Fault-Tolerant Additive Weighted Geometric Spanners

Sukanya Bhattacharjee[1] and R. Inkulu[1*]

Department of Computer Science & Engineering
IIT Guwahati, India
{bsukanya,rinkulu}@iitg.ac.in

Abstract. Let S be a set of n points and let w be a function that assigns non-negative weights to points in S. The additive weighted distance $d_w(p, q)$ between two points $p, q \in S$ is defined as $w(p) + d(p, q) + w(q)$ if $p \neq q$ and it is zero if $p = q$. Here, $d(p, q)$ is the geodesic Euclidean distance between p and q. For a real number $t > 1$, a graph $G(S, E)$ is called a *t-spanner* for the weighted set S of points if for any two points p and q in S the distance between p and q in graph G is at most $t.d_w(p, q)$ for a real number $t > 1$. For some integer $k \geq 1$, a t-spanner G for the set S is a (k, t)-*vertex fault-tolerant additive weighted spanner*, denoted with (k, t)-VFTAWS, if for any set $S' \subset S$ with cardinality at most k, the graph $G \setminus S'$ is a t-spanner for the points in $S \setminus S'$. For any given real number $\epsilon > 0$, we present algorithms to compute a $(k, 4 + \epsilon)$-VFTAWS for the metric space (S, d_w) resulting from the points in S belonging to any of the following: $\mathbb{R}^d$, simple polygon, polygonal domain, and terrain. Note that $d(p, q)$ is the geodesic Euclidean distance between p and q in the case of simple polygons and terrains whereas in the case of $\mathbb{R}^d$ it is the Euclidean distance along the line segment joining p and q.

Keywords: Computational Geometry, Geometric Spanners, Approximation Algorithms

1 Introduction

When designing geometric networks on a given set of points in a metric space, it is desirable for the network to have short paths between any pair of nodes while being sparse with respect to the number of edges. Let $G(S, E)$ be an edge-weighted geometric graph on a set S of n points in $\mathbb{R}^d$. The weight of any edge $(p, q) \in E$ is the Euclidean distance $|pq|$ between p and q. The distance in G between any two nodes p and q, denoted by $d_G(p, q)$, is defined as the length of a shortest (that is, minimum-weighted) path between p and q in G. The graph G is called a *t-spanner* for some $t \geq 1$ if for any two points $p, q \in S$ we have $d_G(p, q) \leq t.|pq|$. The smallest t for which G is a t-spanner is called the *stretch factor* of G, and the number of edges of G is called its size.

Peleg and Schäffer [40] introduced spanners in the context of distributed computing and by Chew [27] in a geometric context. Althöfer et al. [9] first attempted to study sparse spanners on edge-weighted graphs that have the triangle-inequality property. The text by Narasimhan and Smid [39], handbook chapter [31], and Gudmundsson and Knauer [32] detail various results on Euclidean spanners, including a $(1 + \epsilon)$-spanner for the set S of n points in $\mathbb{R}^d$ that has $O(\frac{n}{\epsilon^{d-1}})$ edges for any $\epsilon > 0$.

Many variations of sparse spanners have been studied, including spanners of low degree [11,25,18,42], spanners of low weight [17,30,33], spanners of low diameter [14,13], planar spanners [10,27,29,36], spanners of low chromatic number [16], fault-tolerant spanners [3,28,37,38,35,43], low power spanners [7,41,45], kinetic spanners [2,5], angle-constrained spanners [26], and combinations of these [12,15,21,22,23,24]. When the doubling dimension of a metric space is bounded, results applicable to the Euclidean settings are given in [44].

As mentioned in Abam et al., [4], the cost of traversing a path in a network is not only determined by the lengths of the edges on the path but also by the delays occurring at the nodes on the path. The result in [4] models these delays with the additive weighted metric. Let S be a set of n points in $\mathbb{R}^d$. For every $p \in S$, let $w(p)$ be the non-negative weight associated to p. The following additive weighted distance function d_w

on S defining the metric space (S, d_w) is considered in [4]: for any $p, q \in S$, $d_w(p, q)$ equals to 0 if $p = q$; otherwise, it is equal to $w(p) + |pq| + w(q)$.

Recently, Abam et al. [6] showed that there exists a $(2 + \epsilon)$-spanner with a linear number of edges for the metric space (S, d_w) that has bounded doubling dimension in [34].And, [4] gives a lower bound on the stretch factor, showing that $(2 + \epsilon)$ stretch is nearly optimal. Bose et al. [19] studied the problem of computing spanners for a weighted set of points. They considered the points that lie on the plane to have positive weights associated to them; and defined the distance d_w between any two distinct points $p, q \in S$ as $d(p, q) - w(p) - w(q)$. Under the assumption that the distance between any pair of points is non-negative, they showed the existence of a $(1 + \epsilon)$-spanner with $O(\frac{n}{\epsilon})$ edges.

A simple polygon $P_{\mathcal{D}}$ containing $h \geq 0$ number of disjoint simple polygonal holes within it is termed the *polygonal domain* $\mathcal{D}$. (When h equals to 0, the polygonal domain $\mathcal{D}$ is a simple polygon.) The free space $\mathcal{F}(\mathcal{D})$ of the given polygonal domain $\mathcal{D}$ is defined as the closure of $P_{\mathcal{D}}$ excluding the union of the interior of polygons contained in $P_{\mathcal{D}}$. Essentially, a path between any two given points in $\mathcal{F}(\mathcal{D})$ needs to be in the free space $\mathcal{F}(\mathcal{D})$ of $\mathcal{D}$. Given a set S of n points in the free space $\mathcal{F}(\mathcal{D})$ defined by the polygonal domain $\mathcal{D}$, computing geodesic spanners in $\mathcal{F}(\mathcal{D})$ is considered in Abam et al. [1]. For any two distinct points $p, q \in S$, $d_\pi(p, q)$ is defined as the geodesic Euclidean distance along a shortest path $\pi(p, q)$ between p and q in $\mathcal{F}(\mathcal{D})$. [1] showed that for the metric space (S, π), for any constant $\epsilon > 0$, there exists a $(5 + \epsilon)$-spanner of size $O(\sqrt{h}n(\lg n)^2)$. Further, for any constant $\epsilon > 0$, [1] gave a $(\sqrt{10} + \epsilon)$-spanner with $O(n(\lg n)^2)$ edges when $h = 0$ i.e., the polygonal domain is a simple polygon with no holes. Given a set S of n points on a polyhedral terrain $\mathcal{T}$, the geodesic Euclidean distance between any two points $p, q \in S$ is the distance along any shortest path between p and q on $\mathcal{T}$. [6] showed that for a set of unweighted points on a polyhedral terrain, for any constant $\epsilon > 0$, there exists a $(2 + \epsilon)$-geodesic spanner with $O(n \lg n)$ edges.

For a network to be vertex fault-tolerant, i.e., when a subset of nodes is removed, the induced network on the remaining nodes requires to be connected. Formally, a graph $G(S, E)$ is a *k-vertex fault-tolerant t-spanner*, denoted by (k, t)-VFTS, for a set S of n points in $\mathbb{R}^d$ if for any subset S' of S with size at most k, the graph $G \setminus S'$ is a t-spanner for the points in $S \setminus S'$. Algorithms in Levcopoulos et al., [37], Lukovszki [38], and Czumaj and Zhao [28] compute a (k, t)-VFTS for the set S of points in $\mathbb{R}^d$. These algorithms are also presented in [39]. [37] devised an algorithm to compute a (k, t)-VFTS of size $O(\frac{n}{(t-1)^{(2d-1)(k+1)}})$ in $O(\frac{n \lg n}{(t-1)^{4d-1}} + \frac{n}{(t-1)^{(2d-1)(k+1)}})$ time and another algorithm to compute a (k, t)-VFTS with $O(k^2 n)$ edges in $O(\frac{kn \lg n}{(t-1)^d})$ time. [38] gives an algorithm to compute a (k, t)-VFTS of size $O(\frac{kn}{(t-1)^{d-1}})$ in $O(\frac{1}{(t-1)^d}(n \lg^{d-1} n \lg k + kn \lg \lg n))$ time. The algorithm in [28] computes a (k, t)-VFTS having $O(\frac{kn}{(t-1)^{d-1}})$ edges in $O(\frac{1}{(t-1)^{d-1}}(kn \lg^d n + nk^2 \lg k))$ time with total weight of edges upper bounded by a $O(\frac{k^2 \lg n}{(t-1)^d})$ multiplicative factor of the weight of MST of the given set of points.

For a real number $t > 1$, a graph $G(S, E)$ is called a *t-spanner* for the weighted set S of points if for any two points p and q in S the distance between p and q in graph G is at most $t.d_w(p, q)$ for a real number $t > 1$. For some integer $k \geq 1$, a t-spanner G for the set S is a (k, t)-*vertex fault-tolerant additive weighted spanner*, denoted with (k, t)-VFTAWS, if for any set $S' \subset S$ with cardinality at most k, the graph $G \setminus S'$ is a t-spanner for the points in $S \setminus S'$.

Our results. The spanners computed in this paper are first of their kind as we combine fault-tolerance with the additive weighted set of points. We devise the following algorithms for computing vertex fault-tolerant additive weighted geometric spanners (VFTAWS) for any $\epsilon > 0$ and $k \geq 1$:

* Given a set S of n weighted points in $\mathbb{R}^d$, our first algorithm presented herewith computes a $(k, 4 + \epsilon)$-VFTAWS having $O(kn)$ edges. We incorporate fault-tolerance to the recent results of [6] while retaining the same stretch factor and increasing the number of edges in the spanner by a multiplicative factor of $O(k)$.

* Given a set S of n weighted points in a simple polygon, we present an algorithm to compute a $(k, 4 + \epsilon)$-VFTAWS that has $O(\frac{kn}{\epsilon^2} \lg n)$ edges. Our algorithm combines the clustering based algorithms from [1] and [4], and with the careful addition of more edges, we show that k fault-tolerance is achieved.

* Given a set S of n weighted points in a polygonal domain, we compute a $(k, 4 + \epsilon)$-VFTAWS having $O(\frac{k\sqrt{h}n}{\epsilon^2} \lg n)$ edges. We extend the structures used in [1] to achieve the vertex fault-tolerance.

* Given a set S of n weighted points on a terrain, we computes a $(k, 4 + \epsilon)$-VFTAWS having $O(\frac{kn}{\epsilon^2} \lg n)$ edges. For achieving the fault-tolerance, our algorithm adds a minimal set of edges to the spanner constructed in [6].

Unless specified otherwise, the points are assumed to be in Euclidean space $\mathbb{R}^d$. The Euclidean distance between two points p and q is denoted by $|pq|$. The distance between two points p, q in the metric space X is denoted by $d_X(p, q)$. The length of the shortest path between p and q in a graph G is denoted by $d_G(p, q)$.

Section 2 details the algorithm and its analysis to compute a $(k, 4+\epsilon)$-VFTAWS when the input weighted points are in $\mathbb{R}^d$. For the input weighted points located in a simple polygon, Section 3 describes an algorithm to compute a $(k, 4 + \epsilon)$-VFTAWS. Section 4 details algorithm for $(k, 4 + \epsilon)$-VFTAWS for a points located in the polygonal domain. Section 5 presents an algorithm to compute a $(k, 4 + \epsilon)$-VFTAWS when the input points are located on a terrain. Conclusions are in Section 6.

