'$ points that are

1. below $h_1 + a$,
2. above $h_2 + a$, and
3. furthest to the left.

### 3.2.1 A simpler problem

Let us first consider a simpler problem: Among all points of $S \setminus \{a, b\}$, find $k$ points that are (i) above $h_2 + a$, and (ii) furthest to the left.

We store the points of $S$ in a data structure $T$ that is defined as follows. There is a balanced binary search tree, storing the points of $S$ in its leaves, sorted in the direction of $H_2$. Each node $u$ of this tree stores information to guide searches. Let $S_u$ be the set of points stored in the subtree of $u$. Then, $u$ stores a list $L_u$ containing the $\min(k+1, |S_u|)$ points of $S_u$ that are furthest to the left, sorted by their $x$-coordinates.

Let $(a, b)$ be a query pair of distinct points in the plane. Then we initialize an empty set $M$, and follow the path from the root to the leftmost leaf storing a point that is above $h_2 + a$, i.e., that is larger than $a$ w.r.t. the direction of $H_2$. For each node $u$ on this path, if we move to its left child, then we insert the right child of $u$ into $M$. The final set $M$ has the property that the collection $S_u$, $u \in M$, forms a partition of all points of $S \setminus \{a\}$ that are above $h_2 + a$. Hence, we want to find the $k$ leftmost points in $(\cup_{u \in M} L_u) \setminus \{b\}$. Since the lists $L_u$ are sorted by $x$-coordinate, we can find these $k$ points, by using a heap, in $O(|M| + k \log |M|)$ time. Since the set $M$ has size $O(\log n)$, the entire query algorithm takes $O(\log n + k \log \log n)$ time.

The data structure $T$ can be maintained under insertions of points. Note that $T$ is similar to a two-dimensional range tree, see Lueker [10]. Therefore, we can use the algorithm of Mehlhorn and Näher [11] that uses dynamic fractional cascading—for the insertions-only case. In this way, an insertion takes $O(\log n)$ time.

#### 3.2.2 Solving the off-line sequence of queries

Now we are ready to solve our actual problem. Recall that we are given a set $S$ of $n$ points in the plane, an off-line sequence $F$ of $m$ query pairs, and a cone $C$.

Let $V := \{S \cup \{a : \exists b, (a, b) \in F\}$. We sort the points of $V$ according to the direction of line $H_1$. Then we initialize an empty data structure $T$ of Section 3.2.1, and visit the points of $V$ in decreasing order, i.e., in the opposite direction of $H_1$. If we encounter a point of $S$, then we insert it into $T$. Otherwise, we encounter a point $a$ such that $(a, b) \in F$, for some point $b$. At this moment, the points of $S$ that are below the line $h_1 + a$ are precisely the points stored in $T$. Therefore, by querying $T$ using the algorithm given in Section 3.2.1, we find the $k' = \min(k, |S_{C,a}|)$ points of $S \setminus \{a, b\}$ that are below $h_1 + a$, above $h_2 + a$, and furthest to the left.

It follows from the results of Section 3.2.1, that the total time of this algorithm is bounded by

$$O((n+m)\log(n+m) + km \log\log n).$$

In this way, we get—for our fixed cone $C$—for each pair $(a,b) \in F$ the corresponding set $Y_{C,a}$ that was defined in the beginning of Section 3.1.

We run this algorithm for each cone $C \in \mathcal{C}$ separately. Then, for each pair $(a,b)$, we extract the sequence $c'_1, c'_2, \ldots, c'_k$ of strong $t$-approximate neighbors from the set $Z_a = \cup_C Y_{C,a}$, and sort them such that $|ac'_i| + |c'_i b| \leq |ac'_{i+1}| + |c'_{i+1} b|$, $1 \leq i < k$. This final step can be done in $O(|\mathcal{C}|k + k \log k)$ time per pair $(a,b)$.

**Theorem 2** *Let $S$ be a set of $n$ points in $\mathbb{R}^d$, $F$ an off-line sequence of $m$ pairs of distinct points in $\mathbb{R}^d$, $t > 1$ a real constant, and $k$ an integer such that $1 \leq k \leq n-2$. In*

$$O\left(\beta_{dt}\left((n+m)(\log(n+m) + \log^{d-1} n) + km(\log\log n + \log k)\right)\right)$$

*time, we can compute for each pair $(a,b)$ in $F$ the sequence of $k$ strong $t$-approximate neighbors of Lemma 4. The value of $\beta_{dt}$ only depends on $d$ and $t$, and if $t \downarrow 1$, then $\beta_{dt} \sim (c/(t-1))^{d-1}$ for some constant $c$.*

**Proof.** For $d = 2$, the claim follows from the discussion above. This result can be generalized to any dimension $d$ by using standard results on range trees. (See e.g. [11].) The asymptotic behavior of $\beta_{dt}$ follows from the facts that $|\mathcal{C}| \sim (c/\theta)^{d-1}$, and $t = 1/(\cos\theta - \sin\theta) \sim 1 + \theta$. ■

# 4 An edge fault-tolerant spanner

In this section, we define a graph $G$ based on the well-separated pair decomposition and strong approximate neighbors, and show that it is an edge fault-tolerant spanner.

Let $S$ be a set of $n$ points in $\mathbb{R}^d$, $t > 1$ a real constant, and $k$ an integer such that $1 \leq k \leq n-2$. Consider an arbitrary WSPD

$$\{A_1, B_1\}, \{A_2, B_2\}, \ldots, \{A_m, B_m\}$$

for $S$, with separation constant

$$s = \max\left(\frac{8t+4}{t-1}, \frac{4}{t^{1/12}-1}\right).$$

Our graph $G$ has the points of $S$ as its vertices. Below, we define for each $i$, $1 \leq i \leq m$, a set $E_i$ of edges. The edge set $E$ of $G$ is then defined as $E := \cup_i E_i$.

Let $1 \leq i \leq m$, and consider the well-separated pair $\{A_i, B_i\}$. We assume without loss of generality that $|A_i| \geq |B_i|$. To define $E_i$, we distinguish four cases.

