SIAM J. DISCRETE MATH.
Vol. 38, No. 1, pp. 790–827

STABLE APPROXIMATION ALGORITHMS FOR THE DYNAMIC BROADCAST RANGE-ASSIGNMENT PROBLEM*

MARK DE BERG†, ARPAN SADHUKHAN†, AND FRITS SPIEKSMA†

Abstract. Let P be a set of points in $\mathbb{R}^d$, where each point $p \in P$ has an associated transmission range, denoted $\rho(p)$. The range assignment ρ induces a directed communication graph $\mathcal{G}_\rho(P)$ on P, which contains an edge (p,q) iff $|pq| \leqslant \rho(p)$. In the broadcast range-assignment problem, the goal is to assign the ranges such that $\mathcal{G}_\rho(P)$ contains an arborescence rooted at a designated root node and the cost $\sum_{p\in P}\rho(p)^2$ of the assignment is minimized. We study the dynamic version of this problem. In particular, we study trade-offs between the stability of the solution—the number of ranges that are modified when a point is inserted into or deleted from P—and its approximation ratio. To this end we study *k-stable algorithms,* which are algorithms that modify the range of at most k points when they update the solution. We also introduce the concept of a *stable approximation scheme,* or *SAS* for short. A SAS is an update algorithm ALG that, for any given fixed parameter $\varepsilon > 0$, is $k(\varepsilon)$-stable and that maintains a solution with approximation ratio $1+\varepsilon$, where the stability parameter $k(\varepsilon)$ only depends on ε and not on the size of P. We study such trade-offs in three settings. (1) For the problem in $\mathbb{R}^1$, we present a SAS with $k(\varepsilon) = O(1/\varepsilon)$. Furthermore, we prove that this is tight in the worst case: any SAS for the problem must have $k(\varepsilon) = \Omega(1/\varepsilon)$. We also present 1-, 2-, and 3-stable algorithms with constant approximation ratio. (2) For the problem in $\mathbb{S}^1$ (that is, when the underlying space is a circle) we prove that no SAS exists. This is in spite of the fact that, for the static problem in $\mathbb{S}^1$, we prove that an optimal solution can always be obtained by cutting the circle at an appropriate point and solving the resulting problem in $\mathbb{R}^1$. (3) For the problem in $\mathbb{R}^2$, we also prove that no SAS exists, and we present a $O(1)$-stable $O(1)$-approximation algorithm. Most results generalize to the setting where, for any given constant $\alpha > 1$, the range-assignment cost is $\sum_{p\in P}\rho(p)^\alpha$.

Key words. computational geometry, online algorithms, broadcast range assignement, stable approximation schemes, bounded recourse

MSC codes. 68Q25, 68R01, 68U05, 52C99

DOI. 10.1137/23M1545975

1. Introduction.

The broadcast range-assignment problem. Let P be a set of points in $\mathbb{R}^d$, representing transmission devices in a wireless network. By assigning each point $p \in P$ a transmission range $\rho(p)$, we obtain a *communication graph* $\mathcal{G}_\rho(P)$. The nodes in $\mathcal{G}_\rho(P)$ are the points from P and there is a directed edge (p,q) iff $|pq| \leqslant \rho(p)$, where $|pq|$ denotes the Euclidean distance between p and q. The energy consumption of a device depends on its transmission range: the larger the range, the more energy it needs. More precisely, the energy needed to obtain a transmission range $\rho(p)$ is given by $\rho(p)^\alpha$, for some real constant $\alpha > 1$ called the *distance-power gradient.* In practice, α depends on the environment and ranges from 1 to 6 [36]. Thus, if we denote the set of ranges given to the points in P by $\rho(P)$, then the total cost of a

*Received by the editors January 11, 2023; accepted for publication (in revised form) November 13, 2023; published electronically February 9, 2024. A preliminary version of this work appeared in the Proceedings of the 18th Scandinavian Symposium and Workshops on Algorithm Theory, SWAT 2022, [LIPIcs Leibniz Int. Proc. Inform. Vol. 227, Schloss Wagstuhl, Wadern, Germany].
https://doi.org/10.1137/23M1545975
Funding: The first, second, and third authors are supported by the Dutch Research Council (NWO) through Gravitation-grant NETWORKS-024.002.003.
†Department of Mathematics and Computer Science, TU Eindhoven, Eindhoven, the Netherlands (m.t.d.berg@tue.nl, a.sadhukhan@tue.nl, f.c.r.spieksma@tue.nl).

range assignment is $\mathrm{cost}_\alpha(\rho(P)) := \sum_{p\in P} \rho(p)^\alpha$. The goal of the range-assignment problem is to assign the ranges such that $\mathcal{G}_\rho(P)$ has certain connectivity properties while minimizing the total cost [12]. Desirable connectivity properties are that $\mathcal{G}_\rho(P)$ is (h-hop) strongly connected [14, 15, 16, 32] or that $\mathcal{G}_\rho(P)$ contains a *broadcast tree,* that is, an arborescence rooted at a given source $s \in P$. The latter property leads to the *broadcast range-assignment problem,* which is the topic of our paper.

The broadcast range-assignment problem has been studied extensively, sometimes with the extra condition that any point in P is reachable in at most h hops from the source s. For $\alpha = 1$, the problem is trivial in any dimension: setting the range of the source s to $\max\{|sp| : p \in P\}$ and all other ranges to zero is optimal. However, Fuchs [25] proves that, for any fixed $\alpha > 1$, the broadcast range-assignment problem is NP-hard in $\mathbb{R}^2$, and that the problem is NP-hard to approximate within $1 + \frac{1}{50}$ in $\mathbb{R}^3$. There are several approximation algorithms for the problem, typically based on finding a minimum spanning tree [10, 13]. Caragiannis, Flammini, and Moscardelli [9] describe a polynomial-time approximation algorithm that achieves an approximation ratio 4.2 in the case $d = 2$, a ratio 6.49 in the case $d = 3$, and a ratio $2.2d + 0.61$ in the case $d > 3$. As far as we are aware, these results summarize the state of the art of the approximability of the static broadcast range-assignment problem in various dimensions.

Many of our results will be on the 1-dimensional (or linear) broadcast range-assignment problem. Linear networks are important for modeling road traffic information systems [6, 34] and as such they have received ample attention. In $\mathbb{R}^1$, the broadcast range-assignment problem is no longer NP-hard, and several polynomial-time algorithms have been proposed, for the standard version, the h-hop version, as well as the weighted version [4, 10, 13, 19, 20]. The currently fastest algorithms for the (standard and h-hop) broadcast range-assignment problem run in $O(n^2)$ time [19].

All results mentioned so far are for the static version of the problem. Our interest lies in the dynamic version, where points can be inserted into and deleted from P (except the source, which is fixed and should always be present). This corresponds to new sensors being deployed and existing sensors being removed or, in a traffic scenario, cars entering and exiting the highway. Recomputing the range assignment from scratch when P is updated may result in all ranges being changed. The question we want to answer is, therefore, is it possible to maintain a close-to-optimal range assignment that is relatively stable, that is, an assignment for which only a few ranges are modified when a point is inserted into or deleted from P? And which trade-offs can be achieved between the quality of the solution and its stability?