2 Vertex fault-tolerant additive weighted spanner for points in $\mathbb{R}^d$

In this section, we describe an algorithm to compute a (k, t)-VFTAWS for the set S of n non-negative weighted points in $\mathbb{R}^d$, where $t > 1$ and $k \geq 1$ are real numbers. For any two points $p, q \in S$, the additive weighted distance $d_w(p, q)$ is defined as the $w(p) + |pq| + w(q)$. Following the algorithm mentioned in [6], we first partition all the points belonging to S into at least $k + 1$ clusters. For creating these clusters, the points in set S are sorted in non-decreasing order with respect to their weights. Then the first $k + 1$ points in this sorted list are chosen as the centers of $k + 1$ distinct clusters. As the algorithm progress, more points are added to these clusters as well as more clusters (with cluster centers) may also be created. In any iteration of the algorithm, for any point p in the remaining sorted list, among the current set of cluster centers, we determine the cluster center c_j nearest to p. Let C_j be the cluster to which c_j is the center. It adds p to the cluster C_j if $|pc_j| \leq \epsilon.w(p)$; otherwise, a new cluster C_p with p as its centre is initiated. Let $C = \{c_1, \ldots, c_z\}$ be the final set of cluster centers obtained through this procedure. For every $i \in [1, z]$, the cluster to which c_i is the center is denoted by C_i. Using the algorithm from [43], we compute a $(k, (2 + \epsilon))$-VFTS $\mathcal{B}$ for the set C of cluster centers. We note that the degree of each vertex of $\mathcal{B}$ is $O(k)$. We denote the stretch of $\mathcal{B}$ by $t_\mathcal{B}$. First, the graph $\mathcal{G}$ is initialized to $\mathcal{B}$; further, points in $S \setminus C$ are included in $\mathcal{G}$ as vertices. Our algorithm to compute a $(k, 4 + \epsilon)$-VFTAWS differs from [6] with respect to both the algorithm used in computing $\mathcal{B}$ and the set of edges added to $\mathcal{B}$. The latter part is described now. For every $i \in [1, z]$, let C_i' be the set comprising of $\min\{k + 1, |C_i|\}$ least weighted points of cluster C_i. For each point $p \in S \setminus C$, if p belongs to cluster C_l, then for each $v \in B_l \cup C_l'$, our algorithm introduces an edge between p and v with weight $|pv|$ to

$\mathcal{G}$. Here, B_l is the set comprising of all the neighbors of the center (c_l) of cluster C_l in the graph $\mathcal{B}$. We state our algorithm to compute a $(k, (2+\epsilon))$-vertex fault tolerant spanner(VFTS) for the set S.

Algorithm 1: k-ADDITIVEFTS(S, k, ϵ)

Input : The set S of n additive weighted points in $\mathbb{R}^d$ and an integer $k \geq 1$ and a real number ϵ
        lying in the range $[\frac{\lg k}{\lg \frac{n}{k+1}+1}, 1]$ if $n \geq k(k+1)$ else ϵ is in the range $[\frac{k+1}{n}, 1]$

Output: $(k, (4+\epsilon))$-VFTS G

1: create the list P containing the points of S sorted in non-decreasing order of their weights
2: if $n \geq k(k+1)$, then create $\frac{n}{k^{\frac{1}{\epsilon}}}$ clusters; otherwise, create $\epsilon.n$ clusters
3: assign the first point $q \in P$ to be the centre c_1 of the first cluster C_1 and set z to 1
4: add q to set C of cluster centres
5: **for** every point $p \in P \setminus \{q\}$ **do**
6:     find the cluster centre c_j from the set C nearest to p
7:     **if** $|pc_j| \leq \epsilon.w(p)$ **then**
8:         add p to the cluster C_j
9:     **else**
10:         set z as $z + 1$
11:         create a new cluster C_z and assign p as the centre c_z of C_z
12:         add p to the set C
13:     **end if**
14: **end for**
15: construct a $(k, (1+\epsilon))$-VFTS $\mathcal{B}$ for the set C using the algorithm given in [43]
16: **for** every $p \in S \setminus C$ **do**
17:     find the cluster C_l to which p belongs; let c_l be the centre of C_l
18:     let C'_l be the set of $min\{k+1, card(C_l)\}$ least weighted points of cluster C_l; for every $p' \in C'_l$, introduce an edge between p and p'
19:     let B_l be the k nearest neighbors of c_l in $\mathcal{B}$; for every $p' \in B_l$, introduce an edge between p and p'
20: **end for**

In the following theorem, we prove that the graph $\mathcal{G}$ is indeed a $(k, 4+\epsilon)$-VFTAWS with $O(kn)$ edges.

Theorem 1. *Let S be a set of n weighted points in $\mathbb{R}^d$ with non-negative weights associated to points with weight function w. For any fixed constant $\epsilon > 0$, the graph $\mathcal{G}$ is a $(k, (4+\epsilon))$-VFTAWS with $O(kn)$ edges for the metric space (S, d_w).*

Proof: From [43], the number of edges in $\mathcal{B}$ is $O(k\,|C|)$, which is essentially $O(kn)$. Further, the degree of each node in $\mathcal{B}$ is $O(k)$. From each point in $S \setminus C$, we are adding at most $O(k)$ edges. Hence, the number of edges in $\mathcal{G}$ is $O(kn)$.

In proving that $\mathcal{G}$ is a $(k, 4+\epsilon)$-VFTAWS for the metric space (S, d_w), we show that for any set $S' \subset S$ with $|S'| \leq k$ and for any two points $p, q \in S \setminus S'$ there exists a $(4+\epsilon)$-spanner path between p and q in $\mathcal{G} \setminus S'$. Following are the possible cases based on the role p and q play with respect to clusters formed and their centers:

Case 1: Both p and q are cluster centres of two distinct clusters i.e., $p, q \in C$.
Since $\mathcal{B}$ is a $(k, (2+\epsilon))$-VFTS for the set C,

$$d_{\mathcal{G} \setminus S'}(p, q) = d_{\mathcal{B} \setminus S'}(p, q)$$
$$\leq t_{\mathcal{B}} \cdot d_w(p, q).$$

Case 2: Both p and q are in the same cluster C_i and one of them, w.l.o.g., say p, is the centre of C_i. Since p is the least weighted point in C_i, there exists an edge joining p and q in $\mathcal{G}$. Hence,

$$d_{\mathcal{G} \setminus S'}(p, q) = d_w(p, q).$$

Case 3: Both p and q are in the same cluster, say C_i; $p \neq c_i$, $q \neq c_i$; and, $c_i \notin S'$. Then,

$$
\begin{aligned}
d_{\mathcal{G}\setminus S'}(p,q) &= d_w(p,c_i) + d_w(c_i,q) \\
&= w(p) + |pc_i| + w(c_i) + w(c_i) + |c_iq| + w(q) \\
&\leq w(p) + \epsilon \cdot w(p) + w(c_i) + w(c_i) + \epsilon \cdot w(q) + w(q)
\end{aligned}
$$

[since a point x is added to cluster C_l only if $|xc_l| \leq \epsilon \cdot w(x)$]

$$
\leq w(p) + \epsilon \cdot w(p) + w(p) + w(q) + \epsilon \cdot w(q) + w(q)
$$

[since the points are sorted in the non-decreasing order of their weights and the first
point added to any cluster is taken as its center]

$$
\begin{aligned}
&= (2+\epsilon) \cdot [w(p) + w(q)] \\
&< (2+\epsilon) \cdot [w(p) + |pq| + w(q)] \\
&= (2+\epsilon) \cdot d_w(p,q).
\end{aligned}
$$

Case 4: Both p and q are in the same cluster, say C_i; $p \neq c_i$, $q \neq c_i$; and, $c_i \in S'$.
In the case of $|C_i| \leq k$, there exists an edge between p and q in $\mathcal{G}$. Hence, suppose that $|C_i| > k$. Let S'' be
the set of $k+1$ least weighted points from C_i. If $p,q \in S''$ then there exists an edge between p and q in $\mathcal{G}$.
If $p \in S''$ and $q \notin S''$ then as well there exists an edge between p and q. (Argument for the other case in
which $q \in S''$ and $p \notin S''$ is analogous.) Now consider the case in which both $p,q \notin S''$. Since p and q are
connected to every point in S'' and $|S''| = k+1$, there exists an $r \in S''$ such that $r \notin S'$ and the edges (p,r)
and (r,q) belong to $\mathcal{G} \setminus S'$. Therefore,

$$
\begin{aligned}
d_{\mathcal{G}\setminus S'}(p,q) &= d_w(p,r) + d_w(r,q) \\
&= w(p) + |pr| + w(r) + w(r) + |rq| + w(q) \\
&\leq w(p) + |pc_i| + |c_ir| + w(r) + w(r) + |rc_i| + |c_iq| + w(q)
\end{aligned}
$$

[by triangle inequality]

$$
\leq w(p) + \epsilon \cdot w(p) + \epsilon \cdot w(r) + w(r) + w(r) + \epsilon \cdot w(r) + \epsilon \cdot w(q) + w(q)
$$

[since a point x is added to cluster C_l only if $|xc_l| \leq \epsilon \cdot w(x)$]

$$
\leq w(p) + \epsilon \cdot w(p) + \epsilon \cdot w(p) + w(p) + w(q) + \epsilon \cdot w(q) + \epsilon \cdot w(q) + w(q)
$$

[since for any point the edges are added to the $k+1$ least weighted
points of the cluster to which it belongs]

$$
\begin{aligned}
&= (2+2\epsilon) \cdot [w(p) + w(q)] \\
&< (2+2\epsilon) \cdot [w(p) + |pq| + w(q)] \\
&= (2+2\epsilon) \cdot d_w(p,q).
\end{aligned}
$$

Case 5: Points p and q belong to two distinct clusters, say $p \in C_i$ and $q \in C_j$. In addition, $p \neq c_i$ and $q \neq c_j$,
and neither of the cluster centres belong to S'.
Then,

$$
\begin{aligned}
d_{\mathcal{G}\setminus S'}(p,q) &= d_w(p,c_i) + d_{\mathcal{B}}(c_i,c_j) + d_w(c_j,q) \\
&= w(p) + |pc_i| + w(c_i) + d_{\mathcal{B}}(c_i,c_j) + w(c_j) + |c_jq| + w(q) \\
&\leq w(p) + \epsilon \cdot w(p) + w(c_i) + d_{\mathcal{B}}(c_i,c_j) + w(c_j) + \epsilon \cdot w(q) + w(q)
\end{aligned}
$$

[since a point x is added to cluster C_l only if $|xc_l| \leq \epsilon \cdot w(x)$]

$$
\leq (1+\epsilon) \cdot [w(p) + w(q)] + w(c_i) + w(c_j) + t_{\mathcal{B}} \cdot d_w(c_i,c_j)
$$

[since $\mathcal{B}$ is a $(k, t_{\mathcal{B}})$-vertex fault-tolerant spanner for the set C]

5

$$\leq (1+\epsilon) \cdot [w(p) + w(q)] + w(p) + w(q) + t_{\mathcal{B}} \cdot d_w(c_i, c_j)$$

[since the points are sorted in the non-decreasing order of their weights and the first
point added to any cluster is taken as center of that cluster]

$$= (2+\epsilon) \cdot [w(p) + w(q)] + t_{\mathcal{B}} \cdot [w(c_i) + |c_i c_j| + w(c_j)]$$

$$\leq (2+\epsilon) \cdot [w(p) + w(q)] + t_{\mathcal{B}} \cdot [w(p) + |c_i c_j| + w(q)]$$

[since the points are sorted in the non-decreasing order of their weights and the first
point added to any cluster is taken as its center]

$$\leq (2+\epsilon) \cdot [w(p) + w(q)] + t_{\mathcal{B}} \cdot [w(p) + w(q) + |c_i p| + |pq| + |qc_j|]$$

[by triangle inequality]

$$\leq (2+\epsilon) \cdot [w(p) + w(q)] + t_{\mathcal{B}} \cdot [w(p) + w(q) + \epsilon \cdot w(p) + |pq| + \epsilon \cdot w(q)]$$

[since a point x is added to cluster C_l only if $|xc_l| \leq \epsilon \cdot w(x)$]

$$= (2+\epsilon) \cdot [w(p) + w(q)] + t_{\mathcal{B}} \cdot [(1+\epsilon) \cdot [w(p) + w(q)] + |pq|]$$

$$< (2+\epsilon) \cdot [w(p) + w(q) + |pq|] + t_{\mathcal{B}} \cdot (1+\epsilon) \cdot [w(p) + w(q) + |pq|]$$

$$< t_{\mathcal{B}}(2+\epsilon) \cdot [w(p) + w(q) + |pq|] \quad \text{when } t_B \geq (2+\epsilon)$$

[since each point has non-negative weight associated with it]

$$= t_{\mathcal{B}} \cdot (2+\epsilon) \cdot d_w(p, q).$$

Case 6: Both the points p and q are in two distinct clusters, w.l.o.g., say $p \in C_i$ and $q \in C_j$, one of them,
say p is the centre of C_i (i.e., $p = c_i$), and $c_j \notin S'$. Then,

$$d_{\mathcal{G} \setminus S'}(p, q) = d_{\mathcal{B}}(c_i, c_j) + d_w(c_j, q)$$

$$\leq t_{\mathcal{B}} \cdot d_w(c_i, c_j) + d_w(c_j, q)$$

[since $\mathcal{B}$ is a $(k, t_{\mathcal{B}})$-vertex fault-tolerant spanner for the set C]

$$= t_{\mathcal{B}} \cdot d_w(c_i, c_j) + w(c_j) + |c_j q| + w(q)$$

$$= t_{\mathcal{B}} \cdot [w(c_i) + |c_i c_j| + w(c_j)] + w(c_j) + |c_j q| + w(q)$$

$$\leq t_{\mathcal{B}} \cdot [w(p) + |c_i c_j| + w(q)] + w(q) + |c_j q| + w(q)$$

[since the points are sorted in the non-decreasing order of their weights and the first
point added to any cluster is taken as its center]

$$\leq t_{\mathcal{B}} \cdot [w(p) + |c_i c_j| + w(q)] + w(q) + \epsilon \cdot w(q) + w(q)$$