**Case 1:** $|B_i| \geq k+1$.

Choose $k+1$ arbitrary, but pairwise distinct points $a_j \in A_i$, $1 \leq j \leq k+1$, and $k+1$ arbitrary, but pairwise distinct points $b_j \in B_i$, $1 \leq j \leq k+1$. The edge set $E_i$ consists of the $k+1$ edges $(a_j, b_j)$, $1 \leq j \leq k+1$.

**Case 2:** $|B_i| \leq k$ and $|A_i| \geq k+1$.

Choose $k+1$ arbitrary, but pairwise distinct points $a_j \in A_i$, $1 \leq j \leq k+1$. Let $B_i = \{b_1, b_2, \ldots, b_x\}$, where $x = |B_i| \leq k$. The edge set $E_i$ consists of the $x(k+1)$ edges $(a_j, b_l)$, $1 \leq j \leq k+1$, $1 \leq l \leq x$. Hence, $E_i$ has size at most $k(k+1)$.

**Case 3:** $|A_i| \leq k$ and $|A_i| + |B_i| \geq k+2$.

In this case, the set $E_i$ is defined as the edge set of the complete Euclidean graph on the points of $A_i \cup B_i$. Note that $E_i$ has size $\binom{|A_i|+|B_i|}{2} \leq \binom{2k}{2} = k(2k-1)$.

**Case 4:** $|A_i| + |B_i| \leq k+1$.

Choose an arbitrary point $a \in A_i$, and an arbitrary point $b \in B_i$. Let $c'_1, c'_2, \ldots, c'_k$ be a sequence of $k$ strong $t^{1/3}$-approximate neighbors of the pair $(a, b)$ in the set $S$. The set $E_i$ is defined as the edge set of the complete Euclidean graph on the points of $A_i \cup B_i \cup \{c'_1, c'_2, \ldots, c'_k\}$. Note that $E_i$ has size at most $\binom{|A_i|+|B_i|+k}{2} \leq \binom{2k+1}{2} = k(2k+1)$.

This concludes the definition of our graph $G$. Note that $E$, the edge set of $G$, has size $O(k^2 m)$.

## 4.1 The graph $G$ is a $(k,t)$-EFTS for $S$

We now prove that the above construction does have the requisite properties. The following lemma turns out to be useful.

**Lemma 5** *Let $n \geq 3$, $K_n$ the complete graph on $n$ vertices, and $E$ an arbitrary set of at most $n-2$ edges from $K_n$. For any two vertices $p$ and $q$, there is a $pq$-path in the graph $K_n \setminus E$ having at most two edges.*

**Proof.** If $p = q$, then the claim trivially holds. So assume that $p \neq q$. In $K_n$, there are $n-1$ edge-disjoint $pq$-paths having at most two edges; one of them consists of the single edge $(p,q)$, whereas the other $n-2$ paths contain exactly two edges. By removing at most $n-2$ edges from $K_n$, at least one of these $n-1$ paths remains intact. ∎

Let $p$ and $q$ be two distinct points of $S$, and let $E'$ be an arbitrary subset of $E$ of size at most $k$. We will prove that the graph $G \setminus E'$ contains a $pq$-path having length at most $t$ times the length of a shortest $pq$-path in the graph $K_S \setminus E'$.

Let $i$, $1 \leq i \leq m$, be the index such that $p \in A_i$ and $q \in B_i$. According to Definition 3, $i$ exists and is, in fact, unique.

**Lemma 6** *Assume that $|A_i| + |B_i| \leq k+1$. Then there is a pq-path in $G \setminus E'$ of length at most $t$ times the length of a shortest pq-path in $K_S \setminus E'$.*

**Proof.** First note that we are in Case 4 of the construction of $G$. Consider the points $a \in A_i$ and $b \in B_i$, and the sequence $c'_1, c'_2, \ldots, c'_k$ of $k$ strong $t^{1/3}$-approximate neighbors of the pair $(a,b)$ that were chosen for the construction of the edge set $E_i$. Recall that we number these points such that $|ac'_j| + |c'_j b| \leq |ac'_{j+1}| + |c'_{j+1} b|$ for all $1 \leq j < k$. By construction, $G$ contains the complete Euclidean graph on the points of $A_i \cup B_i \cup \{c'_1, c'_2, \ldots, c'_k\}$.

Let $c_1, c_2, \ldots, c_k$ be the sequence of $k$ exact neighbors of the pair $(a,b)$. These points are numbered such that $|ac_j| + |c_j b| \leq |ac_{j+1}| + |c_{j+1} b|$ for all $1 \leq j < k$.

Let $P$ be a shortest $pq$-path in the graph $K_S \setminus E'$. We will prove that $G \setminus E'$ contains a $pq$-path of length at most $t$ times the length of $P$.

First, we claim that there is a $pq$-path $P'$ in $K_S \setminus E'$, whose vertices are contained in $V := \{p, q, a, b, c_1, c_2, \ldots, c_k\}$, and that has length at most $(1 + 4/s)^2$ times the length of $P$.

Clearly, this is true if all vertices of $P$ are contained in $V$, because then we can take $P' = P$. So, assume that $P$ contains a vertex $v$ not belonging to $V$. Then, by the triangle inequality, $P$ has length at least $|pv| + |vq|$, which, by Lemma 2, is at least equal to $\frac{1}{1+4/s}(|av| + |vb|)$. Since $v \notin \{a, b, c_1, c_2, \ldots, c_k\}$, we have $|av| + |vb| \geq |ac_k| + |c_k b|$. Hence, the length of $P$ satisfies

$$|P| \geq \frac{1}{1+4/s}(|ac_k| + |c_k b|).$$

Note that the set $\{p, q, a, b, c_1, c_2, \ldots, c_k\}$ has size at least $k+2$ (equality can occur). Therefore, by Lemma 5, the graph $K_S \setminus E'$ contains a $pq$-path $P'$

that consists of (i) the single edge $(p, q)$, (ii) two edges $(p, a)$ and $(a, q)$, (iii) two edges $(p, b)$ and $(b, q)$, or (iv) two edges $(p, c_j)$ and $(c_j, q)$ for some index $1 \leq j \leq k$.