To the best of our knowledge, the dynamic problem has not been studied so far. The online problem, where there are only insertions, was studied by De Berg, Markovic, and Umboh [22] under the restriction that it is not allowed to decrease ranges. This restriction is arguably unnatural, and it has the consequence that a bounded approximation ratio cannot be achieved. Indeed, let the source s be at $x = 0$, and suppose that first the point $x = 1$ arrives, forcing us to set $\rho(s) := 1$. Then the points $x = i/n$ arrive for $1 \leqslant i < n$. In the optimal static solution at the end of this scenario all points, except the rightmost one, have range $1/n$; for $\alpha = 2$ this induces a total cost of $n \cdot (1/n)^2 = 1/n$. But if we are not allowed to decrease the range of s after setting $\rho(s) = 1$, the total cost will be at least 1, leading to an unbounded approximation ratio. Therefore, De Berg, Markovic, and Umboh [22] analyze the *competitive* ratio: they compare the cost of their algorithm to the cost of an optimal offline algorithm, which knows the future arrivals, *but must still maintain a valid solution at all times without decreasing any range.* As we will see, by allowing one to also decrease a few ranges, we are able to maintain solutions whose cost is close even to the static optimum.

Stable algorithms. Taking a general perspective, let us consider an arbitrary dynamic optimization problem where we face the task of maintaining a feasible solution at all times. Our interest is in designing and analyzing update algorithms that perform this task with the objective to modify the existing solution as little as possible, while simultaneously having a good objective value. The number of modifications needed by such an algorithm will be referred to as its *stability*; we give precise definitions later. To analyze this property, we need to be able to define how two (consecutive) solutions differ [40]. For many problems there is a natural choice for the difference between two solutions. In our dynamic broadcast range-assignment problem, we focus on the number of points whose range changes after the arrival of a new point (an insertion), or after a point disappears (a deletion).

The concept of stability is not new and has been studied for online algorithms under the name of *bounded recourse.* It should be noted that in most papers on bounded-recourse algorithms, only insertions (and no deletions) are considered. Moreover, instead of considering the approximation ratio of the solution, the competitive ratio is considered. As explained above for the range-assignment problem, this can make a huge difference. Some representative examples of work related to bounded recourse include dynamic versions of problems such as minimum spanning trees and the Traveling Salesman Problem (TSP) [35], Steiner trees [26, 27, 30], knapsack [29, 31], packing problems [5], clustering [17, 24, 33], and matching [3, 7, 8, 28].

Our contribution. Before we state our results, we first define the framework we use to analyze our algorithms. Let P be a dynamic set of points in $\mathbb{R}^d$, which includes a fixed source point s that cannot be deleted.

An *update algorithm* ALG for the dynamic broadcast range-assignment problem is an algorithm that, given the current solution (the current ranges of the points in the current set P) and the location of the new point to be inserted into P, or the point to be deleted from P, modifies the range assignment so that the updated solution is a valid broadcast range assignment for the updated set P. We call such an update algorithm *k-stable* if it modifies at most k ranges when a point is inserted into or deleted from P. Here we define the range of a point currently not in P to be zero. Thus, if a newly inserted point receives a positive range it will be counted as receiving a modified range; similarly, if a point with positive range is deleted, then it will be counted as receiving a modified range. To get a more detailed view of the stability, we sometimes distinguish between the number of increased ranges and the number of decreased ranges, in the worst case. When these numbers are k^+ and k^-, respectively, we say that ALG is *(k^+, k^-)-stable.* This is especially useful when we separately report on the stability of insertions and deletions; often, when insertions are (k_1, k_2)-stable then deletions will be (k_2, k_1)-stable.

We are not only interested in the stability of our update algorithms, but also in the quality of the solutions they provide. We measure this in the usual way, by considering the approximation ratio of the solution. *We emphasize that the approximation ratio compares the cost of the current solution to the static optimum for the current point set.* As mentioned, we are interested in trade-offs between the stability of an algorithm and its approximation ratio. Of particular interest are so-called stable approximation schemes, defined as follows.

DEFINITION 1.1. *A* stable approximation scheme*, or* SAS *for short, is an update algorithm* ALG *that, for any given yet fixed parameter $\varepsilon > 0$, is $k(\varepsilon)$-stable and that maintains a solution with approximation ratio $1+\varepsilon$, where the stability parameter $k(\varepsilon)$ only depends on ε and not on the size of P.*

Notice that in the definition of a SAS we do not take the computational complexity of the update algorithm into account. We point out that, in the context of dynamic scheduling problems (where jobs arrive and disappear in an online fashion, and it is allowed to reassign jobs), a related concept has been introduced under the name *robust PTAS*: a polynomial-time algorithm that, for any given parameter $\varepsilon > 0$, computes a $(1+\varepsilon)$-approximation with reassignment costs only depending on ε; see, e.g., [37, 38].

We now present our results. Recall that $\text{cost}_\alpha(\rho(P)) := \sum_{p\in P} \rho(p)^\alpha$, is the cost of a range assignment ρ, where $\alpha > 1$ is a fixed constant. To make the results easier to interpret, we state the results for $\alpha = 2$; the dependencies of the bounds on the parameter α can be found in the theorems presented in later sections.

- In section 3, we present a SAS for the broadcast range-assignment problem in $\mathbb{R}^1$, with $k(\varepsilon) = O(1/\varepsilon)$. We prove that this is tight in the worst case, by showing that any SAS for the problem must have $k(\varepsilon) = \Omega(1/\varepsilon)$.
- Our SAS (as well as some other algorithms) needs to know an optimal solution after each update. The fastest existing algorithms to compute an optimal solution in $\mathbb{R}^1$ run in $O(n^2)$ time. In section 2 we show how to recompute an optimal solution in $O(n \log n)$ time after each update, which we believe to be of independent interest. As a consequence, our SAS also runs in $O(n \log n)$ time per update.
- In section 4, we study the problem in $\mathbb{S}^1$, that is, when the underlying 1-dimensional space is circular. This version has, as far as we know, not been studied so far. We first prove that in $\mathbb{S}^1$ an optimal solution for the static problem can always be obtained by cutting the circle $\mathbb{S}^1$ at an appropriate point and solving the resulting problem in $\mathbb{R}^1$. This leads to an algorithm to solve the static problem optimally in $O(n^2 \log n)$ time. We also prove that, in spite of the similarity between the structure of an optimal solution in $\mathbb{S}^1$ and in $\mathbb{R}^1$, a SAS does not exist in $\mathbb{S}^1$.
- We consider in section 5 the problem in $\mathbb{R}^2$. Based on the no-SAS proof in $\mathbb{S}^1$, we show that the 2-dimensional problem does not admit a SAS either. In addition, we present a 17-stable 12-approximation algorithm for the 2-dimensional version of the problem.
- Finally, in section 6, we return to $\mathbb{R}^1$ and study algorithms with a very small stability parameter. There is a very simple 2-stable 2-approximation algorithm. We show that a 1-stable algorithm with bounded approximation ratio does not exist when both insertions and deletions must be handled. For the insertion-only case, however, we give a 1-stable $(6+2\sqrt{5})$-approximation algorithm. We have not been able to improve upon the approximation ratio 2 with a 2-stable algorithm, but we show that with a 3-stable we can get a 1.97-approximation.

2. Maintaining an optimal solution in $\mathbb{R}^1$. Before we can present our stable algorithms for the broadcast range-assignment problem in $\mathbb{R}^1$, we first introduce some terminology and we discuss the structure of optimal solutions. We also present an efficient subroutine to maintain an optimal solution.