[since a point x is added to cluster C_l only if $|xc_l| \leq \epsilon \cdot w(x)$]

$$\leq t_{\mathcal{B}} \cdot [w(p) + |c_i p| + |pq| + |qc_j| + w(q)] + w(q) + \epsilon \cdot w(q) + w(q)$$

[by triangle inequality]

$$\leq t_{\mathcal{B}} \cdot [w(p) + \epsilon \cdot w(p) + |pq| + \epsilon \cdot w(q) + w(q)] + w(q) + \epsilon \cdot w(q) + w(q)$$

[since a point x is added to cluster C_l only if $|xc_l| \leq \epsilon . w(x)$]

$$= t_{\mathcal{B}} \cdot [(1+\epsilon) \cdot [w(p) + w(q)] + |pq|] + (2+\epsilon) \cdot w(q)$$

$$\leq t_{\mathcal{B}} \cdot [(1+\epsilon) \cdot [w(p) + w(q)] + |pq|] + (2+\epsilon) \cdot [w(p) + w(q) + |pq|]$$

[since each point has non-negative weight associated with it]

$$\leq t_{\mathcal{B}} \cdot (2+\epsilon) \cdot [w(p) + w(q) + |pq|] \quad \text{when } t_B \geq (2+\epsilon)$$

$$\leq t_{\mathcal{B}} \cdot (2+\epsilon) \cdot d_w(p, q).$$

Case 7: Both the points p and q are in two distinct clusters, say $p \in C_i$ and $q \in C_j$; $p \neq c_i$, $q \neq c_j$; and, one
of these centers, say c_j, belongs to S' and the other center $c_i \notin S'$. Since q is connected to all the neighbors

of c_j in $\mathcal{B}$, for any neighbor c_r of c_j with $c_r \in C$ and $c_r \notin S'$, the edge (q, c_r) belongs to $\mathcal{G} \setminus S'$. Therefore,

$$d_{\mathcal{G} \setminus S'}(p, q) = d_w(p, c_i) + d_{\mathcal{B}}(c_i, c_r) + d_w(c_r, q)$$

$$= w(p) + |pc_i| + w(c_i) + d_{\mathcal{B}}(c_i, c_r) + w(c_r) + |c_r q| + w(q)$$

$$\leq w(p) + \epsilon \cdot w(p) + w(c_i) + d_{\mathcal{B}}(c_i, c_r) + w(c_r) + |c_r q| + w(q)$$

[since a point x is added to cluster C_l only if $|xc_l| \leq \epsilon \cdot w(x)$]

$$\leq w(p) + \epsilon \cdot w(p) + w(c_i) + d_{\mathcal{B}}(c_i, c_r) + w(c_r) + |c_r c_j| + |c_j q| + w(q)$$

[by triangle inequality]

$$\leq w(p) + \epsilon \cdot w(p) + w(c_i) + d_{\mathcal{B}}(c_i, c_r) + w(c_r) + |c_r c_j| + \epsilon \cdot w(q) + w(q)$$

[since a point x is added to cluster C_l only if $|xc_l| \leq \epsilon \cdot w(x)$]

$$\leq (1 + \epsilon) \cdot [w(p) + w(q)] + w(c_i) + d_{\mathcal{B}}(c_i, c_r) + w(c_r) + |c_j c_r|. \tag{1}$$

Since $\mathcal{B}$ is a $(k, t_{\mathcal{B}})$-VFTS, there are at least $k + 1$ vertex disjoint $t_{\mathcal{B}}$-spanner paths between c_j and c_i in $\mathcal{B}$. Suppose c_r is the neighbor of c_j in $\mathcal{B}$ such that one of these $k + 1$ paths from c_j to c_i passes through c_r. We have the following:

$$d_{\mathcal{B}}(c_i, c_r) + d_w(c_r, c_j) < t_{\mathcal{B}} \cdot d_w(c_i, c_j)$$

$$\Rightarrow d_{\mathcal{B}}(c_i, c_r) + w(c_r) + |c_r c_j| + w(c_j) < t_{\mathcal{B}} \cdot d_w(c_i, c_j)$$

$$\Rightarrow d_{\mathcal{B}}(c_i, c_r) + |c_r c_j| + w(c_r) < t_{\mathcal{B}} \cdot d_w(c_i, c_j) - w(c_j). \tag{2}$$

Substituting (2) in (1),

$$d_{\mathcal{G} \setminus S'}(p, q) < (1 + \epsilon) \cdot [w(p) + w(q)] + w(c_i) + t_{\mathcal{B}} \cdot d_w(c_i, c_j) - w(c_j)$$

$$\leq (1 + \epsilon) \cdot [w(p) + w(q)] + w(c_i) + t_{\mathcal{B}} \cdot d_w(c_i, c_j)$$

[since weight associated with any point is non-negative]

$$= (1 + \epsilon) \cdot [w(p) + w(q)] + w(c_i) + t_{\mathcal{B}} \cdot [w(c_i) + |c_i c_j| + w(c_j)]$$

$$\leq (1 + \epsilon) \cdot [w(p) + w(q)] + w(p) + t_{\mathcal{B}} \cdot [w(p) + |c_i c_j| + w(q)]$$

[since the points are sorted in the non-decreasing order of their weights and the first

point added to any cluster is taken as its center]

$$\leq (1 + \epsilon) \cdot [w(p) + w(q)] + w(p) + t_{\mathcal{B}} \cdot [w(p) + |c_i p| + |pq| + |qc_j| + w(q)]$$

[by triangle inequality]

$$\leq (1 + \epsilon) \cdot [w(p) + w(q)] + w(p) + t_{\mathcal{B}} \cdot [w(p) + \epsilon \cdot w(p) + |pq| + \epsilon \cdot w(q) + w(q)]$$

[since a point x is added to cluster C_l only if $|xc_l| \leq \epsilon \cdot w(x)$]

$$\leq (2 + \epsilon) \cdot [w(p) + w(q)] + t_{\mathcal{B}} \cdot [(1 + \epsilon) \cdot [w(p) + w(q)] + |pq|]$$

$$\leq t_{\mathcal{B}} \cdot [(2 + \epsilon) \cdot [w(p) + w(q)] + |pq|] \quad \text{when } t_{\mathcal{B}} \geq (2 + \epsilon)$$

$$\leq t_{\mathcal{B}} \cdot (2 + \epsilon) \cdot [w(p) + w(q)] + t_{\mathcal{B}}[w(p) + |pq| + w(q)]$$

[since weight associated with any point is non-negative]

$$\leq t_{\mathcal{B}} \cdot (2 + \epsilon) \cdot d_w(p, q).$$

Case 8: Points p and q are in two distinct clusters, say $p \in C_i$ and $q \in C_j$; $p \neq c_i$, $q \neq c_j$; and both $c_i, c_j \in S'$.

Since p (resp. q) is connected to k nearest neighbor of c_i (resp. c_j), there exists $c_r, c_l \in C$ such that $c_r, c_l \notin S'$ and the edges (p, c_r) and (c_l, q) belong to $\mathcal{G} \setminus S'$. Then,

$$d_{\mathcal{G}\setminus S'}(p,q) = d_w(p,c_r) + d_{\mathcal{B}}(c_r,c_l) + d_w(c_l,q)$$
$$= w(p) + |pc_r| + w(c_r) + d_{\mathcal{B}}(c_r,c_l) + w(c_l) + |c_l q| + w(q)$$
$$\leq w(p) + |pc_i| + |c_i c_r| + w(c_r) + d_{\mathcal{B}}(c_r,c_l)$$
$$+ w(c_l) + |c_j c_l| + |c_j q| + w(q) \tag{3}$$
$$[\text{by triangle inequality}].$$

Since $\mathcal{B}$ is a $(k,t_{\mathcal{B}})$-VFTS, there are at least $k+1$ vertex disjoint $t_{\mathcal{B}}$-spanner paths between c_j and c_i in $\mathcal{B}$. Suppose c_r (resp. c_l) is the neighbor of c_i (resp. c_j) in $\mathcal{B}$ such that one of these $k+1$ paths from c_j to c_i passes through c_r (rsp. c_l). We have the following:

$$d_w(c_i,c_r) + d_{\mathcal{B}}(c_r,c_l) + d_w(c_l,c_j) < t_{\mathcal{B}} \cdot d_w(c_i,c_j)$$
$$\Rightarrow w(c_r) + |c_i c_r| + w(c_i) + d_{\mathcal{B}}(c_r,c_l) + w(c_j) + |c_j c_l| + w(c_l) < t_{\mathcal{B}} \cdot d_w(c_i,c_j)$$
$$\Rightarrow w(c_r) + |c_i c_r| + d_{\mathcal{B}}(c_r,c_l) + w(c_l) + |c_j c_l| < t_{\mathcal{B}} \cdot d_w(c_i,c_j) - w(c_i) - w(c_j). \tag{4}$$

Substituting (4) in (3), we get
$$d_{\mathcal{G}\setminus S'}(p,q) < w(p) + |pc_i| + t_{\mathcal{B}} \cdot d_w(c_i,c_j) - w(c_i) - w(c_j) + |c_j q| + w(q)$$
$$\leq w(p) + |pc_i| + t_{\mathcal{B}} \cdot d_w(c_i,c_j) + |c_j q| + w(q)$$
$$[\text{since the weight associated with each point is non-negative}]$$
$$= w(p) + |pc_i| + t_{\mathcal{B}} \cdot [w(c_i) + |c_i c_j| + w(c_j)] + |c_j q| + w(q)$$
$$\leq w(p) + |pc_i| + t_{\mathcal{B}} \cdot [w(c_i) + |c_i p| + |pq| + |qc_j| + w(c_j)] + |c_j q| + w(q)$$
$$[\text{by triangle inequality}]$$
$$\leq w(p) + \epsilon \cdot w(p) + t_{\mathcal{B}} \cdot [w(c_i) + \epsilon \cdot w(p) + |pq| + \epsilon \cdot w(q) + w(c_j)] + \epsilon \cdot w(q) + w(q)$$
$$[\text{since a point } x \text{ is added to cluster } C_l \text{ only if } |xc_l| \leq \epsilon \cdot w(x)]$$
$$\leq w(p) + \epsilon \cdot w(p) + t_{\mathcal{B}} \cdot [w(p) + \epsilon \cdot w(p) + |pq| + \epsilon \cdot w(q) + w(q)] + \epsilon \cdot w(q) + w(q)$$
$$[\text{since the points are sorted in the non-decreasing order of their weights and the first}$$
$$\text{point added to any cluster is taken as its center}]$$
$$\leq (1+\epsilon) \cdot [w(p) + w(q)] + t_{\mathcal{B}} \cdot [(1+\epsilon) \cdot [w(p) + w(q)] + |pq|]$$
$$\leq t_{\mathcal{B}} \cdot (2+\epsilon) \cdot [w(p) + w(q) + |pq|] \quad \text{when } t_{\mathcal{B}} \geq (1+\epsilon)$$
$$= t_{\mathcal{B}} \cdot (2+\epsilon) \cdot d_w(p,q).$$

Considering the analysis in all these cases proves that $\mathcal{G}$ is a k-VFTAWS with stretch t upper bounded by $t_{\mathcal{B}} \cdot (2+\epsilon)$. We had chosen $t_{\mathcal{B}}$ to be equal to $(2+\epsilon)$, so that it satisfies all the above cases. Since $t_{\mathcal{B}}$ is $(2+\epsilon)$, $t = (2+\epsilon)^2 \leq (4+5\epsilon)$. Hence, $\mathcal{G}$ is a $(k, 4+5\epsilon)$-VFTAWS for the metric space (S, d_w). $\qquad\square$

3 Vertex fault-tolerant additive weighted spanner for points in simple polygon

Given a set S of n points in a simple polygon P, for any two points $p, q \in S$, the shortest path between p and q in P is denoted by $\pi(p,q)$, and the length of that path is indicated by $d_\pi(p,q)$. For a $t \geq 1$, a geodesic t-spanner of S is a graph $\mathcal{G}(S, E')$ such that $d_\pi(p,q) \leq d_{\mathcal{G}}(p,q) \leq t \cdot d_\pi(p,q)$ for every two points $p, q \in S$. We detail an algorithm to compute a geodesic vertex fault-tolerant additive weighted spanner for the set S of n weighted points located in a simple polygon P.