If (i) holds, then $P'$ has length $|pq|$, which is clearly at most $(1+4/s)^2|P|$. Assume that (ii) holds. Since the sets $A_i$ and $B_i$ are well-separated, there are two disjoint balls $C_{A_i}$ and $C_{B_i}$ having the same radius, say $\rho$, that contain the bounding rectangles $R(A_i)$ and $R(B_i)$, respectively, and that have distance at least $s\rho$. In particular, we have $|ap| \leq 2\rho$ and $|pq| \geq s\rho$. This, together with the triangle inequality, implies that

$$|P'| \leq |pa| + |ap| + |pq| \leq 4\rho + |pq| \leq (1 + 4/s)\,|pq| \leq (1 + 4/s)^2\,|P|.$$

If (iii) holds, then it follows in exactly the same way that $|P'| \leq (1+4/s)^2|P|$.

Assume that (iv) holds. Then $P'$ has length $|pc_j| + |c_jq|$, which, by Lemma 2, is at most equal to $(1 + 4/s)\,(|ac_j| + |c_jb|)$. Hence, since $j \leq k$,

$$|P'| \leq (1 + 4/s)\,(|ac_k| + |c_kb|) \leq (1 + 4/s)^2\,|P|.$$

This proves the claim.

In the rest of the proof, we will show that $G \setminus E'$ contains a $pq$-path $Q$ of length at most $t$ times the length of $P$. If all vertices of $P'$ are contained in the set $\{p, q, a, b, c'_1, c'_2, \ldots, c'_k\}$, then $P'$ is a $pq$-path in the graph $G \setminus E'$, and we take $Q = P'$. We have

$$s \geq \frac{4}{t^{1/12} - 1} \geq \frac{4}{\sqrt{t} - 1},$$

which implies that $(1 + 4/s)^2 \leq t$. Hence, in this case, $|Q| \leq t|P|$.

Assume that the path $P'$ contains a vertex $w$ that is not contained in $\{p, q, a, b, c'_1, c'_2, \ldots, c'_k\}$. Let $l$ be the index such that $w = c_l$. Applying the triangle inequality, we see that

$$|P'| \geq |pw| + |wq| = |pc_l| + |c_lq|.$$

Note that the set $\{p, q, a, b, c'_1, c'_2, \ldots, c'_k\}$ has size at least $k + 2$, and that $G$ contains the complete graph on this set. Therefore, by Lemma 5, the graph $G \setminus E'$ contains a $pq$-path $Q$ consisting of (i) the edge $(p, q)$, (ii) two edges $(p, a)$ and $(a, q)$, (iii) two edges $(p, b)$ and $(b, q)$, or (iv) two edges $(p, c'_j)$ and $(c'_j, q)$, for some index $j$.

If (i) holds, then $Q$ has length at most $t$ times the length of $P$. If (ii) holds, then $Q$ has length $|pa| + |aq|$. As in (ii) above, it follows that

$$|Q| \leq (1 + 4/s)\,|pq| \leq (1 + 4/s)\,|P|.$$

Since $s \geq (8t+4)/(t-1)$, a straightforward calculation shows that $1+4/s \leq t$. That is, path $Q$ has length at most $t|P|$. If (iii) holds, then it follows in the same way that the path $Q$ has length at most $t|P|$.

Finally, assume that (iv) holds. If $j \leq l$, then

$$\begin{aligned}
|Q| &= |pc'_j| + |c'_j q| \\
&\leq (1 + 4/s)\,(|ac'_j| + |c'_j b|) \\
&\leq (1 + 4/s)\,(|ac'_l| + |c'_l b|) \\
&\leq t^{1/3}(1 + 4/s)\,(|ac_l| + |c_l b|) \\
&\leq t^{1/3}(1 + 4/s)^2\,(|pc_l| + |c_l q|) \\
&\leq t^{1/3}(1 + 4/s)^2\,|P'| \\
&\leq t^{2/3}(1 + 4/s)^2\,|P'| \\
&\leq t^{2/3}(1 + 4/s)^4\,|P|.
\end{aligned}$$

Note that we have chosen $s$ such that $s \geq 4/(t^{1/12} - 1)$, which is equivalent to $(1 + 4/s)^4 \leq t^{1/3}$. Therefore, the length of $Q$ is at most $t$ times the length of $P$. It remains to consider the case when $l < j$. We claim that

$$|ac'_j| + |c'_j b| \leq t^{1/3}\,(|ac'_l| + |c'_l b|).$$

Indeed, if this is not true, then the third condition in Definition 4 implies that $w = c_l \in \{c'_1, c'_2, \ldots, c'_k\}$, which is a contradiction. Hence,

$$\begin{aligned}
|Q| &= |pc'_j| + |c'_j q| \\
&\leq (1 + 4/s)\,(|ac'_j| + |c'_j b|) \\
&\leq t^{1/3}(1 + 4/s)\,(|ac'_l| + |c'_l b|) \\
&\leq t^{2/3}(1 + 4/s)\,(|ac_l| + |c_l b|) \\
&\leq t^{2/3}(1 + 4/s)^2\,(|pc_l| + |c_l q|) \\
&\leq t^{2/3}(1 + 4/s)^2\,|P'| \\
&\leq t^{2/3}(1 + 4/s)^4\,|P|.
\end{aligned}$$

Then in the same way as above, we obtain $|Q| \leq t|P|$. ∎

Now we can give the complete proof of the fact that the graph $G \setminus E'$ contains a $pq$-path of length at most $t$ times the length of a shortest $pq$-path in $K_S \setminus E'$. Recall that $i$ is the index such that $p \in A_i$ and $q \in B_i$. We assume without loss of generality that $|A_i| \geq |B_i|$.

The proof is by induction on the rank of the distance $|pq|$ in the sorted sequence of $\binom{n}{2}$ distances in $S$. First assume that $p, q$ is a closest pair in $S$. Then, by Lemma 1, $A_i = \{p\}$ and $B_i = \{q\}$, and the claim follows from Lemma 6.