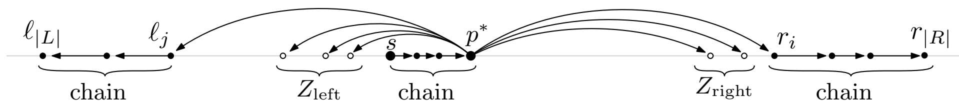

FIG. 1. *The structure of an optimal solution. The nonfilled points are zero-range points, the solid black points all have a standard range (for $\ell_{|L|}$ and $r_{|R|}$ the standard range is zero), except for the root-crossing point which (in this example) has a long range.*

2.1. The structure of an optimal solution. Several papers have characterized the structure of optimal broadcast range assignments in $\mathbb{R}^1$, in a more or less explicit manner. We use the characterization by Caragiannis, Kaklamanis, and Kanellopoulos [10], which is illustrated in Figure 1 and described next.

Let $P := L \cup \{s\} \cup R$ be a point set in $\mathbb{R}^1$. Here s is the designated source node, $L := \{\ell_1, \ldots, \ell_{|L|}\}$ contains all points from P to the left of s, and $R := \{r_1, \ldots, r_{|R|}\}$ contains all points to the right of s. The points in L are numbered in order of increasing distance from s, and the same is true for the points in R. The points $\ell_{|L|}$ and $r_{|R|}$ are called *extreme points*. In the following, and with a slight abuse of notation, we sometimes use p or q to refer a generic point from P—that is, a point that could be s, or a point from R, or a point from L. Furthermore, we will not distinguish between points in P and the corresponding nodes in the communication graph $\mathcal{G}_\rho(P)$.

For a nonextreme point $r_i \in R$, we define r_{i+1} to be its *successor*; similarly, ℓ_{i+1} is the successor of ℓ_i. The source s has (at most) two successors, namely, r_1 and ℓ_1. The successor of a point p is denoted by $\text{succ}(p)$; for an extreme point p we define $\text{succ}(p) = \text{NIL}$. If $\text{succ}(p) = q \neq \text{NIL}$, then we call p the *predecessor* of q and we write $\text{pred}(q) = p$. A *chain* is a path in the communication graph $\mathcal{G}_\rho(P)$ that only consists of edges connecting a point to its successor. Thus a chain either visits consecutive points from $\{s\} \cup R$ from left to right, or it visits consecutive points from $\{s\} \cup L$ from right to left. It will be convenient to consider the empty path from s to itself to be a chain as well.

Consider a range assignment ρ. We say that a point $q \in P$ is *within reach* of a point $p \in P$ if $|pq| \leqslant \rho(p)$. Let $\mathcal{B}$ be a broadcast tree in $\mathcal{G}_\rho(P)$—that is, $\mathcal{B}$ is an arborescence rooted at s. A point in $R \cup L$ in $\mathcal{B}$ is called a *root-crossing* in $\mathcal{B}$ if it has a child on the other side of s; the source s is root-crossing if it has a child in L and a child in R. Here we say that a point q is a child of p if (p, q) is a (directed) edge in the broadcast tree $\mathcal{B}$. The following theorem, which holds for any distance-power gradient $\alpha > 1$, is proven by Caragiannis, Kaklamanis, and Kanellopoulos [10].

THEOREM 2.1 ([10]). *Let P be a point set in $\mathbb{R}^1$. If all points in $P \setminus \{s\}$ lie to the same side of the source s, then the optimal solution induces a chain from s to the extreme point in P. Otherwise, there is an optimal range assignment ρ such that $\mathcal{G}_\rho(P)$ contains a broadcast tree $\mathcal{B}$ with the following structure:*

- *$\mathcal{B}$ has a single root-crossing point, p^*.*
- *$\mathcal{B}$ contains a chain from s to p^*.*
- *All points within reach of p^*, except those on the chain from s to p^*, are children of p^*.*
- *Let r_i and ℓ_j be the rightmost and leftmost point within reach of p^*, respectively. Then $\mathcal{B}$ contains a chain from r_i to $r_{|R|}$, and a chain from ℓ_j to $\ell_{|L|}$.*

From now on, whenever we talk about optimal range assignments and their induced broadcast trees, we implicitly assume that the broadcast tree has the structure described in Theorem 2.1. Note that the communication graph $\mathcal{G}_\rho(P)$ induced by an optimal range assignment ρ can contain more edges than the ones belonging to the broadcast tree $\mathcal{B}$. Obviously, for ρ to be optimal it must be a minimum-cost assignment inducing $\mathcal{B}$.

Define the *standard range* of a nonextreme point $r_i \in R$ to be $|r_i r_{i+1}|$; the standard range of the extreme point $r_{|R|}$ is defined to be zero. The standard ranges of the points in L are defined similarly. The source s has two standard ranges, $|s\ell_1|$ and $|sr_1|$. A range assignment in which every point has a standard range is called a *standard solution*; a standard solution may or may not be optimal. Note that, in the static problem, it is never useful to give a point a nonzero range that is smaller than its standard range(s). Hence, we only need to consider three types of points: *standard-range points, zero-range points,* and *long-range points.* Here zero-range points are nonextreme points with a zero range, and a point is said to have a *long range* if its range is greater than its standard range. Theorem 2.1 implies that an optimal range assignment has the following properties; see also Figure 1.

- There is at most one long-range point.
- The set $Z \subset P$ of zero-range points (which may be empty) can be partitioned into two subsets, Z_{left} and Z_{right}, such that Z_{left} consists of consecutive points that lie to the left of the source s, and Z_{right} consists of consecutive points to that lie to the right of s.

2.2. An efficient update algorithm. Using Theorem 2.1 an optimal solution for the broadcast range-assignment problem can be computed in $O(n^2)$ time [19]. Below we show that maintaining an optimal solution under insertions and deletions can be done more efficiently than by re-computing it from scratch: using a suitable data structure, we can update the solution in $O(n \log n)$ time, as stated in Theorem 2.4. We use this result in later sections, when we give algorithms that maintain a stable solution.

Recall that an optimal solution for a given point set P has a single root-crossing point, p^*. Once the range $\rho(p^*)$ is fixed, the solution is completely determined. Since $\rho(p^*) = |p^*p|$ for some point $p \neq p^*$, there are $n-1$ candidate ranges for a given choice of the root-crossing point p^*. The idea of our solution is to implicitly store the cost of the range assignment for each candidate range of p^* such that, upon the insertion or deletion of a point in P, we can find the best range for p^* in $O(\log n)$ time. By maintaining n such data structures $\mathcal{T}_{p^*}$, one for each choice of the root-crossing point p^*, we can then find the overall best solution.

Besides the data structures $\mathcal{T}_{p^*}$ which are described below, we also maintain a global data structure $\mathcal{T}_P$ that supports the following operations.

- Find the predecessor pred(q) and successor succ(q) in P of a query point q.
- Given two points $p, p' \in P$, report the total cost of the chain from p to p'.
- Insert or delete points from P.

By implementing $\mathcal{T}_P$ as a suitably augmented binary search tree, each of these operations can be performed in $O(\log n)$ time. In particular, $\mathcal{T}_P$ is a red-black tree on the points from P, where each internal node v is augmented with an extra field that stores the total cost of the chain on the points in the subtree rooted at v. How to maintain such an augmented tree and how to answer queries is standard (see the book by Cormen et al. [18, Chapter 15]) so we omit further details.