The following definition for the distance function $d_{\pi,w}$ for the set S of points is considered in [4]: For any $p, q \in S$, $d_{\pi,w}(p,q)$ equals to 0 if $p = q$; otherwise, it is equal to $w(p) + d_\pi(p,q) + w(q)$. Further, $(S, d_{\pi,w})$ was shown as a metric space in [4].

We devise a divide-and-conquer based algorithm to compute a $(k, 4 + \epsilon)$-VFTAWS for the metric space $(S, d_{\pi,w})$. Following [1], we define few terms. Let S' be a set of points contained in a simple polygon P'. A vertical line segment that splits P' into two simple sub-polygons of P' such that each sub-polygon contains at most two-thirds of the points in S' is termed a *splitting segment* with respect to S' and P'. (In the following description, S' and P' are not mentioned with the splitting segment whenever they are clear from the context.) The *geodesic projection* p_l of a point p onto a splitting segment l is a point on l that has the minimum geodesic Euclidean distance from p among all the points of l. By extending [1], we give an algorithm to compute a $(k, 4 + \epsilon)$-VFTAWS $\mathcal{G}$ for the metric space $(S, d_{\pi,w})$.

Our algorithm partitions P containing points in S into two simple sub-polygons P' and P'' with a splitting segment l. For every point $p \in S$, we compute its geodesic projection p_l onto l and assign $w(p) + d_\pi(p, p_l)$ as the weight of p_l. Let S_l be the set comprising of all the geodesic projections of S onto l. Also, let $d_{l,w}$ be the additive weighted metric associated with points in S_l. We use the algorithm from Section 2 to compute a $(k, 4 + \epsilon)$-VFTAWS $\mathcal{G}_l$ for the metric space $(S_l, d_{l,w})$. For every edge (r, s) in $\mathcal{G}_l$, we add an edge between p and q to $\mathcal{G}$ with weight $d_\pi(p, q)$, wherein r (resp. s) is the geodesic projection of p (resp. q) onto l. Let S' (resp. S'') be the set of points contained in the sub-polygon P' (resp. P'') of P. We recursively process P' (resp. P'') with points in S' (resp. S'') unless $|S'|$ (resp. $|S''|$) is less than or equal to one.

Algorithm 2: k-ADDITIVEPOLYGONCLUSTFTS(S, P, k, ϵ)

Input : The simple polygon P, the set S on n additive weighted points, integer $k \geq 1$ and a real number $0 < \epsilon \leq 1$

Output: $(k, (12 + \epsilon))$-VFTS $\mathcal{G}$

1: **while** $card(P \cap S) \geq 1$ **do**
2:     compute a splitting segment l for P using the algorithm given in [20]
3:     initialize the set S_l of the projections to ϕ
4:     **for** every $p \in S$ **do**
5:       find the projection p_l of p on l
6:       assign a weight $w(p) + d_\pi(p, p_l)$ to p_l
7:       $S_l := S_l \cup \{p_l\}$
8:     **end for**
9:     compute a $(k, (4 + \epsilon))$-vertex fault-tolerant spanner $\mathcal{G}_l$ for the set S_l using Algorithm 1
10:     for every edge (r, s) in $\mathcal{G}_l$, add the edge (p, q) to $\mathcal{G}$, where
      r(resp. s) is the projection of p(resp. q) on l
11:     k-ADDITIVEPOLYGONCLUSTFTS$(S \cap P'_l, P'_l, k, \epsilon)$
      $//P'_l$ denotes the sub-polygon to the left of l
12:     k-ADDITIVEPOLYGONFTS$(S \cap P''_l, P''_l, k, \epsilon)$
      $//P''_l$ denotes the sub-polygon to the right of l
13: **end while**

We prove that the graph $\mathcal{G}$ is a $(k, (12 + 15\epsilon))$-VFTAWS for the metric space $(S, d_{\pi,w})$. (Later, with further refinements to this graph, we improve the stretch factor to $(4 + 14\epsilon)$.) We show that by removing any subset S' with $|S'| \leq k$ from $\mathcal{G}$, for any two points p and q in $S \setminus S'$, there exists a path between p and q in $\mathcal{G} \setminus S'$ such that the $d_{\mathcal{G}}(p, q)$ is at most $(12 + 15\epsilon)d_{\pi,w}(p, q)$. First, we note that there exists a splitting segment l at some iteration of the algorithm so that p and q are on different sides of l. Let r be a point belonging to $l \cap \pi(p, q)$. Let S'_l be the set comprising of geodesic projections of points in S' on l. Since $\mathcal{G}_l$ is a $(k, (4 + 5\epsilon))$-VFTAWS for the metric space $(S_l, d_{l,w})$, there exists a path Q between p_l and q_l in $\mathcal{G}_l \setminus S'_l$ whose length is upper bounded by $(4 + 5\epsilon) \cdot d_{l,w}(p_l, q_l)$. Let Q' be a path between p and q in $\mathcal{G} \setminus S'$ which is obtained by replacing each vertex v_l of Q by v in S such that the point v_l is the geodesic projection of v on l. In the following, we show that the length of Q', which is $d_{\mathcal{G} \setminus S'}(p, q)$, is upper bounded by $(12 + 15\epsilon) \cdot d_{\pi,w}(p, q)$.

For every $x, y \in S$,

$$d_{\pi,w}(x, y) = w(x) + d_\pi(x, y) + w(y)$$
$$\leq w(x) + d_\pi(x, x_l) + d_\pi(x_l, y_l) + d_\pi(y_l, y) + w(y)$$
[by triangle inequality]
$$= w(x_l) + d_\pi(x_l, y_l) + w(y_l)$$
[since the weight associated with projection z_l of every point z is $w(z) + d_\pi(z, z_l)$]
$$= d_{l,w}(x_l, y_l). \tag{5}$$

This implies,

$$d_{\mathcal{G} \setminus S'}(p, q) \leq \sum_{x_l, y_l \in Q} d_{\pi,w}(x, y)$$
$$\leq \sum_{x_l, y_l \in Q} d_{l,w}(x_l, y_l)$$
[from (5)]
$$\leq (4 + 5\epsilon) \cdot d_{l,w}(p_l, q_l) \tag{6}$$
[since $\mathcal{G}_l$ is a $(k, (4 + 5\epsilon))$-vertex fault-tolerant geodesic spanner]
$$= (4 + 5\epsilon) \cdot [w(p_l) + d_l(p_l, q_l) + w(q_l)]$$
$$= (4 + 5\epsilon) \cdot [w(p_l) + d_\pi(p_l, q_l) + w(q_l)]$$
[since P contains l, shortest path between p_l and q_l
is same as the geodesic shortest path between p_l and q_l]
$$= (4 + 5\epsilon) \cdot [w(p) + d_\pi(p, p_l) + d_\pi(p_l, q_l) + d_\pi(q_l, q) + w(q)] \tag{7}$$
[since the weight associated with projection z_l of every point z is $w(z) + d_\pi(z, z_l)$].

Since r is a point belonging to both l as well as to $\pi(p, q)$,
$$d_\pi(p, p_l) \leq d_\pi(p, r) \text{ and } d_\pi(q, q_l) \leq d_\pi(q, r). \tag{8}$$

Substituting (18) into (17),

$$d_{\mathcal{G} \setminus S'}(p, q) \leq (4 + 5\epsilon) \cdot [w(p) + d_\pi(p, r) + d_\pi(p_l, q_l) + d_\pi(r, q) + w(q)]$$
$$\leq (4 + 5\epsilon) \cdot [w(p) + d_\pi(p, r) + w(r) + d_\pi(p_l, q_l) + w(r) + d_\pi(r, q) + w(q)]$$
[since the weight associated with every point is non-negative]
$$= (4 + 5\epsilon) \cdot [d_{\pi,w}(p, r) + d_\pi(p_l, q_l) + d_{\pi,w}(r, q)]$$
$$= (4 + 5\epsilon) \cdot [d_{\pi,w}(p, q) + d_\pi(p_l, q_l)] \tag{9}$$
[since $\pi(p, q)$ intersects l at r, by optimal substructure property of shortest
paths, $\pi(p, q) = \pi(p, r) + \pi(r, q)$].

10

Consider

$$d_\pi(p_l, q_l) \le d_\pi(p_l, p) + d_\pi(p, q) + d_\pi(q, q_l)$$

[since π follows triangle inequality]

$$\le d_\pi(r, p) + d_\pi(p, q) + d_\pi(q, r)$$

[using (18)]

$$\le w(r) + d_\pi(r, p) + w(p) + w(p) + d_\pi(p, q) + w(q) + w(q) + d_\pi(q, r) + w(r)$$

[since weight associated with every point is non-negative]

$$= d_{\pi,w}(p, r) + d_{\pi,w}(p, q) + d_{\pi,w}(r, q)$$

$$= d_{\pi,w}(p, q) + d_{\pi,w}(p, q)$$

[since $\pi(p, q)$ intersects l at r, by optimal substructure property of shortest

paths, $\pi(p, q) = \pi(p, r) + \pi(r, q)$]

$$= 2d_{\pi,w}(p, q). \tag{10}$$

Substituting (20) into (19),

$$d_{G\setminus S'}(p, q) \le 3(4 + 5\epsilon) \cdot d_{\pi,w}(p, q).$$

Hence, the graph $\mathcal{G}$ computed as described above is a $(k, 12+\epsilon)$-VFTAWS for the metric space $(S, d_{\pi,w})$. We further improve the stretch factor of $\mathcal{G}$ by applying the refinement given in [6] to the above-described algorithm. In doing this, for each point $p \in S$, we compute the geodesic projection p_γ of p on the splitting line γ and we construct a set $S(p, \gamma)$ as defined herewith. Let $\gamma(p) \subseteq \gamma$ be $\{x \in \gamma : d_{\gamma,w}(p_\gamma, x) \le (1+2\epsilon) \cdot d_\pi(p, p_\gamma)\}$. Here, for any $p, q \in S$, $d_{\gamma,w}(p, q)$ is equal to 0 if $p = q$; otherwise, equals to $w(p) + d_\gamma(p, q) + w(q)$. We divide $\gamma(p)$ into c pieces with $c \in O(1/\epsilon^2)$: each piece is denoted by $\gamma_j(p)$ for $1 \le j \le c$, and the piece length is at most $\epsilon \cdot d_\pi(p, p_\gamma)$. For every piece j, we compute the point $p_\gamma^{(j)}$ nearest to p in $\gamma_j(p)$. The set $S(p, \gamma)$ is defined as $\{p_\gamma^{(j)} : p_\gamma^{(j)} \in \gamma_j(p)$ and $1 \le j \le c\}$. For every $r \in S(p, \gamma)$, the non-negative weight $w(r)$ of r is set to $w(p) + d_\pi(p, r)$. Let S_γ be $\cup_{p \in S} S(p, \gamma)$.

We replace the set S_l in computing $\mathcal{G}$ with the set S_γ and compute a $(k, (4+5\epsilon))$-VFTAWS $\mathcal{G}_l$ using the algorithm from Section 2 for the set S_γ instead. Further, for every edge (r, s) in $\mathcal{G}_l$, we add the edge (p, q) to $\mathcal{G}$ with weight $d_\pi(p, q)$ whenever $r \in S(p, l)$ and $s \in S(q, l)$. The rest of the algorithm remains the same.

In the following, we restate a lemma from [6], which is useful for our analysis.