Assume from now on that $p, q$ is not a closest pair. Moreover, assume that for any pair $a, b \in S$ with $|ab| < |pq|$, the graph $G \setminus E'$ contains an $ab$-path of length at most $t$ times the length of a shortest $ab$-path in $K_S \setminus E'$. Since the sets $A_i$ and $B_i$ are well-separated, there are two balls $C_{A_i}$ and $C_{B_i}$ having the same radius, say $\rho$, that contain the bounding rectangles $R(A_i)$ and $R(B_i)$, respectively, and that have distance at least $s\rho$.

We distinguish four cases.

### 4.1.1 Case 1: $|B_i| \geq k + 1$

Consider the $k+1$ points $a_j \in A_i$, $1 \leq j \leq k+1$, and the $k+1$ points $b_j \in B_i$, $1 \leq j \leq k + 1$, that were chosen in the construction of $G$.

**Lemma 7** *There is an index $j$, $1 \leq j \leq k + 1$, such that the graph $G \setminus E'$ contains*

1. *the edge $(a_j, b_j)$,*

2. *a path $P$ between $p$ and $a_j$ of length at most $4t\rho$, and*

3. *a path $Q$ between $q$ and $b_j$ of length at most $4t\rho$.*

**Proof.** We will first prove that for each $j$ such that $(a_j, b_j)$ is an edge in $G \setminus E'$, there is a $pa_j$-path $P_j$ in $G \setminus E'$ of length at most $4t\rho$.

First assume that $p \notin \{a_1, a_2, \ldots, a_{k+1}\}$. Since $E'$ has size at most $k$, there is an index $j$, such that $(a_j, b_j)$ is an edge in $G \setminus E'$. Let $j$ be an arbitrary index having this property.

Since $|pq| \geq s\rho$, $|pa_j| \leq 2\rho$, and $s > 2$, we must have $|pa_j| < |pq|$. Therefore, by the induction hypothesis, the graph $G \setminus E'$ contains a path $P_j$ between $p$ and $a_j$ having length at most $t$ times the length of a shortest $pa_j$-path in $K_S \setminus E'$. Note that such a path may contain vertices that do not belong to $A_i$.

Since $A_i$ contains at least $k+2$ points, Lemma 5 implies that the graph $K_{A_i} \setminus E'$ contains a $pa_j$-path $P'$ having at most two edges. Each edge on this path is contained in the graph $K_S \setminus E'$, and has length at most $2\rho$. Hence, $P'$ has length at most $4\rho$. It follows that the shortest $pa_j$-path in $K_S \setminus E'$ has length at most $4\rho$. Therefore, path $P_j$ has length at most $4t\rho$.

Next assume that $p \in \{a_1, a_2, \ldots, a_{k+1}\}$. Let $l$ be the index such that $p = a_l$. If $G \setminus E'$ contains the edge $(a_l, b_l)$, then we take for $P_l$ the empty $pa_l$-path, having length zero. So assume that $(a_l, b_l)$ is not an edge of $G \setminus E'$. Let $j$ be an arbitrary index such that $G \setminus E'$ contains the edge $(a_j, b_j)$. The induction hypothesis implies that $G \setminus E'$ contains a path $P_j$ between $p$ and $a_j$ having length at most $t$ times the length of a shortest $pa_j$-path in $K_S \setminus E'$.

Since $E'$ contains $(a_l, b_l)$, at most $k-1$ edges of the subgraph $K_{A_i}$ are contained in $E'$. Then, since $A_i$ contains at least $k+1$ points, Lemma 5 implies that the graph $K_{A_i} \setminus E'$ contains a $pa_j$-path having at most two edges. Each edge on this path is contained in $K_S \setminus E'$, and has length at most $2\rho$. Therefore, path $P_j$ has length at most $4t\rho$.

This completes the proof of the claim mentioned in the beginning of this proof. Now assume that $q \notin \{b_1, b_2, \ldots, b_{k+1}\}$. Let $j$ be an arbitrary index such that $(a_j, b_j)$ is an edge in $G \setminus E'$. Then, in exactly the same way as above, it can be shown that $G \setminus E'$ contains a $qb_j$-path $Q_j$, having length at most $4t\rho$. Hence, the index $j$, and the paths $P_j$ and $Q_j$ satisfy the three conditions mentioned in the lemma.

Finally, we consider the case when $q \in \{b_1, b_2, \ldots, b_{k+1}\}$. Let $l$ be the index such that $q = b_l$. If $(a_l, b_l)$ is an edge in $G \setminus E'$, then we let $Q_l$ be the empty $qb_l$-path, having length zero. In this case, the index $l$, and the paths $P_l$ and $Q_l$ satisfy the three conditions of the lemma. Assume that $(a_l, b_l)$ is not an edge in $G \setminus E'$. In this case, let $j$ be an arbitrary index such that $G \setminus E'$ contains $(a_j, b_j)$. In exactly the same way as above, it follows that $G \setminus E'$ contains a $qb_j$-path $Q_j$, having length at most $4t\rho$. Hence, the index $j$, and the paths $P_j$ and $Q_j$ satisfy the three conditions of the lemma. ∎

We can now complete the proof for Case 1. Consider the index $j$, and the corresponding paths $P$ and $Q$, of Lemma 7. Let $R$ be the $pq$-path in $G \setminus E'$ obtained by concatenating path $P$, edge $(a_j, b_j)$, and path $Q$. We will show that $|R| \leq t|pq|$.

First note that $|R| \leq 8t\rho + |a_j b_j|$. The triangle inequality implies that $|a_j b_j| \leq |a_j p| + |pq| + |qb_j|$. Furthermore, $|a_j p| \leq 2\rho$ and $|qb_j| \leq 2\rho$. Therefore,

$$|R| \leq (8t+4)\rho + |pq|.$$

Since $|pq| \geq s\rho$ and $s \geq (8t+4)/(t-1)$, it follows that $|R| \leq t|pq|$. Hence, the length of $R$ is surely at most $t$ times the length of a shortest $pq$-path in $K_S \setminus E'$. This completes the proof of Case 1.