The data structure $\mathcal{T}_{p^*}$ for a given root-crossing point p^*. Next we explain our data structure for a given candidate root-crossing point p^*. We assume without loss of generality that p^* lies to the right of the source point s; it is straightforward to adapt the structure to the (symmetric) case where p^* lies to the left of s, and to the case where $p^* = s$.

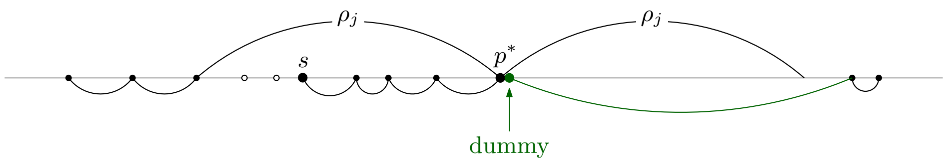

FIG. 2. *A dummy point is inserted to ensure that points to the right of p^* can be reached. For clarity the dummy is drawn to the right of p^*, but it actually coincides with p^*.*

Let $\mathcal{R}_{p^*}$ be the set of all ranges we need to consider for p^* for the current set P. The range of a root-crossing point must extend beyond the source point. Hence,

$$\mathcal{R}_{p^*} := \{|p^*p| : p \in P \text{ and } |p^*p| > |p^*s|\}.$$

Let $\lambda_1, \ldots, \lambda_m$ denote the sequence of ranges in $\mathcal{R}_{p^*}$, ordered from small to large. (If $\mathcal{R}_{p^*} = \emptyset$, there is nothing to do and our data structure is empty.) As mentioned, once we fix a range λ_j for the given root-crossing point p^*, the solution is fully determined by Theorem 2.1: there is a chain from s to p^*, a chain from the rightmost point within range of p^* to the right-extreme point, and a chain from the leftmost point within range of p^* to the left-extreme point. We denote the resulting range assignment[1] for P by $\Gamma(P, p^*, \lambda_j)$.

There is one subtlety in the definition of $\Gamma(P, p^*, \lambda_j)$, namely, when there are no points within reach of p^* to, say, the right of p^*; see Figure 2. Such a solution can never be optimal, but we must maintain it nevertheless, because the range λ_j may become relevant later. To deal with this situation, we will insert a dummy point whose location coincides with p^* and that is defined to be the predecessor of $\text{succ}(p^*)$. The dummy will become a zero-range point as soon as an actual point is inserted that is within the range of p^* and lies to the same side of p^* as the dummy.

Our data structure, which implicitly stores the costs of the range assignments $\Gamma(P, p^*, \lambda_j)$ for all $\lambda_j \in \mathcal{R}_{p^*}$, is an augmented balanced binary search tree $\mathcal{T}_{p^*}$, defined as follows.

- The leaves of $\mathcal{T}_{p^*}$ are in one-to-one correspondence with the candidate ranges in $\mathcal{R}_{p^*}$: the leftmost leaf corresponds to λ_1, the next left to λ_2, and so on. From now on, with a slight abuse of notation, we use λ_j to refer to a range in $\mathcal{R}_{p^*}$ as well as to the corresponding leaf.
- Each leaf stores, besides the corresponding range λ_j, a value $f(\lambda_j)$. Together with certain other values stored in $\mathcal{T}_{p^*}$, the value $f(\lambda_j)$ will help us to determine $\Gamma(P, p^*, \lambda_j)$; see invariant (2.1) below.
- The internal nodes of $\mathcal{T}_{p^*}$ are augmented with extra information, as follows. For an internal node v, let $\mathcal{R}_{p^*}(v) \subseteq \mathcal{R}_{p^*}$ be the set of all ranges stored in the leaves of the subtree rooted at v. The node v stores the following additional information, besides the splitting values that we have because $\mathcal{T}_{p^*}$ is a search tree on the ranges in $\mathcal{R}_{p^*}$:
  - A *correction value* $\Delta(v) \in \mathbb{R}$.
  - A value $\text{min-cost}(v)$ defined as follows. For a range $\lambda_j \in \mathcal{R}_{p^*}(v)$ define the *local cost of* λ_j *at* v to be $f(\lambda_j) + \sum_u \Delta(u)$, where the sum is over all

[1] When all points in P lie to the same side of s, then the range assignment is formally not root-crossing, but we will permit ourselves this slight abuse of terminology. Notice that in this case the range assignment induced by considering s as a root-crossing point and setting $\rho(s) := |s\,\text{succ}(s)|$ gives a chain from s to the extreme point as the solution, which is optimal.

nodes u on the path from v (and including v) to λ_j. Then min-cost(v) is defined to be the minimum local cost over all ranges in $\mathcal{R}_{p^*}(v)$.

– A range $\lambda_j \in \mathcal{R}_{p^*}(v)$ whose local cost at v is min-cost(v). This range is denoted by best-range(v).

Our update algorithm will ensure the following invariant:

$$\text{For any range } \lambda_j \in \mathcal{R}_{p^*}, \text{ the total cost of } \Gamma(P,p^*,\lambda_j) \text{ is equal to } f(\lambda_j)+\sum_u \Delta(u), \text{ where the sum is over all nodes on the search path from } \mathrm{root}(\mathcal{T}_{p^*}) \text{ to } \lambda_j. \tag{2.1}$$

In other words, invariant (2.1) states that, for any range λ_j, the local cost of λ_j at the root of $\mathcal{T}_{p^*}$ is equal to the actual cost of $\Gamma(P,p^*,\lambda_j)$. Since $\mathcal{R}_{p^*}(\mathrm{root}(\mathcal{T}_{p^*})) = \mathcal{R}_{p^*}$, this implies that min-cost(root($\mathcal{T}_{p^*}$)) equals the minimum cost that can be obtained by a solution that uses p^* as a root-crossing point.

Updating the data structure. We now describe how to update the structure upon the insertion of a new point. Deletions can be handled in a symmetrical manner. To simplify the presentation, we assume that no two points in P coincide; the solution is easily adapted to the case where P is a multiset. Let Δ_j be the (signed) difference of the cost of the range assignment $\Gamma(P,p^*,\lambda_j)$ before and after the insertion of q, where Δ_j is positive if the cost increases. Figure 3 shows various possible values for Δ_j, depending on the location of the new point q. The figure is for the generic case, when $Z_{\text{left}}, Z_{\text{right}} \neq \emptyset$ and there are points to the right as well as to the left of the interval that are within reach of the root-crossing point p^*. Lemma 2.2, which is easy to verify, gives the values for Δ_j for all cases, where we write $p<q$ when a point p is to the left of a point q.