Lemma 1. *Let P be a simple polygon. Consider two points $x, y \in P$. Let r be the point at which shortest path $\pi(x, y)$ between x and y intersects a splitting segment γ. If $r \notin \gamma(x)$, point x'_γ (resp. y'_γ) is set as x_γ (resp. y_γ). Otherwise x'_γ (resp. y'_γ) is set as the point from $S(x, \gamma)$ (resp. $S(y, \gamma)$) which is nearest to x (resp. y). Then $d_\pi(x, x'_\gamma) + d_\gamma(x'_\gamma, r)$ (resp. $d_\gamma(r, y'_\gamma) + d_\pi(y'_\gamma, y)$) is less than or equal to $(1+\epsilon) \cdot d_\pi(x, r)$ (resp. $(1+\epsilon) \cdot d_\pi(r, y)$).*

Theorem 2. *Let S be a set of n weighted points in simple polygon P with non-negative weights associated to points with weight function w. For any fixed constant $\epsilon > 0$, there exists a $(k, (4+\epsilon))$-vertex fault-tolerant additive weighted geodesic spanner with $O(\frac{kn}{\epsilon^2} \lg n)$ edges for the metric space $(S, d_{\pi,w})$.*

Proof: In constructing a $(k, (4+\epsilon))$-VFTAWS $\mathcal{G}_l$ for the set S_γ of $\frac{n}{\epsilon^2}$ points, we add $O(\frac{kn}{\epsilon^2})$ edges to $\mathcal{G}$ in one iteration. Let $S(n)$ be the size of $\mathcal{G}$ when there are n points. Then $S(n) = S(n_1) + S(n_2) + \frac{kn}{\epsilon^2}$ where n_1, n_2 are the number of points in each of the partitions formed by the splitting segment. Since $n_1, n_2 \ge n/3$, $S(n)$ is $O(\frac{kn}{\epsilon^2} \lg n)$.

For proving that $\mathcal{G}$ is a $(k, (4+\epsilon))$-VFTAWS for the metric space $(S, d_{\pi,w})$, we show that for any set $S' \subset S$ with $|S'| \le k$ and for any two points $p, q \in S \setminus S'$ there exists a $(4+\epsilon)$-spanner path between p and q in $\mathcal{G} \setminus S'$. First, we note that there exists a splitting segment l at some iteration of the algorithm so that p and q are on different sides of l. Let r be a point belonging to $l \cap \pi(p, q)$. Let S'_l be the set comprising of geodesic projections of points in S' on l. Since $\mathcal{G}_l$ is a $(k, (4+5\epsilon))$-VFTAWS for the metric space $(S_l, d_{l,w})$,

there exists a path Q between p_l and q_l in $\mathcal{G}_l \setminus S'_l$ whose length is upper bounded by $(4+5\epsilon) \cdot d_{l,w}(p_l, q_l)$. Let Q' be a path between p and q in $\mathcal{G} \setminus S'$ which is obtained by replacing each vertex v_l of Q by v in S such that the point v_l is the geodesic projection of v on l. In the following, we show that the length of Q', which is $d_{\mathcal{G}\setminus S'}(p, q)$, is upper bounded by $(4 + 14\epsilon) \cdot d_{\pi,w}(p, q)$.

Following Lemma 1, if $r \notin l(p)$, point p'_l (resp. q'_l) is set as p_l (resp. q_l). Otherwise p'_l (resp. q'_l) is set as the point from $S(p, l)$ (resp. $S(q, l)$) which is nearest to p (resp. q).

$$
\begin{aligned}
d_{l,w}(p'_l, q'_l) &= w(p'_l) + d_l(p'_l, q'_l) + w(q'_l) \\
&\leq w(p'_l) + d_l(p'_l, r) + d_l(r, q'_l) + w(q'_l) \\
&\quad \text{[by triangle inequality]} \\
&\leq w(p'_l) + d_l(p'_l, r) + w(r) + w(r) + d_l(r, q'_l) + w(q'_l) \\
&\quad \text{[since the weight associated with each point is non-negative]} \\
&= w(p) + d_\pi(p, p'_l) + d_l(p'_l, r) + w(r) \\
&\quad + w(r) + d_l(r, q'_l) + d_\pi(q'_l, q) + w(q) \\
&\quad \text{[due to weight assigned to geodesic projections]}.
\end{aligned}
\tag{11}
$$

Applying Lemma 1 with p'_l and q'_l,
$$
d_\pi(p, p'_l) + d_l(p'_l, r) \leq (1 + \epsilon) \cdot d_\pi(p, r), \text{ and}
\tag{12}
$$
$$
d_l(r, q'_l) + d_\pi(q'_l, q) \leq (1 + \epsilon) \cdot d_\pi(r, y).
\tag{13}
$$

Substituting (22) and (23) in (11),
$$
\begin{aligned}
d_{l,w}(p'_l, q'_l) &\leq w(p) + (1 + \epsilon) \cdot d_\pi(p, r) + w(r) + w(r) + (1 + \epsilon) \cdot d_\pi(r, q) + w(q) \\
&\leq (1 + \epsilon) \cdot [d_{\pi,w}(p, r) + d_{\pi,w}(r, q)] \\
&= (1 + \epsilon) \cdot d_{\pi,w}(p, q) \\
&\quad \text{[since } r \in l \cap \pi(p, q), \text{ by the optimal substructure property of shortest} \\
&\quad \text{paths, } \pi(p, q) = \pi(p, r) + \pi(r, q)].
\end{aligned}
\tag{14}
$$

Replacing p_l (resp. q_l) by p_l' (resp. q_l') in inequality (6),
$$
\begin{aligned}
d_{\mathcal{G}\setminus S'}(p, q) &\leq (4 + 5\epsilon) \cdot d_{l,w}(p_l', q_l') \\
&\leq (4 + 5\epsilon)(1 + \epsilon) \cdot d_{\pi,w}(p, q) \\
&\quad \text{[from (24)]} \\
&\leq (4 + 14\epsilon) \cdot d_{\pi,w}(p, q).
\end{aligned}
$$

Thus, $\mathcal{G}$ is a $(k, (4 + \epsilon))$-vertex fault-tolerant additive weighted geodesic spanner for the set S of points located in the simple polygon P. $\qquad\square$

4 Vertex fault-tolerant additive weighted spanner for points in a polygonal domain

We devise an algorithm to compute a geodesic $(k, (4+\epsilon))$-vertex fault-tolerant spanner for a set S of n points lying in the free space $\mathcal{D}$ of the given polygonal domain $\mathcal{P}$ while each input point is associated with a non-negative weight. The polygonal domain $\mathcal{P}$ consists of a simple polygon and h simple polygonal holes located interior to to that polygon. Our algorithm depends on the algorithm given in [1] to compute a $(5+\epsilon)$-spanner

for a set of unweighted points lying in $\mathcal{D}$. We decompose the free space $\mathcal{D}$ into simple polygons using $O(h)$ splitting segments such that no splitting segment crosses any of the holes of $\mathcal{D}$ and each of the resultant simple polygons has at most three splitting segments bounding it. As part of this decomposition, two vertical line segments are drawn, one upwards and the other downwards, respectively from the leftmost and rightmost extreme (along the x-axis) vertices of each hole. If any of the resulting simple polygons has more than three splitting segments on its boundary, then that simple polygon is further decomposed. To achieve efficiency, a splitting segment is chosen such that it has around half of its bounding splitting segments on either of its sides. This algorithm results in partitioning $\mathcal{D}$ into $O(h)$ simple polygons. Further, a graph $\mathcal{G}_d$ is constructed where each vertex of $\mathcal{G}_d$ corresponds to a simple polygon of this decomposition. Each vertex v of $\mathcal{G}_d$ is associated with a weight equal to the number of points that lie inside the simple polygon corresponding to v. Two vertices are connected by an edge in $\mathcal{G}_d$ whenever their corresponding simple polygons are adjacent to each other in the decomposition. We note that $\mathcal{G}_d$ is a planar graph. Hence, we use the following theorem from [8] to compute a $O(\sqrt{h})$-separator R of $\mathcal{G}_d$.

Theorem 3. *Suppose $G = (V, E)$ is a planar vertex-weighted graph with $|V| = m$. Then, an $O(\sqrt{m})$-separator for G can be computed in $O(m)$ time. That is, V can be partitioned into sets P, Q and R such that $|R| = O(\sqrt{m})$, there is no edge between P and Q, and $w(P), w(Q) \leq \frac{2}{3}w(V)$. Here, $w(X)$ is the sum of weights of all vertices in X.*

We compute a $O(\sqrt{h})$-separator R for the graph $\mathcal{G}_d$ using Theorem 3. Let P, Q, and R be the sets into which the vertices of $\mathcal{G}_d$ is partitioned. For each vertex $r \in R$, we collect the bounding splitting segments of the simple polygon corresponding to r into H i.e., $O(\sqrt{h})$ splitting segments are collected into a set H. For each splitting segment l in H, we proceed as follows. For each point p that lies in the given simple polygon, we find the projection p_l of p on l; we assign the weight $w(p) + d_\pi(p, p_l)$ to point p_l and include p_l into the set S_l corresponding to points projected on to line l. We compute the $(k, 4+\epsilon)$-VFTAWS $\mathcal{G}_l$ for the set S_l of points using the algorithm given in Section 2 for additive weighted points located in $\mathbb{R}^d$. For every edge (r, s) in $\mathcal{G}_l$, we introduce an edge (p, q) in G, where r (resp. s) is the projection of p (resp. q) on l. Recursively, we compute vertex-fault tolerant additive weighted spanner for points lying in simple polygon corresponding to P (resp. Q). The recursion is continued till P (resp. Q) contains exactly one vertex. We first prove that this algorithm computes a $(k, (12 + \epsilon))$-vertex fault-tolerant spanner. Further, we modify this algorithm to compute a $(k, (4 + \epsilon))$-vertex fault-tolerant spanner.

Lemma 2. *The spanner G is a geodesic $(k, (12 + 15\epsilon))$-vertex fault-tolerant additive weighted spanner for points in $\mathcal{D}$.*

Proof: Using induction on the number of points, we show that there exists a $(12+15\epsilon)$-spanner path between p and q in $\mathcal{G} \setminus S'$. The induction hypothesis assumes that for the number of points $k' < |S|$, there exists a $(12+15\epsilon)$-spanner path between any two points belonging to G S. Consider a set $S' \subset S$ such that $|S'| \leq k$ and two arbitrary points p and q from the set $S \setminus S'$. Here, as described above, P, Q, and R correspond to vertices of a planar graph $\mathcal{G}_d$. The union of simple polygons that correspond to vertices of P (resp. Q, R) is denoted with $poly(P)$ (resp. $poly(Q), poly(R)$). Also, the set H is as described in the algorithm. Based on the location of p and q, the following cases arise: (i) both p and q are lying in $P' \in \{poly(P), poly(Q), \text{ and } poly(R)\}$ and the geodesic Euclidean shortest path between p and q does not intersect any splitting segment from the set H, and (ii) p is lying in $P' \in \{poly(P), poly(Q), poly(R)\}$ and q is lying in $P'' \in \{poly(P), poly(Q), poly(R)\}$ where $P' \neq P''$. In case (i), if P' is a simple polygon, then we can apply algorithm for simple polygons from Section 3 and obtain a $(4 + 14\epsilon)$-path between p and q. Otherwise, from the induction hypothesis, there exists a $(12 + 15\epsilon)$-path between p and q. In case (ii), a shortest path from p and q intersects at least one of the $O(\sqrt{h})$ splitting segments in H, say l. Let $\pi(p, q)$ be a shortest path between p and q that intersects l at some point. Let r be this point of intersection. Since $\mathcal{G}_l$ is a $(k, (4 + 5\epsilon))$-VFTAWS, there exists a path P' between p_l and q_l in $\mathcal{G}_l$ with length at most $(4 + 5\epsilon)d_{l,w}(p_l, q_l)$. By replacing each vertex x_l of P' by $x \in S$ such that x_l is the projection of x on l, gives a path between p and q in $\mathcal{G} \setminus S'$. Thus, the length of the path $d_{\mathcal{G} \setminus S'}(p, q)$ is less than or equal to the length of the path P' in $\mathcal{G}_l$. For every $x, y \in S$,

Algorithm 3: k-ADDITIVEPOLYDOMCULSTFTS$(S, \mathcal{D}, k, \epsilon)$

Input : The polygonal domain $\mathcal{D}$, the set S on n points, non-negative weights associated to points in S, integer $k \geq 1$ and a real number $0 < \epsilon \leq 1$

Output: $(k, (12 + \epsilon))$-VFTS $\mathcal{G}$

1: decompose $\mathcal{D}$ into $O(h)$ simple polygons such that each of these simple polygons have at most three splitting lines on its boundary

2: construct a planar graph $\mathcal{G}_d$ (description of $\mathcal{G}_d$ is in text)

3: initialize the set X to contain all the vertices of $\mathcal{G}_d$

4: **while** $card(X) \geq 1$ **do**

5:   **if** $card(X) = 1$ **then**

6:     k-ADDITIVEPOLYGONCULSTFTS$(S \cap simplepoly(X), simplepoly(X), k, \epsilon)$
    // $simplepoly(X)$ denotes the simple polygon corresponding to the only vertex in X