### 4.1.2 Case 2: $|B_i| \leq k$ and $|A_i| \geq k+1$

Consider the $k+1$ points $a_j \in A_i$, $1 \leq j \leq k+1$, that were chosen in the construction of $G$. Let $b_j$, $1 \leq j \leq x = |B_i|$ be the points of $B_i$. Note that $q$ is one of the $b_j$'s. Also, in $G$, point $q$ is connected to each point $a_j$, $1 \leq j \leq k+1$.

First assume that $p \notin \{a_1, a_2, \ldots, a_{k+1}\}$. Let $j$ be an index such that $(a_j, q)$ is an edge of $G \setminus E'$. Since $A_i$ contains at least $k+2$ points, it follows in exactly the same way as in the proof of Lemma 7, that $G \setminus E'$ contains a $pa_j$-path $P$ of length at most $4t\rho$. Then, just as in Case 1, it can be shown that the path consisting of $P$, followed by edge $(a_j, q)$, is a $pq$-path in $G \setminus E'$ of length at most $t|pq|$. Hence, the length of this path is at most $t$ times the length of a shortest $pq$-path in $K_S \setminus E'$.

Next consider the case when $p \in \{a_1, a_2, \ldots, a_{k+1}\}$. Let $l$ be the index such that $p = a_l$. If $(p, q)$ is an edge in $G \setminus E'$, then this graph clearly contains a $pq$-path of length at most $t$ times the length of a shortest $pq$-path in $K_S \setminus E'$. Otherwise, $(p, q)$ is an edge in $E'$. Let $j$ be an index such that $G \setminus E'$ contains $(a_j, q)$. Note that $A_i$ contains at least $k+1$ points. Moreover, at most $k-1$ edges of the complete Euclidean graph $K_{A_i}$ are contained in $E'$. Therefore, it follows in the same way as in the proof of Lemma 7, that $G \setminus E'$ contains a $pa_j$-path $P$ of length at most $4t\rho$. The path consisting of $P$, followed by edge $(a_j, q)$, is a $pq$-path in $G \setminus E'$ of length at most $t$ times the length of a shortest $pq$-path in $K_S \setminus E'$.

### 4.1.3 Case 3: $|A_i| \leq k$ and $|A_i| + |B_i| \geq k+2$

In this case, $G$ contains the complete Euclidean graph on $A_i \cup B_i$ as a subgraph. Denote this complete subgraph by $K$. By Lemma 5, $K \setminus E'$ and, hence, $G \setminus E'$, contains a $pq$-path $P$ having at most two edges. This path either consists of the edge $(p, q)$, or it contains two edges $(p, x)$ and $(x, q)$ for some point $x \in (A_i \cup B_i) \setminus \{p, q\}$. It can be shown in the same way as in Case 1 that the length of $P$ is at most equal to $t|pq|$. Hence, $P$ has length at most $t$ times the length of a shortest $pq$-path in $K_S \setminus E'$.

### 4.1.4 Case 4: $|A_i| + |B_i| \leq k + 1$

This case has been treated already in Lemma 6.

We have proved the following result.

**Theorem 3** *Let $S$ be a set of $n$ points in $\mathbb{R}^d$, $k$ an integer such that $1 \leq k \leq n-2$, and $t > 1$ a real constant. Let*

$$\{A_1, B_1\}, \{A_2, B_2\}, \ldots, \{A_m, B_m\}$$

*be an arbitrary WSPD for $S$, with separation constant*

$$s = \max\left(\frac{8t+4}{t-1}, \frac{4}{t^{1/12}-1}\right).$$

*The graph $G = (S, E)$ defined above is a $(k,t)$-EFTS for $S$. This graph contains $O(k^2 m)$ edges.*

## 4.2 Constructing the graph $G$

The algorithm for constructing the graph $G$ follows immediately from the results of the previous sections. Given the set $S$, the positive integer $k$, and the real constant $t > 1$, we use the algorithm of [5], see Theorem 1, to compute a WSPD for $S$ of size $m = O(n)$. Then, for each pair $\{A_i, B_i\}$ in this WSPD, we construct the corresponding edge set $E_i$. If Case 1 applies, then we construct $E_i$ in $O(k)$ time. If Case 2 or 3 applies, then we need $O(k^2)$ time to construct $E_i$. Finally, we take all pairs $\{A_i, B_i\}$ for which Case 4 applies, choose arbitrary points $a_i \in A_i$ and $b_i \in B_i$, and use Theorem 2 to compute a sequence of $k$ strong $t^{1/3}$-approximate neighbors for each such pair $(a_i, b_i)$. Note that there are at most $m$ such pairs. Given these approximate neighbors for $(a_i, b_i)$, we can construct the set $E_i$ in $O(k^2)$ time.

**Theorem 4** *Let $S$ be a set of $n$ points in $\mathbb{R}^d$, $k$ an integer such that $1 \leq k \leq n-2$, and $t > 1$ a real constant.*

1. *There exists a $(k,t)$-EFTS for $S$ containing at most $\gamma_{dt} k^2 n$ edges. The value of $\gamma_{dt}$ only depends on $d$ and $t$, and if $t \downarrow 1$, then $\gamma_{dt} \sim (c/(t-1))^d$ for some constant $c$.*

2. *This $(k,t)$-EFTS can be computed in time*

$$O\left(\delta_{dt}\left(n\log^{d-1}n + kn\log\log n + k^2 n\right)\right).$$

*The value of $\delta_{dt}$ only depends on $d$ and $t$, and if $t \downarrow 1$, then $\delta_{dt} \sim (c/(t-1))^{2d-1}$ for some constant $c$.*

**Proof.** Consider the constants $\alpha_{ds}$ and $\beta_{dt}$ in Theorems 1 and 2. Let $t_0 := t^{1/3}$. Then the total running time of the algorithm for constructing $G$ is proportional to

$$\beta_{dt_0}\cdot\alpha_{ds}\left(n\log^{d-1}n + kn\log\log n + k^2 n\right).$$

Also, the number of edges of $G$ is proportional to $\alpha_{ds}k^2n$.