LEMMA 2.2. *Let* $\Delta_j := \mathrm{cost}(\Gamma(P\cup\{q\},p^*,\lambda_j)) - \mathrm{cost}(\Gamma(P,p^*,\lambda_j))$. *If* $s<q<p^*$ *or* $p^*<q<s$ *we have*

$$\Delta_j = |\,\mathrm{pred}(q)\;q|^\alpha + |q\;\mathrm{succ}(q)|^\alpha - |\,\mathrm{pred}(q)\;\mathrm{succ}(q)|^\alpha.$$

Otherwise we have

$$\Delta_j = \begin{cases} |q\;\mathrm{succ}(q)|^\alpha - |\,\mathrm{pred}(q)\;\mathrm{succ}(q)|^\alpha & \text{if } |p^*q| \leqslant \lambda_j < |p^*\;\mathrm{succ}(q)|, \\ 0 & \text{if } \lambda_j \geqslant |p^*\;\mathrm{succ}(q)| \\ & \text{or } \big(\lambda_j \geqslant |p^*q| \text{ and } \mathrm{succ}(q) = \textsc{nil}\big), \\ |\,\mathrm{pred}(q)\;q|^\alpha + |q\;\mathrm{succ}(q)|^\alpha - |\,\mathrm{pred}(q)\;\mathrm{succ}(q)|^\alpha & \\ & \text{if } \lambda_j < |p^*q| \text{ and } \mathrm{succ}(q) \neq \textsc{nil}, \\ |\,\mathrm{pred}(q)\;q|^\alpha & \text{if } \lambda_j < |p^*q| \text{ and } \mathrm{succ}(q) = \textsc{nil}. \end{cases}$$

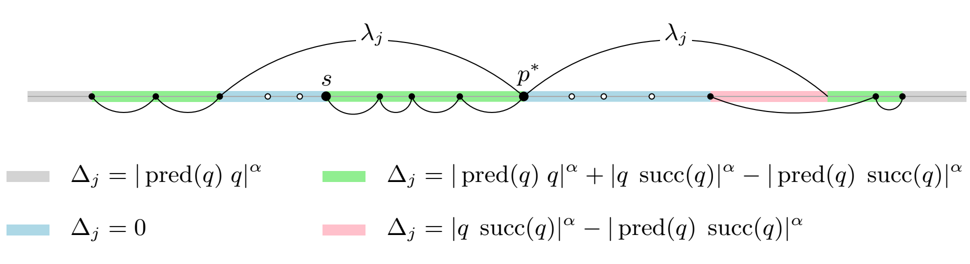

FIG. 3. *Various cases that can arise when a new point q is inserted into P. Open disks indicate zero-range points. The arcs indicate the ranges of the points before the insertion of q, where the range of the root-crossing point is drawn both to its right and to its left. The colored intervals relate the possible locations of q to the corresponding values Δ_j. Note: color appears only in the online article.*

Lemma 2.2 implies that, after computing pred(q) and succ(q), we can update our data structure using $O(1)$ *bulk updates* of the following form:

given an interval I of range values and an update value Δ, add Δ to the cost of $\Gamma(P,p^*,\lambda_j)$ for all $\lambda_j \in I$.

We cannot afford to do this explicitly, so we implement bulk updates by updating the auxiliary information stored in $O(\log n)$ nodes in $\mathcal{T}_{p^*}$, as follows.

1. Let $\lambda_{\min}$ and $\lambda_{\max}$ be the two endpoints of the interval I (possibly $\lambda_{\max}=\infty$). By searching with $\lambda_{\min}$ and $\lambda_{\max}$ in $\mathcal{T}_{p^*}$, identify a collection $\mathcal{C}(I)$ of $O(\log n)$ nodes in $\mathcal{T}_{p^*}$ such that $\lambda_i \in I$ iff the leaf storing λ_j is a descendant of a node in $\mathcal{C}(I)$.
2. Add Δ to the correction values $\Delta(v)$ of all nodes $v \in \mathcal{C}(I)$ and to the value min-cost(v).
3. Update the values $\Delta(v)$, min-cost(v), and best-range(v) of the $O(\log n)$ ancestors of the nodes in $\mathcal{C}(v)$ in a bottom-up manner.

Since algorithms for updating this type of auxiliary information are rather standard we omit further details. After updating the auxiliary information as described above, invariant (2.1) has been restored.

Besides updating the auxiliary information in the tree $\mathcal{T}_{p^*}$, we may also need to introduce another candidate range for p^*. In particular, we need to introduce the range $\lambda_{\text{new}} := |p^*q|$ if $|p^*q| > |p^*s|$. To this end we need to compute the cost of the range assignment $\Gamma(P,p^*,\lambda_{\text{new}})$. After computing pred($q$), succ($q$), and the cost of $O(1)$ chains—this can all be done in $O(\log n)$ time using the global tree $\mathcal{T}_P$—we can compute the cost of $\Gamma(P,p^*,\lambda_{\text{new}})$ in $O(1)$ time. We then insert a leaf w for the range λ_{new} into $\mathcal{T}_{p^*}$ with

$$f(\lambda_{\text{new}}) := \text{cost}(\Gamma(P,p^*,\lambda_{\text{new}})) - \sum_u \Delta(u),$$

where the sum is over all nodes on the search path from root($\mathcal{T}_{p^*}$) to w. Initializing $f(\lambda_{\text{new}})$ in this manner ensures that invariant (2.1) is satisfied for the new range λ_{new} as well. Finally, we update the values min-cost(v) and best-range(v) of the ancestors v of w whose current value of min-cost(v) is larger than $f(\lambda_{\text{new}})$). (Rebalancing $\mathcal{T}_{p^*}$, when necessary, can be done in a standard manner [18, Chapter 15].)

The following lemma summarizes the discussion above.

LEMMA 2.3. *$\mathcal{T}_{p^*}$ can be updated in $O(\log n)$ time per insertion and deletion.*

Putting it all together. To summarize, upon the insertion of a new point q into P, we first update each tree $\mathcal{T}_{p^*}$, as described above. This takes $O(\log n)$ time per tree, so $O(n \log n)$ time in total. Then we update the global tree $\mathcal{T}_P$ in $O(\log n)$ time. Finally, we create a tree $\mathcal{T}_q$ with q being the root-crossing point. This can be done in $O(n\log n)$ time, by inserting the points from P one by one as described above. Thus inserting a new point q can be done in $O(n \log n)$ time in total, after which we know the cost of the optimal solution for $P \cup \{q\}$. Deletions can be handled in a similar manner, so we obtain the following theorem.

THEOREM 2.4. *An optimal solution to the broadcast range-assignment problem for a point set P in $\mathbb{R}^1$ can be maintained in $O(n \log n)$ per insertion and deletion.*

3. A stable approximation scheme in $\mathbb{R}^1$.

In this section we use the structure of an optimal solution provided by Theorem 2.1 to obtain a SAS for the 1-dimensional broadcast range-assignment problem. Our SAS has stability parameter $k(\varepsilon) = O((1/\varepsilon)^{1/(\alpha-1)})$, which we will show to be asymptotically optimal.

The optimal range assignment can be very unstable. Indeed, suppose the current point set is $P := \{s, r_1, \ldots, r_n\}$ with $s = 0$ and $r_i = i$ $(1 \leqslant i \leqslant n)$, and take any $\alpha > 1$. Then the (unique) optimal assignment ρ_{opt} has $\rho_{\mathrm{opt}}(s) = \rho_{\mathrm{opt}}(r_1) = \cdots = \rho_{\mathrm{opt}}(r_{n-1}) = 1$ and $\rho_{\mathrm{opt}}(r_n) = 0$. If now the point $\ell_1 = -n$ is inserted, then the optimal assignment becomes $\rho_{\mathrm{opt}}(s) = n$ and $\rho_{\mathrm{opt}}(r_1) = \cdots = \rho_{\mathrm{opt}}(r_n) = \rho_{\mathrm{opt}}(\ell_1) = 0$, causing n ranges to be modified.