7:   **else**

8:     compute a $O(\sqrt{h})$-separator R for the graph $\mathcal{G}_d$ using Theorem 3;
    let P, Q and R be the sets into which the vertices of $\mathcal{G}_d$ is partitioned

9:     for every $r \in R$, add the bounding splitting segments of the simple polygon corresponding to r to a set H

10:     **for** every $l \in H$ **do**

11:       **for** every $p \in \mathcal{D} \cap S$ **do**

12:         find the projection p_l of p on l

13:         assign weight equal to $w(p) + d_\pi(p, p_l)$ to p_l

14:         $S_l := S_l \cup \{p_l\}$

15:       **end for**

16:       compute the $(k, (4 + \epsilon))$-vertex fault-tolerant spanner for the set S_l using Algorithm 1

17:       for every edge (r, s) in $\mathcal{G}_l$, add the edge (p, q) to $\mathcal{G}$, where
      r (resp. s) is the projection of p(resp. q) on l

18:     **end for**

19:     k-ADDITIVEPOLYDOMCULSTFTS$(S \cap unionpoly(P), \mathcal{D} \cap unionpoly(P), k, \epsilon)$
    // $unionpoly(P)$ denotes the set of simple polygons corresponding to each vertex of P

20:     k-ADDITIVEPOLYDOMCULSTFTS$(S \cap unionpoly(Q), \mathcal{D} \cap unionpoly(Q), k, \epsilon)$
    // $unionpoly(Q)$ denotes the set of simple polygons corresponding to each vertex of Q

21:   **end if**

22: **end while**

$$d_{\pi,w}(x,y) = w(x) + d_\pi(x,y) + w(y)$$
$$\leq w(x) + d_\pi(x,x_l) + d_\pi(x_l,y_l) + d_\pi(y_l,y) + w(y)$$
[by the triangle inequality]
$$= w(x_l) + d_\pi(x_l,y_l) + w(y_l)$$
[since the weight associated with projection z_l of every point z is
$w(z) + d_\pi(z,z_l)$]
$$= d_{l,w}(x_l,y_l) \tag{15}$$

This implies,
$$d_{\mathcal{G}\setminus S'}(p,q) = \sum_{x_l,y_l \in P} d_{\pi,w}(x,y)$$
$$\leq \sum_{x_l,y_l \in P} d_{l,w}(x_l,y_l)$$
[from (15)]
$$\leq (4+5\epsilon).d_{l,w}(p_l,q_l) \tag{16}$$
[since $\mathcal{G}_l$ is a geodesic $(k,(4+5\epsilon))$-VFTAWS]
$$= (4+5\epsilon) \cdot [w(p_l) + d_l(p_l,q_l) + w(q_l)]$$
$$= (4+5\epsilon) \cdot [w(p_l) + d_\pi(p_l,q_l) + w(q_l)]$$
[since P contains l, shortest path between p_l and q_l along l
is same as the geodesic shortest path between p_l and q_l]
$$= (4+5\epsilon) \cdot [w(p) + d_\pi(p,p_l) + d_\pi(p_l,q_l) + d_\pi(q_l,q) + w(q)] \tag{17}$$
[since the weight associated with projection z_l of every point z is
$w(z) + d_\pi(z,z_l)$].

Since r is a point belonging to both l as well as to $\pi(p,q)$,
$$d_\pi(p,p_l) \leq d_\pi(p,r) \text{ and } d_\pi(q,q_l) \leq d_\pi(q,r). \tag{18}$$

Substituting (18) into (17),
$$d_{\mathcal{G}\setminus S'}(p,q) \leq (4+5\epsilon) \cdot [w(p) + d_\pi(p,r) + d_\pi(p_l,q_l) + d_\pi(r,q) + w(q)]$$
$$\leq (4+5\epsilon) \cdot [w(p) + d_\pi(p,r) + w(r) + d_\pi(p_l,q_l) + w(r) + d_\pi(r,q) + w(q)]$$
[since the weight associated with every point is non-negative]
$$= (4+5\epsilon) \cdot [d_{\pi,w}(p,r) + d_\pi(p_l,q_l) + d_{\pi,w}(r,q)]$$
$$= (4+5\epsilon) \cdot [d_{\pi,w}(p,q) + d_\pi(p_l,q_l)] \tag{19}$$
[since $\pi(p,q)$ intersects l at r, by optimal substructure property of shortest
paths, $\pi(p,q) = \pi(p,r) + \pi(r,q)$].

Consider

$$d_\pi(p_l, q_l) \le d_\pi(p_l, p) + d_\pi(p, q) + d_\pi(q, q_l)$$

[since π follows triangle inequality]

$$\le d_\pi(r, p) + d_\pi(p, q) + d_\pi(q, r)$$

[using (18)]

$$\le w(r) + d_\pi(r, p) + w(p) + w(p) + d_\pi(p, q) + w(q) + w(q) + d_\pi(q, r) + w(r)$$

[since weight associated with every point is non-negative]

$$= d_{\pi,w}(p, r) + d_{\pi,w}(p, q) + d_{\pi,w}(r, q)$$

$$= d_{\pi,w}(p, q) + d_{\pi,w}(p, q)$$

[since $\pi(p, q)$ intersects l at r, by optimal substructure property of shortest

paths, $\pi(p, q) = \pi(p, r) + \pi(r, q)$]

$$= 2d_{\pi,w}(p, q). \tag{20}$$

Substituting (20) into (19), $d_{G \setminus S'}(p, q) \le 3(4 + 5\epsilon) \cdot d_{\pi,w}(p, q).$ $\qquad \square$

We further improve the stretch factor of $\mathcal{G}$ by applying the refinement given in [6] to the above-described algorithm. In doing this, for each point $p \in S$, we compute the geodesic projection p_γ of p on the splitting line γ and we construct a set $S(p, \gamma)$ as defined herewith. Let $\gamma(p) \subseteq \gamma$ be $\{x \in \gamma : d_{\gamma,w}(p_\gamma, x) \le (1+2\epsilon) \cdot d_\pi(p, p_\gamma)\}$. Here, for any $p, q \in S$, $d_{\gamma,w}(p, q)$ is equal to 0 if $p = q$; otherwise, it is equal to $w(p) + d_\gamma(p, q) + w(q)$. We divide $\gamma(p)$ into c pieces with $c \in O(1/\epsilon^2)$: each piece is denoted by $\gamma_j(p)$ for $1 \le j \le c$. For every piece j, we compute the point $p_\gamma^{(j)}$ nearest to p in $\gamma_j(p)$. The set $S(p, \gamma)$ is defined as $\{p_\gamma^{(j)} : p_\gamma^{(j)} \in \gamma_j(p)$ and $1 \le j \le c\}$. For every $r \in S(p, \gamma)$, the non-negative weight $w(r)$ of r is set to $w(p) + d_\pi(p, r)$. Let S_γ be $\cup_{p \in S} S(p, \gamma)$. We replace the set S_l in computing G with the set S_γ and compute a $(k, (4 + 5\epsilon))$-VFTAWS $\mathcal{G}_l$ for the set S_l using the algorithm for points in $\mathbb{R}^d$ given in Section 2. Further, for every edge (r, s) in $\mathcal{G}_l$, we add the edge (p, q) to G such that $r \in S(p, l)$ and $s \in S(q, l)$. The rest of the algorithm remains the same.

Theorem 4. *Let S be a set of n points in a polygonal domain $\mathcal{D}$ with non-negative weights associated to points via weight function w. For any fixed constant $\epsilon > 0$, there exists a $(k, (4 + \epsilon))$-vertex fault-tolerant additive weighted geodesic spanner with $O(\frac{kn\sqrt{h}}{\epsilon^2} \lg n)$ edges for the metric space $(S, d_{\pi,w})$.*

Proof: Let $S(n)$ be the size of $\mathcal{G}$. Our algorithm adds $O(\frac{kn\sqrt{h}}{\epsilon^2})$ edges at each recursive level except for the last level. At every leaf node l of the recurrence tree, we add $O(\frac{kn_x}{\epsilon^2} \lg n_x)$ edges, where n_x is the number of points inside the simple polygon corresponding to l. Hence, the number of edges of G is $O(\frac{kn\sqrt{h}}{\epsilon^2} \lg n)$.

Next, we prove the stretch factor of the spanner. Consider any set $S' \subset S$ such that $|S'| \le k$ and two arbitrary points p and q from the set $S \setminus S'$. We show that there exists a $(4 + 14\epsilon)$-spanner path between p and q in $\mathcal{G} \setminus S'$. If $r \notin l(p)$, then we set p'_l (resp. q'_l) equal to p_l (resp. q_l). Otherwise, p'_l (resp. q'_l) is set as the point from $S(p, l)$ (resp. $S(q, l)$) that is nearest to p (resp. q). (The r is defined before the theorem statement.)

$$d_{l,w}(p'_l, q'_l) = w(p'_l) + d_l(p'_l, q'_l) + w(q'_l)$$

$$\le w(p'_l) + d_l(p'_l, r) + d_l(r, q'_l) + w(q'_l)$$

[by the triangle inequality]

$$\le w(p'_l) + d_l(p'_l, r) + w(r) + w(r) + d_l(r, q'_l) + w(q'_l)$$

[since the weight associated with each point is non-negative]

$$= w(p) + d_\pi(p, p'_l) + d_l(p'_l, r) + w(r) + w(r) + d_l(r, q'_l) +$$

$$d_\pi(q'_l, q) + w(q) \tag{21}$$

[due to the assignment of the weight to the projection of any point]

From the triangle inequality, we know the following:

$$d_\pi(p, p'_l) + d_l(p'_l, r) \leq d_\pi(p, r), \text{ and} \tag{22}$$

$$d_l(r, q'_l) + d_\pi(q'_l, q) \leq d_\pi(r, q). \tag{23}$$

Substituting (22) and (23) in (21),

$$
\begin{aligned}
d_{l,w}(p'_l, q'_l) &\leq w(p) + d_\pi(p, r) + w(r) + w(r) + d_\pi(r, q) + w(q) \\
&= d_{\pi,w}(p, r) + d_{\pi,w}(r, q) \\
&= d_{\pi,w}(p, q) \tag{24}
\end{aligned}
$$

[since $r \in l \cap \pi(p, q)$, by the optimal substructure property of shortest

paths, $\pi(p, q) = \pi(p, r) + \pi(r, q)$].

Replacing p_l (resp. q_l) by p'_l (resp. q'_l) in inequality (16),

$$
\begin{aligned}
d_{\mathcal{G} \setminus S'}(p, q) &\leq (4 + 5\epsilon).d_{l,w}(p'_l, q'_l) \\
&\leq (4 + 5\epsilon)d_{\pi,w}(p, q) \quad \text{[from (24)]}.
\end{aligned}
$$

Thus, G is a geodesic $(k, (4 + \epsilon))$-VFTAWS for S. $\qquad\square$

5   Vertex fault-tolerant additive weighted spanner for points on a terrain

In this section, we present an algorithm to compute a geodesic $(k, (4+\epsilon))$-VFTAWS with $O(\frac{kn \lg n}{\epsilon^2})$ edges for any given set S of n non-negative weighted points lying on a polyhedral terrain $\mathcal{T}$. We denote the boundary of $\mathcal{T}$ with $\partial\mathcal{T}$. The following distance function $d_{\mathcal{T},w} : S \times S \to \mathbb{R} \cup \{0\}$ is used to compute the geodesic distance on $\mathcal{T}$ between any two points $p, q \in S$: $d_{\mathcal{T},w}(p, q) = w(p) + d_\mathcal{T}(p, q) + w(q)$. Here, $w(p)$ (resp. $w(q)$) is the non-negative weight of $p \in S$ (resp. $q \in S$). We denote a geodesic Euclidean shortest path between any two points a and b on $\mathcal{T}$ with $\pi(a, b)$. For any two points $x, y \in \partial\mathcal{T}$, we denote the closed region lying to the right (resp. left) of $\pi(x, y)$ when going from x to y, including the points lying on the shortest path $\pi(x, y)$ with $\pi^+(x, y)$ (resp. $\pi^-(x, y)$). The *projection* p_π of a point p on the shortest path π between two points lying on the polyhedral terrain $\mathcal{T}$ is defined as a point on π that is at the minimum geodesic distance from p among all the points located on π. For three points $u, v, w \in \mathcal{T}$, the closed region bounded by shortest paths $\pi(u, v)$, $\pi(v, w)$, and $\pi(w, u)$ is termed *sp-triangle*, denoted with $\Delta(u, v, w)$. If the points $u, v, w \in \mathcal{T}$ are clear from the context, we denote the sp-triangle with Δ. In the following, we restate a Theorem from [6], which is useful for our analysis.