For $t \downarrow 1$, we have $\alpha_{ds} \sim (c_1 s)^d$, $\beta_{dt_0} \sim (c_2/(t-1))^{d-1}$, and $s \sim c_3/(t-1)$, for some constants $c_1$, $c_2$, and $c_3$. ∎

# 5 A vertex fault-tolerant spanner

In this section, we define a graph $G$ that is based only on well-separated pairs, and that is a vertex fault-tolerant spanner. The construction is the same as in Section 4, except that Case 4 is much simpler.

Let $S$ be a set of $n$ points in $\mathbb{R}^d$, $t > 1$ a real constant, and $k$ an integer such that $1 \leq k \leq n-2$. Consider an arbitrary WSPD

$$\{A_1, B_1\}, \{A_2, B_2\}, \ldots, \{A_m, B_m\}$$

for $S$, with separation constant $s = 4(t+1)/(t-1)$.

As in Section 4, our graph $G$ has the points of $S$ as its vertices. For each $i$, $1 \leq i \leq m$, we will define a set $E_i$ of edges. Then, $E := \cup_i E_i$ will be the edge set of $G$. This set will have size $O(k^2 m)$.

Let $1 \leq i \leq m$, and consider the well-separated pair $\{A_i, B_i\}$. We assume without loss of generality that $|A_i| \geq |B_i|$. To define $E_i$, we distinguish the same four cases as in Section 4. In Case 1, 2, or 3, the edge set $E_i$ is defined as in Section 4. In Case 4, i.e., if $|A_i| + |B_i| \leq k+1$, then $E_i$ is the edge set of the complete Euclidean graph on the points of $A_i \cup B_i$.

The graph $G$ is a $(k,t)$-VFTS for the set $S$. The proof of this claim is basically the same as in Section 4.1, except that the generalization of

Lemma 6 to vertex tolerance trivially holds. Therefore, we leave the details to the reader.

To compute the graph $G$, we first compute a WSPD for $S$ of size $m = O(n)$. This takes $O(n \log n)$ time. Given this WSPD, we can easily construct $G$ in $O(k^2 m)$ time. Hence, the entire algorithm takes $O(n \log n + k^2 m)$ time. Using Theorem 1, the constant factors can easily be estimated.

**Theorem 5** *Let $S$ be a set of $n$ points in $\mathbb{R}^d$, $k$ an integer such that $1 \leq k \leq n-2$, and $t > 1$ a real constant.*

1. *There exists a $(k,t)$-VFTS for $S$ containing at most $\gamma_{dt} k^2 n$ edges. The value of $\gamma_{dt}$ only depends on $d$ and $t$, and if $t \downarrow 1$, then $\gamma_{dt} \sim (c/(t-1))^d$ for some constant $c$.*
2. *This $(k,t)$-VFTS can be computed in $O(n \log n + \gamma_{dt} k^2 n)$ time.*

# 6 Fault-tolerant spanners of low weight

We now turn to the problem of constructing fault-tolerant spanners whose total edge length is "small". First, we give a simple transformation that turns any spanner $G_0$ into a fault-tolerant spanner $G$. If the degree of $G_0$ is bounded by $D$, then the degree of $G$ is proportional to $D^{k+1}$. Moreover, in this case, the transformation increases the total edge length by at most a factor proportional to $k \cdot D^k$.

Let $S$ be a set of $n$ points in $\mathbb{R}^d$, $t > 1$ a real number, and $k$ an integer such that $2 \leq k \leq n-2$. Let $G_0$ be an arbitrary $t$-spanner for $S$. For each vertex $p \in S$, let $N(p)$ be the set of all vertices of $S \setminus \{p\}$ that are connected to $p$, in $G_0$, by a path consisting of at most $k+1$ edges. Define $E_p := \{(p,q) : q \in N(p)\}$. The transformed graph $G$ has the points of $S$ as its vertices, and it has edge set $E := \cup_{p \in S} E_p$. Note that $G_0$ is a subgraph of $G$. We claim that $G$ is a $(k,t)$-EFTS for $S$.

**Lemma 8** *Let $E'$ be a subset of $E$ having size at most $k$, and let $(p,q)$ be an edge of $K_S \setminus E'$. The graph $G \setminus E'$ contains a $pq$-path of length at most $t$ times the Euclidean distance between $p$ and $q$.*

**Proof.** Since $G_0$ is a $t$-spanner for $S$, there is a $pq$-path

$$P = (q_0 = p, q_1, q_2, \ldots, q_l = q)$$

in $G_0$ of length at most $t|pq|$. We will construct a $pq$-path $Q$ in $G \setminus E'$ that is a subpath of $P$. Then, the triangle inequality implies that the length of $Q$ is at most that of $P$. This will prove the lemma.

First assume that $l \leq k+1$. Then, $q \in N(p)$ and, hence, $(p,q)$ is an edge of $G$. Since $(p,q)$ is an edge of $K_S \setminus E'$, it is not contained in $E'$. Therefore, $(p,q)$ is an edge of $G \setminus E'$, and we can take for $Q$ the path consisting of this single edge.

Assume that $k+2 \leq l$. The following algorithm constructs the path $Q = (p_0, p_1, \ldots)$ incrementally.

**Step 1:** Define $p_0 := p$, $i := 0$, and $j := 0$. Go to Step 2.

**Step 2:** At this moment, $Q = (p_0, p_1, \ldots, p_i)$ is a path in $G \setminus E'$, $j$ is the index such that $p_i = q_j$, and $j + k + 2 \leq l$. (In particular, $p_i \neq q$.)

If there is an index $m$, $j+1 \leq m \leq j+k+1$, such that (i) $m+k+2 \leq l$ and (ii) $(q_j, q_m)$ is an edge of $G \setminus E'$, then go to Step 3. Otherwise, go to Step 4.

**Step 3:** Define $p_{i+1} := q_m$. Moreover, set $i := i+1$, and $j := m$, and go to Step 2.

**Step 4:** We know that $p_i = q_j$ and $j+k+2 \leq l$. Moreover, for all $m$, $j+1 \leq m \leq j+k+1$, such that $(q_j, q_m)$ is an edge of $G \setminus E'$, we have $m + k + 1 \geq l$.