Next, we will define a feasible solution, referred to as a *canonical range assignment* ρ_k that is more stable than an optimal assignment, while still having a cost close to the cost of an optimal solution. Here k is a parameter that allows a trade-off between stability and quality of the solution. The assignment ρ_k for a given point set P will be uniquely determined by the set P; it does not depend on the order in which the points have been inserted or deleted. This means that the update algorithm simply works as follows. Let $\rho_k(P)$ be the canonical range assignment for a point set P, and suppose we update P by inserting a point q. Then the update algorithm computes $\rho_k(P \cup \{q\})$ and it modifies the range of each point $p \in P \cup \{q\}$ whose canonical range in $\rho_k(P \cup \{q\})$ is different from its canonical range in $\rho_k(P)$. The goal is now to specify ρ_k such that (i) many ranges in $\rho_k(P \cup \{q\})$ are the same as in $\rho_k(P)$, and (ii) the cost of $\rho_k(P)$ is close to the cost of $\rho_{\mathrm{opt}}(P)$.

The instance in the example above shows that there can be many points whose range changes from being standard to being zero (or vice versa) when preserving optimality of the consecutive instances. Our idea is therefore to construct solutions where the number of points with zero range is limited, and instead give many points their standard range; if we do this for points whose standard range is relatively small, then the cost of this solution remains bounded compared to the cost of an optimum solution. We now make this idea precise.

Consider a point set P and let ρ_{opt} be an optimal range assignment satisfying the structure described in Theorem 2.1. Assuming there are points in P on both sides of the source, ρ_{opt} induces a broadcast tree $\mathcal{B}$ with the structure depicted in Figure 1. Let $\rho_{\mathrm{st}}(p)$ be the standard range of a point $p \neq s$. The canonical range assignment ρ_k is now defined as follows.

- If all points from P lie to the same side of s, then $\rho_k(p) := \rho_{\mathrm{opt}}(p)$ for all $p \in P$. Note that in this case $\rho_k(p) = \rho_{\mathrm{st}}(p)$ for all $p \in P \setminus \{s\}$.
- Otherwise, let Z be the set of zero-range points in $\rho_{\mathrm{opt}}(P)$. If $|Z| \leqslant k$, then let $Z_k := Z$; otherwise let $Z_k \subseteq Z$ be the k points from Z with the largest standard ranges, with ties broken arbitrarily. We define ρ_k as follows.
  - $\rho_k(p) := \rho_{\mathrm{opt}}(p)$ for all $p \in P \setminus Z$. Observe that this means that $\rho_k(p) = \rho_{\mathrm{st}}(p)$ for all $p \in P \setminus (Z \cup \{s\})$ except (possibly) for the root-crossing point.
  - $\rho_k(p) := 0$ for all $p \in Z_k$.
  - $\rho_k(p) := \rho_{\mathrm{st}}(p)$ for all $p \in Z \setminus Z_k$.

Notice that ρ_k is a feasible solution since $\rho_k(p) \geqslant \rho_{\mathrm{opt}}(p)$ for each $p \in P$. The next lemma analyzes the stability of the canonical range assignment ρ_k. Recall that for any range assignment ρ—hence, also for ρ_k—and any point q not in the current set P, we have $\rho(q) = 0$ by definition.

LEMMA 3.1. *Consider a point set P and a point $q \notin P$. Let $\rho_{\mathrm{old}}(p)$ be the range of a point p in $\rho_k(P)$ and let $\rho_{\mathrm{new}}(p)$ be the range of p in $\rho_k(P \cup \{q\})$. Then*

$$|\{p \in P \cup \{q\} : \rho_{\mathrm{new}}(p) > \rho_{\mathrm{old}}(p)\}| \leqslant k + 3$$

and

$$|\{p \in P \cup \{q\} : \rho_{\text{new}}(p) < \rho_{\text{old}}(p)\}| \leqslant k+3.$$

Proof. The range of a point $p \in P \cup \{q\}$ can increase due to the insertion of q only if

(i) $p = q$ and $\rho_{\text{new}}(q) > 0$, or
(ii) p is a zero-range point in $\rho_k(P)$, or
(iii) p is the root-crossing point in $\rho_k(P \cup \{q\})$, or
(iv) the standard range of p increases due to the insertion of q, or
(v) $p = s$ and, out of the two standard ranges it has, s gets assigned a larger one in $\rho_k(P \cup \{q\})$ than in $\rho_k(P)$.

Recall that we defined ρ_k such that the number of zero-range points is at most k. Furthermore, at most one standard range can increase due to the insertion of q, namely, the standard range of a point that is extreme in P but not in $P \cup \{q\}$. When this happens, however, q is extreme in $P \cup \{q\}$ and so $\rho_{\text{new}}(q) = 0$; this implies that cases (i) and (iv) cannot happen simultaneously. To summarize: cases (i) and (iv) together contribute a range increase of at most one point; case (ii) contributes to at most k range increases; and cases (iii) and (v) each contribute a range increase of at most one point. Hence, $|\{p \in P \cup \{q\} : \rho_{\text{new}}(p) > \rho_{\text{old}}(p)\}| \leqslant k+3$.

The range of a point p can decrease only if

(vi) p is a zero-range point in $\rho_k(P \cup \{q\})$, or
(vii) p is the root-crossing point in $\rho_k(P)$, or
(viii) the standard range of p decreases due to the insertion of q, or
(ix) $p = s$ and, out of the two standard ranges it has, p gets a assigned a smaller one in $\rho_k(P \cup \{q\})$ than in $\rho_k(P)$.

Note that the only point whose standard range decreases is the predecessor of q in P, so case (viii) contributes to a range decrease of at most one point. Clearly, case (vii) and case (ix) each contribute to a range decrease of at most one point as well. Finally, case (vi) contributes to at most k range decreases. So we conclude that $|\{p \in P \cup \{q\} : \rho_{\text{new}}(p) < \rho_{\text{old}}(p)\}| \leqslant k+3$. $\square$

Observe that, while the insertion of q may increase the range of some points and decrease the range of some other points, not all combinations of cases (i)–(v) and (vi)–(ix) can happen. In particular, cases (iv) and (viii) cannot occur simultaneously (since both concern the predecessor of q) and cases (v) and (ix) cannot occur simultaneously (since both concern the source s). Thus the statement of the lemma could be strengthened accordingly. This would have no impact on our final result, Theorem 3.3, however.

Next we bound the approximation ratio of ρ_k.

LEMMA 3.2. *For any set P and any $\alpha > 1$, we have* $\text{cost}_\alpha(\rho_k(P)) \leqslant \left(1 + \frac{2^\alpha}{k^{\alpha-1}}\right) \cdot \text{cost}_\alpha(\rho_{\text{opt}}(P))$.