Theorem 5. *For any set P of n points on a polyhedral terrain $\mathcal{T}$, there exists a balanced sp-separator: a shortest path $\pi(u, v)$ connecting two points $u, v \in \partial\mathcal{T}$ such that $\frac{2n}{9} \leq |\pi^+(u, v) \cap P| \leq \frac{2n}{3}$, or a sp-triangle Δ such that $\frac{2n}{9} \leq |\Delta \cap P| \leq \frac{2n}{3}$.*

Thus, an sp-separator is either bounded by a shortest path (in the former case) or by three shortest paths (in the latter case). Let γ be a shortest path that belongs to an sp-separator. First, a balanced sp-separator as given in Theorem 5 is computed. The sets S_{in} and S_{out} comprising of points are defined as follows: if the sp-separator is a shortest path then define S_{in} to be $\gamma^+(u, v) \cap S$; otherwise, S_{in} is $\Delta \cap S$; points in S that do not belong to S_{in} are in S_{out}. For each $p \in S$, we compute the projection p_γ of p on every shortest path γ of sp-separator, and associate a weight $d_\mathcal{T}(p, p_\gamma)$ with p_γ. Let S_γ be a set defined as $\cup_{p \in S} p_\gamma$. Our algorithm computes a $(2 + \epsilon)$-spanner $\mathcal{G}_\gamma$ for the weighted points in S_γ. Further, for each edge (p_γ, q_γ) in $\mathcal{G}_\gamma$, an edge (p, q) is added to $\mathcal{G}$, where p_γ (resp. q_γ) is the projection of p (resp. q) on γ. The spanners for the sets S_{in} and S_{out} are computed recursively, and the edges from these spanners are added to $\mathcal{G}$. In the base case, if $|S| \leq 3$ then a complete graph on the set S is constructed. We first obtain a $(k, (12 + 15\epsilon))$-vertex

fault-tolerant additive weighted spanner for the set S of points lying on the terrain $\mathcal{T}$. (This construction is later modified to compute a $(k, (4 + \epsilon))$-VFTAWS.) In specific, with every projected point p_γ, instead of associating $d_\mathcal{T}(p, p_\gamma)$ as the weight of p_γ, we associate $w(p) + d_\mathcal{T}(p, p_\gamma)$ as the weight of p_γ. The rest of the algorithm in constructing $\mathcal{G}$ remains the same as in [6].

Algorithm 4: k-ADDITIVETERRAINCLUSTFTS($S, \mathcal{D}, k, \epsilon$)

Input : The polyhedral terrain $\mathcal{T}$, the set S on n additive weighted points, integer $k \geq 1$ and a real number $0 < \epsilon \leq 1$

Output: $(k, (12 + \epsilon))$-VFTS $\mathcal{G}$

1: **while** $card(\mathcal{T} \cap S) \geq 1$ **do**
2:     compute a balanced sp-separator for $\mathcal{T}$ using the algorithm given in [6]
3:     if the balanced sp-separator is a shortest path $\pi(u, v)$ between $u, v \in \partial \mathcal{T}$, then define set S_{in} to be $\pi^+(u, v) \cap S$; otherwise, define set S_{in} to be $\Delta \cap S$, where Δ is a sp-triangle.
4:     further, define the set S_{out} to be $S \setminus S_{in}$
5:     **for** every bounding shortest path γ of the sp-separator **do**
6:         initialize the set S_γ of the projections to ϕ
7:         **for** every $p \in S$ **do**
8:             find the projection p_γ of p on γ
9:             assign a weight equal to $w(p) + d_\mathcal{T}(p, p_\gamma)$ to p_γ
10:            $S_\gamma := S_\gamma \cup \{p_\gamma\}$
11:         **end for**
12:         compute a $(k, (4 + \epsilon))$-vertex fault-tolerant spanner $\mathcal{G}_\gamma$ for the set S_γ using Algorithm 1
13:         for every edge (r, s) in $\mathcal{G}_\gamma$, add the edge (p, q) to $\mathcal{G}$ where r(resp. s) is the projection of p (resp. q) on l
14:     **end for**
15:     k-ADDITIVETERRAINCLUSTFTS(S_{in}, X, k, ϵ)
        //X set to is $\pi^+(u, v)$ if the balanced sp-separator is a shortest path $\pi(u, v)$; else it is set to Δ
16:     k-ADDITIVETERRAINFTS(S_{out}, Y, k, ϵ)
        // Y set to is $\mathcal{T} \setminus \pi^+(u, v)$ if the balanced sp-separator is a shortest path $\pi(u, v)$; else it is set to $\mathcal{T} \setminus \Delta$
17: **end while**

To prove the graph $\mathcal{G}$ is a geodesic $(k, (12 + 15\epsilon))$-VFTAWS for the points in S, we use induction on the number of points. Consider any set $S' \subset S$ such that $|S'| \leq k$ and two arbitrary points p and q from the set $S \setminus S'$. We show that there exists a path between p and q in $\mathcal{G} \setminus S'$ such that $d_\mathcal{G}(p, q)$ is at most $(12 + 15\epsilon)d_{\pi,w}(p, q)$. The induction hypothesis assumes that for the number of points $k' < |S|$ in a region of $\mathcal{T}$, there exists a $(12 + 15\epsilon)$-spanner path between any two points belonging to the given region in $\mathcal{G} \setminus S'$. As part of the inductive step, we extend it to n points. For the case of both p and q are on the same side of a bounding shortest path γ of the balanced separator, i.e., both are in S_{in} or S_{out}, by induction hypothesis (as the number of points in S_{in} or S_{out} is less than $|S|$), there exists a $(12 + 15\epsilon)$-spanner path between p and q in $\mathcal{G} \setminus S'$. The only case remains to be proved is when p lies on one side of γ and q lies on the other side of γ, i.e., $p \in S_{in}$ and $q \in S_{out}$ or, $q \in S_{in}$ and $p \in S_{out}$. W.l.o.g., we assume that the former holds. Let r be a point on γ at which the geodesic shortest path $\pi(p, q)$ between p and q intersects γ. Since $\mathcal{G}_\gamma$ is a $(k, (4 + 5\epsilon))$-VFTS, there exists a path P between p_γ and q_γ in $\mathcal{G}_\gamma$ of length at most $(4 + 5\epsilon).d_{\gamma,w}(p_\gamma, q_\gamma)$. Let P' be the path obtained by replacing each vertex x_γ of P by $x \in S$ such that x_γ is the projection of x on γ. Note that the path P' is between nodes p and q in $\mathcal{G} \setminus S'$. The length $d_{\mathcal{G} \setminus S'}(p, q)$ of path P' is less than or equal to the length of the path P in $\mathcal{G}_\gamma$. In the following, we show that $d_{\mathcal{G} \setminus S'}(p, q) \leq (12 + 15\epsilon)d_{\mathcal{T},w}(p, q)$.

For every $x, y \in S$,
$$d_{\mathcal{T},w}(x,y) = w(x) + d_{\mathcal{T}}(x,y) + w(y)$$
$$\leq w(x) + d_{\mathcal{T}}(x, x_\gamma) + d_{\mathcal{T}}(x_\gamma, y_\gamma) + d_{\mathcal{T}}(y_\gamma, y) + w(y)$$
[by the triangle inequality]
$$= w(x_\gamma) + d_{\mathcal{T}}(x_\gamma, y_\gamma) + w(y_\gamma)$$
[since the weight associated with projection z_γ of every point z is $w(z) + d_{\mathcal{T}}(z, z_\gamma)$]
$$= d_{\gamma,w}(x_\gamma, y_\gamma). \tag{25}$$

This implies,
$$d_{\mathcal{G} \setminus S'}(p,q) = \sum_{x_\gamma, y_\gamma \in P} d_{\mathcal{T},w}(x,y)$$
$$\leq \sum_{x_\gamma, y_\gamma \in P} d_{\gamma,w}(x_\gamma, y_\gamma)$$
[from (25)]
$$\leq (4 + 5\epsilon).d_{\gamma,w}(p_\gamma, q_\gamma) \tag{26}$$
[since $\mathcal{G}_\gamma$ is a $(k, (4+5\epsilon))$-vertex fault tolerant geodesic spanner]
$$= (4 + 5\epsilon).[w(p_\gamma) + d_{\mathcal{T}}(p_\gamma, q_\gamma) + w(q_\gamma)]$$
[since γ is a shortest path on $\mathcal{T}$, shortest path between any two points on γ is a geodesic shortest path on $\mathcal{T}$]
$$= (4 + 5\epsilon).[w(p) + d_{\mathcal{T}}(p, p_\gamma) + d_{\mathcal{T}}(p_\gamma, q_\gamma) + d_{\mathcal{T}}(q_\gamma, q) + w(q)] \tag{27}$$
[since the weight associated with projection z_γ is $w(z) + d_{\mathcal{T}}(z, z_\gamma)$].

By the definition of projection of any point on γ, we know that
$$d_{\mathcal{T}}(p, p_\gamma) \leq d_{\mathcal{T}}(p, r) \text{ and } d_{\mathcal{T}}(q, q_\gamma) \leq d_{\mathcal{T}}(q, r). \tag{28}$$

Substituting (28) into (27),
$$d_{\mathcal{G} \setminus S'}(p,q) \leq (4 + 5\epsilon).[w(p) + d_{\mathcal{T}}(p,r) + d_{\mathcal{T}}(p_\gamma, q_\gamma) + d_{\mathcal{T}}(r,q) + w(q)]$$
$$\leq (4 + 5\epsilon).[w(p) + d_{\mathcal{T}}(p,r) + w(r) + d_{\mathcal{T}}(p_\gamma, q_\gamma) + w(r) + d_{\mathcal{T}}(r,q) + w(q)]$$
[since the weight of every point is non-negative]
$$= (4 + 5\epsilon).[d_{\mathcal{T},w}(p,r) + d_{\mathcal{T}}(p_\gamma, q_\gamma) + d_{\mathcal{T},w}(r,q)]$$
$$= (4 + 5\epsilon).[d_{\mathcal{T},w}(p,q) + d_{\mathcal{T}}(p_\gamma, q_\gamma)] \tag{29}$$
[since $\pi(p,q)$ intersects γ at r]
paths $\pi(p,q) = \pi(p,r) + \pi(r,q)$].

Further,
$$d_{\mathcal{T}}(p_\gamma, q_\gamma) \le d_{\mathcal{T}}(p_\gamma, p) + d_{\mathcal{T}}(p, q) + d_{\mathcal{T}}(q, q_\gamma)$$
$$\text{[by the triangle inequality]}$$
$$\le d_{\mathcal{T}}(r, p) + d_{\mathcal{T}}(p, q) + d_{\mathcal{T}}(q, r)$$
$$\text{[using (28)]}$$
$$\le w(r) + d_{\mathcal{T}}(r, p) + w(p) + w(p) + d_{\mathcal{T}}(p, q) + w(q) + w(q) +$$
$$d_{\mathcal{T}}(q, r) + w(r)$$
$$\text{[since the weight of every point is non-negative]}$$
$$= d_{\mathcal{T},w}(p, r) + d_{\mathcal{T},w}(p, q) + d_{\mathcal{T},w}(r, q)$$
$$= d_{\mathcal{T},w}(p, q) + d_{\mathcal{T},w}(p, q)$$
$$\text{[since } \pi(p, q) \text{ intersects } \gamma \text{ at } r]$$
$$= 2d_{\mathcal{T},w}(p, q). \tag{30}$$

Substituting (30) into (29) yields, $d_{\mathcal{G}\setminus S'}(p, q) \le 3(4 + 5\epsilon).d_{\mathcal{T},w}(p, q)$.

We improve the stretch factor of $\mathcal{G}$ by applying the same refinement as the one used in the algorithm in Section 4. Again, we denote the graph resulted after applying that refinement with $\mathcal{G}$.