We claim that there is an index $m$, $j+1 \leq m \leq j+k+1$, such that $(q_j, q_m)$ and $(q_m, q)$ are both edges of $G \setminus E'$.

Assume this claim is true. Then we define $p_{i+1} := q_m$ and $p_{i+2} := q$, and the construction of the $pq$-path $Q$ is complete.

It remains to prove the claim. First note that all $k+1$ edges $(q_j, q_m)$, $j+1 \leq m \leq j+k+1$, are contained in $G$. Let

$$x := |E' \cap \{(q_j, q_m) : j+1 \leq m \leq j+k+1\}| .$$

Let $M$ be the set of indices $m$, $j+1 \leq m \leq j+k+1$, such that $(q_j, q_m)$ is an edge of $G \setminus E'$. Then $M$ has size $k+1-x$. Let $m$ be an arbitrary element of $M$. Then our assumption implies that $m+k+1 \geq l$. Therefore, $q \in N(q_m)$, and $(q_m, q) \in G$.

Since the set $\{(q_m, q) : m \in M\}$ has size $k+1-x$, and at most $k-x$ edges of this set are contained in $E'$, there exists an index $m \in M$ such that $(q_m, q)$ is an edge of $G \setminus E'$. This proves the claim.

Why does this algorithm terminate? Each time Step 3 is executed, path $Q$ is extended by a new point. Therefore, at some moment, Step 4 must be executed. At that moment, $Q$ reaches $q$, and the algorithm terminates. ∎

**Lemma 9** *The graph $G$ is a $(k,t)$-EFTS for $S$.*

**Proof.** Let $E'$ be a subset of $E$ having size at most $k$, and let $p$ and $q$ be two distinct points of $S$. Let $P = (p_0 = p, p_1, p_2, \ldots, p_l = q)$ be a shortest $pq$-path in $K_S \setminus E'$. Then for each $i$, $0 \leq i < l$, $(p_i, p_{i+1})$ is an edge of $K_S \setminus E'$. Hence by Lemma 8, the graph $G \setminus E'$ contains a path $Q_i$ between $p_i$ and $p_{i+1}$ having length at most $t|p_i p_{i+1}|$. Let $Q$ be the concatenation of $Q_0, Q_1, \ldots, Q_{l-1}$. Then, $Q$ is a $pq$-path in $G \setminus E'$, having length

$$\sum_{i=0}^{l-1} |Q_i| \leq \sum_{i=0}^{l-1} t|p_i p_{i+1}| = t|P|.$$

This proves the lemma. ∎

**Lemma 10** *The graph $G$ is a $(k,t)$-VFTS for $S$.*

**Proof.** The proof is basically the same as those of Lemmas 8 and 9, and is therefore omitted. ∎

**Lemma 11** *Assume that each point of $S$ has degree at most $D$ in $G_0$. Then*

1. *each point of $S$ has degree at most $2 \cdot D^{k+1}$ in $G$, and*
2. *the total edge length of $G$ is at most $8(k+1) \cdot D^k$ times that of $G_0$.*

**Proof.** Let $p \in S$. Then

$$|N(p)| \leq D + D^2 + D^3 + \cdots + D^{k+1} \leq 2 \cdot D^{k+1}.$$

Since $q \in N(p)$ if and only if $p \in N(q)$, it follows that each point has degree at most $2 \cdot D^{k+1}$ in $G$.

To bound the total edge length of $G$, we use the following charging scheme. Let $(p,q)$ be any edge of $G$, and consider any $pq$-path $P = (p_0 = p, p_1, p_2, \ldots, p_l = q)$ in $G_0$ containing $l \leq k+1$ edges. (Note that $P$ exists.) We charge the length $|pq|$ of edge $(p,q)$ to the edges of $P$, in such a way that no edge $(p_i, p_{i+1})$, $0 \leq i < l$, is charged more than $|p_i p_{i+1}|$. Since $|pq| \leq |P|$, this is possible. We do this for all edges of $G$.

For each edge $e$ of $G_0$, let $n_e$ be the number of times this edge is charged. Then the total edge length of $G$ is at most equal to $\sum_{e \in G_0} n_e \cdot |e|$. We will show that $n_e \leq 8(k+1) \cdot D^k$. This will imply that the total edge length of $G$ is at most $8(k+1) \cdot D^k \sum_{e \in G_0} |e|$, which is equal to $8(k+1) \cdot D^k$ times the total edge length of $G_0$.

Let $e$ be an edge of $G_0$, and let it have endpoints $a$ and $b$. Every time $e$ is charged, there are two points $p$ and $q$, such that there is a $pq$-path in $G_0$ containing at most $k+1$ edges, $e$ being one of them. Assume w.l.o.g. that $a$ occurs before $b$ on this path. Let $i$ be the number of edges on the subpath from $p$ to $a$. Then $0 \leq i \leq k$. If $j$ denotes the number of edges on the subpath from $b$ to $q$, then $0 \leq j \leq k - i$.

If we fix $i$ and $j$, then the number of possibilities for $p$ is at most

$$D + D^2 + D^3 + \cdots + D^i \leq 2 \cdot D^i,$$

and the number of possibilities for $q$ is at most

$$D + D^2 + D^3 + \cdots + D^j \leq 2 \cdot D^j.$$

It follows that

$$\begin{aligned} n_e &\leq \sum_{i=0}^{k} 2 \cdot D^i \sum_{j=0}^{k-i} 2 \cdot D^j \\ &= 4 \sum_{i=0}^{k} D^i \left(1 + D + D^2 + \cdots + D^{k-i}\right) \\ &\leq 8 \sum_{i=0}^{k} D^i \cdot D^{k-i} \\ &= 8(k+1)D^k. \end{aligned}$$

∎

Now we can prove the main result of this section.