Proof. If all points in P lie to the same side of s, then $\rho_k(P) = \rho_{\text{opt}}(P)$, and we are done. Otherwise, let p^* be the root-crossing point. The only points receiving a different range in $\rho_k(P)$ when compared to $\rho_{\text{opt}}(P)$ are the points in $Z \setminus Z_k$; these points have $\rho_k(p) = \rho_{\text{st}}(p)$ while $\rho_{\text{opt}}(p) = 0$. This means we are done when $Z \setminus Z_k = \emptyset$. Thus we can assume that $|Z| > k$, so $Z \setminus Z_k \neq \emptyset$. Assume without loss of generality that $\rho_{\text{opt}}(p^*) = 1$. As each $p \in Z$ is within reach of p^*, we have $\sum_{p \in Z} \rho_{\text{st}}(p) \leqslant 2$. Since Z_k contains the k points with the largest standard ranges among the points in Z, we have $\max\{\rho_{\text{st}}(p) : p \in Z \setminus Z_k\} \leqslant 2/k$. Hence,

$$\begin{aligned}\textstyle\sum_{p\in Z\setminus Z_k}\rho_k(p)^\alpha &= \textstyle\sum_{p\in Z\setminus Z_k}\rho_{\mathrm{st}}(p)^\alpha\\ &= \textstyle\sum_{p\in Z\setminus Z_k}\rho_{\mathrm{st}}(p)^{\alpha-1}\cdot\rho_{\mathrm{st}}(p)\\ &\leqslant \left(\tfrac{2}{k}\right)^{\alpha-1}\textstyle\sum_{p\in Z\setminus Z_k}\rho_{\mathrm{st}}(p)\\ &\leqslant \tfrac{2^\alpha}{k^{\alpha-1}}.\end{aligned}$$

(The analysis can be made tighter by using $\sum_{p\in Z\setminus Z_k}\rho_{\mathrm{st}}(p)\leqslant 2-k\max_{p\in Z\setminus Z_k}\rho_{\mathrm{st}}(p)$, but this will not change the approximation ratio asymptotically.) We conclude that

$$\frac{\mathrm{cost}_\alpha(\rho_k(P))}{\mathrm{cost}_\alpha(\rho_{\mathrm{opt}}(P))}\leqslant\frac{\sum_{p\in P\setminus(Z\setminus Z_k)}\rho_k(p)^\alpha+\sum_{p\in Z\setminus Z_k}\rho_k(p)^\alpha}{\sum_{p\in P\setminus(Z\setminus Z_k)}\rho_{\mathrm{opt}}(p)^\alpha}\leqslant 1+\frac{2^\alpha}{k^{\alpha-1}},$$

where the last inequality follows because we have $\rho_k(p)=\rho_{\mathrm{opt}}(p)$ for all $p\in P\setminus(Z\setminus Z_k)$ and $\sum_{p\in P\setminus(Z\setminus Z_k)}\rho_{\mathrm{opt}}(p)^\alpha\geqslant 1$. ∎

By maintaining the canonical range assignment ρ_k for $k:=\lceil(2^\alpha/\varepsilon)^{1/(\alpha-1)}\rceil$—note that this is $O((1/\varepsilon)^{1/(\alpha-1)})$ since α, the distance-power gradient, is a fixed constant greater than 1—we obtain the following theorem.

THEOREM 3.3. *There is a SAS for the dynamic broadcast range-assignment problem in $\mathbb{R}^1$ with stability parameter $k(\varepsilon)=O((1/\varepsilon)^{1/(\alpha-1)})$, where $\alpha>1$ is a constant specifying the distance-power gradient. The time needed by the SAS to compute the new range assignment upon the insertion or deletion of a point is $O(n\log n)$, where n is the number of points in the current set. Moreover, any SAS for the dynamic broadcast range-assignment problem in $\mathbb{R}^1$ must have stability parameter $k(\varepsilon)=\Omega((1/\varepsilon)^{1/(\alpha-1)})$.*

Proof. We maintain the canonical range assignment ρ_k for $k=\lceil(2^\alpha/\varepsilon)^{1/(\alpha-1)}\rceil$. We then have $\mathrm{cost}_\alpha(\rho_k(P))\leqslant(1+\varepsilon)\cdot\rho_{\mathrm{opt}}(P)$ by Lemma 3.2. Furthermore, the number of modified ranges when P is updated is $2k+6$ by Lemma 3.1. This proves the first statement of Theorem 3.3.

To compute the assignment ρ_k, for some given k, we need to know an optimal assignment ρ_{opt} with the structure from Theorem 2.1. Such an optimal assignment can be maintained in $O(n\log n)$ time per update, by Theorem 2.4. Once we have the new optimal assignment, the new optimal assignment can trivially be determined in $O(n)$ time.

It remains to show that any SAS has stability parameter $k(\varepsilon)=\Omega((1/\varepsilon)^{1/(\alpha-1)})$. To this end let ALG be a k-stable algorithm, where $k\geqslant 4$ and $k^{\alpha-1}\geqslant\frac{1}{2^{\alpha+1}(2^{\alpha-1}-1)}$ and k is even, and let ρ_{alg} be the range assignment it maintains. Note that the condition on k is satisfied for k large enough. We will show that the approximation ratio of ALG is at least $1+\frac{1}{2^{\alpha+2}k^{\alpha-1}}$. Since a SAS has approximation ratio $1+\varepsilon$, this implies that the stability parameter $k(\varepsilon)$ of ALG must satisfy $k(\varepsilon)=\Omega((1/\varepsilon)^{1/(\alpha-1)})$. Consider the point set $P:=\{s,r_1,r_2,\dots,r_{2k}\}$, where $s=0$ and $r_i=i/(2k)$ for $i=1,2,\dots,2k$. We consider two cases.

Case I*: The number of zero-range points in $\rho_{\mathrm{alg}}(P)$ is at least $k/2$,* where we assume without loss of generality that all points with range less than $1/(2k)$ actually have range zero. It is easy to verify that the cheapest possible solution in this case is to have exactly $k/2$ zero-range points, k points with range $1/(2k)$, and $k/2$ points with range $1/k$, for a total cost of

$$\mathrm{cost}_\alpha(\rho_{\mathrm{alg}}(P))\geqslant k\cdot\left(\frac{1}{2k}\right)^\alpha+\frac{k}{2}\cdot\left(\frac{1}{k}\right)^\alpha=\left(1+\frac{2^{\alpha-1}-1}{2}\right)\cdot 2k\left(\frac{1}{2k}\right)^\alpha.$$

An optimal solution has cost $2k \cdot (1/(2k))^\alpha$, and so the approximation ratio of ALG in Case I is at least $1 + \frac{2^{\alpha-1}-1}{2}$, which is at least $1 + \frac{1}{2^{\alpha+2}k^{\alpha-1}}$ since $k^{\alpha-1} \geqslant \frac{1}{2^{\alpha+1}(2^{\alpha-1}-1)}$.

Case II*: The number of zero-range points $\rho_{\mathrm{alg}}(P)$ is less than $k/2$.* Now suppose the point $\ell_1 = -1$ arrives. Since $\rho_{\mathrm{alg}}(P)$ had less than $k/2$ zero-range points and ALG can modify at most k ranges, $\rho_{\mathrm{alg}}(P \cup \{\ell_1\})$ has less than $3k/2$ zero-range points. Hence, at least $k/2$ points in $P \cup \{\ell_1\}$ have a range that is at least $1/(2k)$, one of which must have a range at least 1. This implies that $\mathrm{cost}_\alpha(\rho_{\mathrm{alg}}(P \cup \{\ell_1\})) \geqslant 1 + (k/2-1) \cdot \left(\frac{1}{2k}\right)^\alpha \geqslant 1 + \frac{1}{2^{\alpha+2}k^{\alpha-1}}$, where the last inequality holds since $k/2 - 1 \geqslant k/4$ (because $k \geqslant 4$). An optimal range assignment on $P \cup \{\ell_1\}$ has $\rho_{\mathrm{opt}}(s) = 1$ and all other ranges equal to zero, for a total cost of 1, and so the approximation ratio of ALG in Case II is at least $1 + \frac{1}{2^{\alpha+2}k^{\alpha-1}}$ as well. ∎

4. The problem in $\mathbb{S}^1$. We now turn to the setting where the underlying space is $\mathbb{S}^1$, that is, the points in P lie on a circle and distances are measured along the circle. In section 4.1, we prove that the structure of an optimal solution in $\mathbb{S}^1$ is very similar to the structure of an optimal solution in $\mathbb{R}^1$ as formulated in Theorem 2.1. In spite of this, and contrary to the problem in $\mathbb{R}^1$, we prove in section 4.2 that no SAS exists for the problem in $\mathbb{S}^1$.