Theorem 6. *Let S be a set of n weighted points on a polyhedral terrain $\mathcal{T}$ with non-negative weights associated to points via weight function w. For any fixed constant $\epsilon > 0$, there exists a $(k, (4 + \epsilon))$-vertex fault-tolerant additive weighted geodesic spanner with $O(\frac{kn}{\epsilon^2} \lg n)$ edges.*

Proof: The argument for the number of edges is same as in the proof of Theorem 4. To prove that the graph $\mathcal{G}$ is a geodesic $(k, (4 + 14\epsilon))$-VFTAWS for the points in S, we use induction on $|S|$. W.l.o.g., we assume that $p \in S_{in}$ and $q \in S_{out}$. Let r be the point at which the geodesic shortest path $\pi(p, q)$ between p and q intersects r. Since $\mathcal{G}_\gamma$ is a $(k, (4 + 5\epsilon))$-VFTS, there exists a path R between p_γ and q_γ in $\mathcal{G}_\gamma$ of length at most $(4 + 5\epsilon).d_{\gamma,w}(p_\gamma, q_\gamma)$. By replacing each vertex x_γ of R by $x \in S$ such that x_γ is the projection of x on γ, yields a path R' between p and q in $\mathcal{G} \setminus S'$. The length $d_{\mathcal{G}\setminus S'}(p, q)$ of path R' is less than or equal to the length of the path R in $\mathcal{G}_\gamma$. If $r \notin \gamma(p)$, point p'_γ (resp. q'_γ) is set as p_γ (resp. q_γ). Otherwise, p'_γ (resp. q'_γ) is set as the point in $S(p, \gamma)$ (resp. $S(q, \gamma)$) that is nearest to p (resp. q).

$$d_{\gamma,w}(p'_\gamma, q'_\gamma) = w(p'_\gamma) + d_\gamma(p'_\gamma, q'_\gamma) + w(q'_\gamma)$$
$$\le w(p'_\gamma) + d_\gamma(p'_\gamma, r) + d_\gamma(r, q'_\gamma) + w(q'_\gamma)$$
$$\text{[by the triangle inequality]}$$
$$\le w(p'_\gamma) + d_\gamma(p'_\gamma, r) + w(r) + w(r) + d_\gamma(r, q'_\gamma) + w(q'_\gamma)$$
$$\text{[since the weight of each point is non-negative]}$$
$$= w(p) + d_{\mathcal{T}}(p, p'_\gamma) + d_\gamma(p'_\gamma, r) + w(r) + w(r) + d_\gamma(r, q'_\gamma) +$$
$$d_{\mathcal{T}}(q'_\gamma, q) + w(q) \tag{31}$$
$$\text{[due to the association of weight to the projections of points]}.$$

From the triangle inequality, we know that $d_{\mathcal{T}}(p, p'_\gamma) + d_\gamma(p'_\gamma, r) \le d_{\mathcal{T}}(p, r)$, and $d_\gamma(r, q'_\gamma) + d_{\mathcal{T}}(q'_\gamma, q) \le d_{\mathcal{T}}(r, q)$. Hence (31) is written as

$$d_{\mathcal{T},w}(p'_\gamma, q'_\gamma) \le w(p) + d_{\mathcal{T}}(p, r) + w(r) + w(r) + d_{\mathcal{T}}(r, q) + w(q)$$
$$= d_{\mathcal{T},w}(p, r) + d_{\mathcal{T},w}(r, q)$$
$$= d_{\mathcal{T},w}(p, q) \tag{32}$$
$$\text{[since } \pi(p, q) \text{ intersects } \gamma \text{ at } r].$$

20

Replacing p_γ (resp. q_γ) by $p_\gamma{}'$ (resp. $q_\gamma{}'$) in inequality (26),
$$d_{\mathcal{G} \setminus S'}(p, q) \leq (4 + 5\epsilon).d_{\gamma,w}(p'_\gamma, q'_\gamma)$$
$$\leq (4 + 5\epsilon)d_{\mathcal{T},w}(p, q) \quad \text{[from (32).]}$$

Thus $\mathcal{G}$ is a geodesic $(k, (4 + \epsilon))$-VFTAWS for S. $\qquad\qquad\qquad\square$

6   Conclusions

In this paper, we gave algorithms to achieve k vertex fault-tolerance when the metric is additive weighted. We devised algorithms to compute a $(k, 4+\epsilon)$-VFTAWS when the input points belong to any of the following: $\mathbb{R}^d$, simple polygon, polygonal domain, and the terrain. Apart from the efficient computation, it would be interesting to explore the lower bounds on the number of edges for the fault-tolerant additive weighted spanners. Besides, the future work in the context of additive spanners could include finding the relation between the vertex-fault tolerance and the edge-fault tolerance, and optimizing various spanner parameters, like degree, diameter and weight.

References

1. M. A. Abam, M. Adeli, H. Homapour, and P. Z. Asadollahpoor. Geometric spanners for points inside a polygonal domain. In *Proceedings of Symposium on Computational Geometry*, pages 186–197, 2015.
2. M. A. Abam and M. de Berg. Kinetic spanners in $\mathbb{R}^d$. *Discrete & Computational Geometry*, 45(4):723–736, 2011.
3. M. A. Abam, M. de Berg, M. Farshi, and J. Gudmundsson. Region-fault tolerant geometric spanners. *Discrete & Computational Geometry*, 41(4):556–582, 2009.
4. M. A. Abam, M. de Berg, M. Farshi, J. Gudmundsson, and M. H. M. Smid. Geometric spanners for weighted point sets. *Algorithmica*, 61(1):207–225, 2011.
5. M. A. Abam, M. de Berg, and J. Gudmundsson. A simple and efficient kinetic spanner. *Computational Geometry*, 43(3):251–256, 2010.
6. M. A. Abam, M. de Berg, and M. J. R. Seraji. Geodesic spanners for points on a polyhedral terrain. In *Proceedings of Symposium on Discrete Algorithms*, pages 2434–2442, 2017.
7. A. K. Abu-Affash, R. Aschner, P. Carmi, and M. J. Katz. Minimum power energy spanners in wireless ad hoc networks. *Wireless Networks*, 17(5):1251–1258, 2011.
8. N. Alon, P. D. Seymour, and R. Thomas. Planar separators. *SIAM Journal on Discrete Mathematics*, 7(2):184–193, 1994.
9. I. Althöfer, G. Das, D. Dobkins, D. Joseph, and J. Soares. On sparse spanners of weighted graphs. *Discrete & Computational Geometry*, 9(1):81–100, 1993.
10. S. R. Arikati, D. Z. Chen, L. P. Chew, G. Das, M. H. M. Smid, and C. D. Zaroliagis. Planar spanners and approximate shortest path queries among obstacles in the plane. In *Proceedings of European Symposium on Algorithms*, pages 514–528, 1996.
11. B. Aronov, M. de Berg, O. Cheong, J. Gudmundsson, H. Haverkort, M. Smid, and A. Vigneron. Sparse geometric graphs with small dilation. *Computational Geometry*, 40(3):207–219, 2008.
12. S. Arya, G. Das, D. M. Mount, J.S. Salowe, and M. H. M. Smid. Euclidean spanners: short, thin, and lanky. In *Proceedings of Annual ACM Symposium on Theory of Computing*, pages 489–498, 1995.
13. S. Arya, D. M. Mount, and M. Smid. Dynamic algorithms for geometric spanners of small diameter: Randomized solutions. *Computational Geometry*, 13(2):91–107, 1999.
14. S. Arya, D. M. Mount, and M. H. M. Smid. Randomized and deterministic algorithms for geometric spanners of small diameter. In *Proceedings of Annual Symposium on Foundations of Computer Science*, pages 703–712, 1994.
15. S. Arya and M. H. M. Smid. Efficient construction of a bounded-degree spanner with low weight. *Algorithmica*, 17(1):33–54, 1997.
16. P. Bose, P. Carmi, M. Couture, A. Maheshwari, M. Smid, and N. Zeh. Geometric spanners with small chromatic number. *Computational Geometry*, 42(2):134–146, 2009.
17. P. Bose, P. Carmi, M. Farshi, A. Maheshwari, and M. Smid. Computing the greedy spanner in near-quadratic time. *Algorithmica*, 58(3):711–729, 2010.

18. P. Bose, P. Carmi, Chaitman L., S. Collette, M. J. Katz, and S. Langerman. Stable roommates spanner. *Computational Geometry*, 46(2):120–130, 2013.

19. P. Bose, P. Carmi, and Couture M. Spanners of additively weighted point sets. In *Proceedings of Scandinavian Workshop on Algorithm Theory*, pages 367–377, 2008.

20. P. Bose, J. Czyzowicz, E. Kranakis, D. Krizanc, and A. Maheshwari. Polygon cutting: revisited. In *Japanese Conference on Discrete and Computational Geometry and Graphs*, pages 81–92, 1998.

21. P. Bose, R. Fagerberg, A. van Renssen, and S. Verdonschot. On plane constrained bounded-degree spanners. *Algorithmica*, 2018.

22. P. Bose, J. Gudmundsson, and M. Smid. Constructing plane spanners of bounded degree and low weight. *Algorithmica*, 42(3):249–264, 2005.

23. P. Bose, M. Smid, and D. Xu. Delaunay and diamond triangulations contain spanners of bounded degree. *International Journal of Computational Geometry & Applications*, 19(02):119–140, 2009.

24. P. Carmi and L. Chaitman. Bounded degree planar geometric spanners. *CoRR*, abs/1003.4963, 2010.

25. P. Carmi and L. Chaitman. Stable roommates and geometric spanners. In *Proceedings of the 22nd Annual Canadian Conference on Computational Geometry*, pages 31–34, 10 2010.

26. P. Carmi and M. H. M. Smid. An optimal algorithm for computing angle-constrained spanners. *Journal of Computational Geometry*, 3(1):196–221, 2012.

27. L. P. Chew. There are planar graphs almost as good as the complete graph. *Journal of Computer and System Sciences*, 39(2):205–219, 1989.

28. A. Czumaj and H. Zhao. Fault-tolerant geometric spanners. *Discrete & Computational Geometry*, 32(2):207–230, 2004.

29. G. Das and D. Joseph. Which triangulations approximate the complete graph? In *Optimal Algorithms*, pages 168–192, 1989.

30. G. Das and G. Narasimhan. A fast algorithm for constructing sparse Euclidean spanners. *International Journal of Computational Geometry & Applications*, 7(4):297–315, 1997.

31. D. Eppstein. Spanning trees and spanners. In J.-R. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pages 425–461. Elsevier, 1999.

32. J. Gudmundsson and C. Knauer. Dilation and detour in geometric networks. In T. Gonzalez, editor, *Handbook of Approximation Algorithms and Metaheuristics*. Chapman & Hall, 2007.

33. J. Gudmundsson, C. Levcopoulos, and G. Narasimhan. Fast greedy algorithms for constructing sparse geometric spanners. *SIAM Journal on Computing*, 31(5):1479–1500, 2002.

34. S. Har-Peled and M. Mendel. Fast construction of nets in low-dimensional metrics and their applications. *SIAM Journal on Computing*, 35(5):1148–1184, 2006.

35. S. Kapoor and X.-Y. Li. Efficient construction of spanners in d-dimensions. *CoRR*, abs/1303.7217, 2013.

36. J. M. Keil and C. A. Gutwin. The Delaunay triangulation closely approximates the complete Euclidean graph. In F. Dehne, J.-R. Sack, and N. Santoro, editors, *Algorithms and Data Structures*, pages 47–56. Springer, 1989.

37. C. Levcopoulos, G. Narasimhan, and M. H. M. Smid. Improved algorithms for constructing fault-tolerant spanners. *Algorithmica*, 32(1):144–156, 2002.

38. T. Lukovszki. New results of fault tolerant geometric spanners. In *Proceedings of Workshop on Algorithms and Data Structures*, pages 193–204, 1999.

39. G. Narasimhan and M. H. M. Smid. *Geometric spanner networks*. Cambridge University Press, 2007.

40. D. Peleg and A. Schffer. Graph spanners. *Journal of Graph Theory*, 13(1):99–116, 1989.

41. M. Segal and H. Shpungin. Improved multi-criteria spanners for ad-hoc networks under energy and distance metrics. In *IEEE INFOCOM*, pages 6–10, 2010.

42. M. Smid. Geometric spanners with few edges and degree five. In J. Gudmundsson and C. B. Jay, editors, *CATS '06*, pages 7–9. Australian Computer Society, 2006.

43. S. Solomon. From hierarchical partitions to hierarchical covers: optimal fault-tolerant spanners for doubling metrics. In *Proceedings of Symposium on Theory of Computing*, pages 363–372, 2014.

44. K. Talwar. Bypassing the embedding: algorithms for low dimensional metrics. In *Proceedings of ACM Symposium on Theory of Computing*, pages 281–290, 2004.

45. Y. Wang and X.-Y. Li. Minimum power assignment in wireless ad hoc networks with spanner property. *Journal of Combinatorial Optimization*, 11(1):99–112, 2006.