**Theorem 6** *Let $S$ be a set of $n$ points in $\mathbb{R}^d$, $k$ a positive integer, and $t > 1$ a real constant. There exists a $(k,t)$-EFTS $G$ for $S$*

1. *in which each point has degree at most $\alpha^{dk}$, for some constant $\alpha$ that only depends on $t$, and*

2. *whose total edge length is at most $\alpha^{dk}$ times the weight of a minimum spanning tree of $S$.*

*This graph $G$ can be computed in $O(n \log n + \alpha^{dk} n)$ time. Moreover, $G$ is also a $(k,t)$-VFTS for $S$. If $t \downarrow 1$, then $\alpha \sim c/(t-1)$ for some constant $c$.*

**Proof.** In [2], it is shown that in $O(n \log n + \beta_{dt} n)$ time, a $t$-spanner $G_0$ can be constructed whose degree $D$ is bounded by $\beta_{dt}$, and whose total edge length is proportional to $\beta_{dt}$ times the weight of a minimum spanning tree of $S$. The value of $\beta_{dt}$ only depends on $d$ and $t$, and if $t \downarrow 1$, then $\beta_{dt} \sim (c/(t-1))^d$ for some constant $c$.

The definition of $G$ immediately leads to an algorithm for constructing it from $G_0$, in time $O(|N(p)| \cdot n) = O(D^{k+1} n)$. ∎

# 7 Concluding remarks

We have presented efficient algorithms for constructing spanners that are resilient to $k$ edge or vertex faults. If $k$ is a constant, then the best result is that of Theorem 6. It gives a $k$-fault-tolerant $t$-spanner, in which the degree of each vertex is bounded by a constant, and whose weight is proportional to the weight of a minimum spanning tree. Moreover, this spanner can be constructed in $O(n \log n)$ time. Chen et al.[6] showed that constructing a $t$-spanner—that is not necessarily resilient to faults—takes $\Omega(n \log n)$ time in the algebraic computation tree model. Therefore, the result of Theorem 6 is optimal for constant values of $k$.

Some interesting problems remain to be solved. Any graph on $n$ vertices that remains connected after removing at most $k$ edges or vertices, must have $\Omega(kn)$ edges. The reason is that each vertex in such a graph must have degree larger than $k$. Is it possible to construct a $(k,t)$-EFTS or $(k,t)$-VFTS with $O(kn)$ edges, in $O(n \log n + kn)$ time?

Let $k$ be an even integer, and consider a set $A$ of $1 + k/2$ points that are all very close to the origin. Let $B$ be a set of $n - 1 - k/2$ points that are all very close together, but at distance roughly one from the origin. Let $G$ be any $(k,t)$-VFTS for the set $S := A \cup B$, where $t$ is a constant close to one. Then, every point of $A$ has to be connected to at least $1 + k/2$ points of $B$. Hence, $G$ contains $\Omega(k^2)$ edges having length roughly equal to one. The same argument shows that any $(k,t)$-EFTS contains $\Omega(k^2)$ edges having length roughly equal to one. On the other hand, a minimum spanning tree of $S$ has weight roughly one.

Is it possible to construct, for any set $S$ of $n$ points in $\mathbb{R}^d$, a $(k,t)$-EFTS or $(k,t)$-VFTS, such that each vertex has degree $O(k)$, and the weight of this graph is $O(k^2)$ times the weight of a minimum spanning tree of $S$? Can such a graph be constructed in $O(n \log n + kn)$ time?

# References


[1] P. K. Agarwal and J. Matoušek. Ray shooting and parametric search. *SIAM J. Comput.*, 22:794–806, 1993.

[2] S. Arya, G. Das, D. M. Mount, J. S. Salowe, and M. Smid. Euclidean spanners: short, thin, and lanky. In *Proc. 27th Annu. ACM Sympos. Theory Comput.*, pages 489–498, 1995.

[3] S. Arya, D. M. Mount, and M. Smid. Dynamic algorithms for geometric spanners of small diameter: randomized solutions. Report MPI-I-94-156, Max-Planck-Institut für Informatik, Saarbrücken, Germany, 1994.

[4] P. B. Callahan and S. R. Kosaraju. Faster algorithms for some geometric graph problems in higher dimensions. In *Proc. 4th ACM-SIAM Sympos. Discrete Algorithms*, pages 291–300, 1993.

[5] P. B. Callahan and S. R. Kosaraju. A decomposition of multidimensional point sets with applications to $k$-nearest-neighbors and $n$-body potential fields. *J. ACM*, 42:67–90, 1995.

[6] D. Z. Chen, G. Das, and M. Smid. Lower bounds for computing geometric spanners and approximate shortest paths. In *Proc. 8th Canad. Conf. Comput. Geom.*, pages 155–160, 1996.

[7] K. L. Clarkson. Approximation algorithms for shortest path motion planning. In *Proc. 19th Annu. ACM Sympos. Theory Comput.*, pages 56–65, 1987.

[8] G. Das and G. Narasimhan. A fast algorithm for constructing sparse Euclidean spanners. In *Proc. 10th Annu. ACM Sympos. Comput. Geom.*, pages 132–139, 1994.

[9] J. M. Keil and C. A. Gutwin. Classes of graphs which approximate the complete Euclidean graph. *Discrete Comput. Geom.*, 7:13–28, 1992.

[10] G. S. Lueker. A data structure for orthogonal range queries. In *Proc. 19th Annu. IEEE Sympos. Found. Comput. Sci.*, pages 28–34, 1978.

[11] K. Mehlhorn and S. Näher. Dynamic fractional cascading. *Algorithmica*, 5:215–241, 1990.

[12] J. Ruppert and R. Seidel. Approximating the $d$-dimensional complete Euclidean graph. In *Proc. 3rd Canad. Conf. Comput. Geom.*, pages 207–210, 1991.

[13] J. S. Salowe. Constructing multidimensional spanner graphs. *Internat. J. Comput. Geom. Appl.*, 1:99–107, 1991.

[14] P. M. Vaidya. A sparse graph almost as good as the complete graph on points in $K$ dimensions. *Discrete Comput. Geom.*, 6:369–381, 1991.

[15] A. C. Yao. On constructing minimum spanning trees in $k$-dimensional spaces and related problems. *SIAM J. Comput.*, 11:721–736, 1982.