When discussing the problem in $\mathbb{S}^1$, we distinguish the *clockwise distance* from a point $p \in \mathbb{S}^1$ to a point $q \in \mathbb{S}^1$, denoted by $d_{\mathrm{cw}}(p,q)$, and the *counterclockwise distance,* denoted by $d_{\mathrm{ccw}}(p,q)$. The actual distance is then $d(p,q) := \min(d_{\mathrm{cw}}(p,q), d_{\mathrm{ccw}}(p,q))$. The closed and open clockwise intervals from p to q are denoted by $[p,q]^{\mathrm{cw}}$ and $(p,q)^{\mathrm{cw}}$, respectively. As before, the (fixed) source point is denoted by s.

4.1. The structure of an optimal solution in $\mathbb{S}^1$. Here we prove that the structure of an optimal solution in $\mathbb{S}^1$ is very similar to the structure of an optimal solution in $\mathbb{R}^1$. The heart of this proof is Lemma 4.1, stated next. Define the *covered region* of P with respect to a range assignment ρ, denoted by $\mathrm{cov}(\rho, P)$, to be the set of all points $r \in \mathbb{S}^1$ such that there exists a point $p \in P$ with $\rho(p) \geqslant d(p,r)$.

LEMMA 4.1. *Let P be a point set in $\mathbb{S}^1$ with $|P| > 2$ and let ρ_{opt} be an optimal range assignment for P. Then there exists a point $r \in \mathbb{S}^1$ such that $r \notin \mathrm{cov}(\rho_{\mathrm{opt}}, P)$.*

Lemma 4.1 implies that an optimal solution for an instance in $\mathbb{S}^1$ corresponds to an optimal solution for an instance in $\mathbb{R}^1$ in the following way. For a point $r \in \mathbb{S}^1$, define the mapping $\mu_r : P \to \mathbb{R}^1$ such that $\mu_r(s) := 0$, and $\mu_r(p) := d_{\mathrm{cw}}(s,p)$ for all $p \in [s,r]^{\mathrm{cw}}$, and $\mu_r(p) := -d_{\mathrm{ccw}}(s,p)$ for all $p \in [r,s]^{\mathrm{cw}}$. Let $\mu_r(P)$ denote the resulting point set in $\mathbb{R}^1$; informally, the mapping μ_r corresponds to "cutting" the cycle at r; see Figure 4. Then, by Lemma 4.1, there is a point $r \in \mathbb{S}^1$ such that an optimal solution

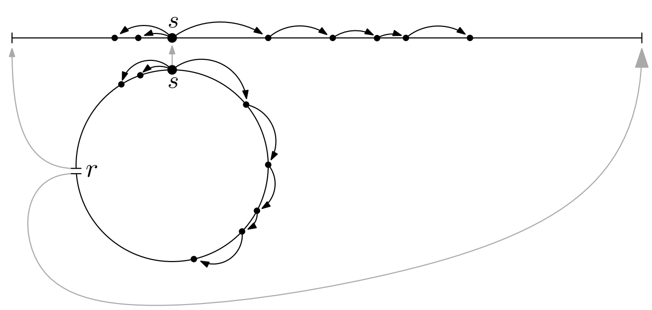

FIG. 4. *The mapping μ_r from $\mathbb{S}^1$ to $\mathbb{R}^1$. The solution for $\mathbb{S}^1$ that is shown has no arcs crossing the point r and so it induces a solution in $\mathbb{R}^1$ of exactly the same cost.*

for $\mu_r(P)$ induces an optimal solution for P. We postpone the proof of Lemma 4.1 to the end of this subsection, and proceed by stating the following result.

THEOREM 4.2. *Let P be an instance of the broadcast range-assignment problem in $\mathbb{S}^1$. There exists a point $r \in \mathbb{S}^1$ such that an optimal range assignment for $\mu_r(P)$ in $\mathbb{R}^1$ induces an optimal range assignment for P. Moreover, we can compute an optimal range assignment for P in $O(n^2 \log n)$ time.*

Proof. Let $r \in \mathbb{S}^1$ be a point such that $r \notin \text{cov}(\rho_{\text{opt}}, P)$ (such a point exists by Lemma 4.1). Consider the mapping μ_r. Any feasible range assignment for $\mu_r(P)$ induces a feasible range assignment for P in $\mathbb{S}^1$, since $d(p,q) \leqslant |\mu_r(p)\mu_r(q)|$ for any two points $p, q \in P$. Conversely, an optimal range assignment for P induces a feasible range assignment for $\mu_r(P)$, since the point r is not covered in the optimal solution. This proves the first part of the theorem.

We now show that an optimal range assignment for P can be computed in $O(n^2 \log n)$ time. Note that we cannot use an algorithm for the problem in $\mathbb{R}^1$ directly, since we do not know the point r where we have to cut $\mathbb{S}^1$. Hence, we proceed as follows. Let $P := \{s, p_1, \ldots, p_n\}$, where the points p_i are ordered clockwise from s. For $0 \leqslant i \leqslant n$, let r_i be a point in $(p_i, p_{i+1})^{\text{cw}}$, where $p_0 = p_{n+1} = s$. Since $\mu_{r_i} = \mu_r$ for any $r \in (p_i, p_{i+1})^{\text{cw}}$, an optimal solution can be computed by finding the best solution over all mappings μ_{r_i}. The only difference between μ_{r_i} and $\mu_{r_{i+1}}$ is the location that p_{i+1} is mapped to, so after computing an optimal solution for $\mu_1(P)$ in $O(n^2 \log n)$ time, we can go through the mappings $\mu_2, \ldots, \mu_n$ and update the optimal solution in $O(n \log n)$ time using Theorem 2.4. We then report the best of all solutions that were generated. This way, an optimal range assignment for P can be computed in $O(n^2 \log n)$ time. ∎

Without loss of generality we identify $\mathbb{S}^1$ with a circle of perimeter 1. Let ρ_{opt} be a fixed optimal range assignment on P. To prove Lemma 4.1 we will need the following lemma.

LEMMA 4.3. *If $|P| > 2$, then $\rho_{\text{opt}}(p) < \frac{1}{2}$ for all $p \in P$.*

Proof. Note that setting $\rho(s) = \frac{1}{2}$ and $\rho(p) = 0$ for all $p \in P \setminus \{s\}$ gives a feasible solution. Since $\rho(s) > 0$ in any feasible solution, this means that $\rho_{\text{opt}}(p) < \frac{1}{2}$ for all $p \neq s$. Hence, it suffices to show that $\rho_{\text{opt}}(s) < \frac{1}{2}$. If there is no point $p \in P$ which is diametrically opposite s, then clearly $\rho_{\text{opt}}(s) < \frac{1}{2}$. Now suppose some point $p \in P$ lies diametrically opposite s. Let $q \in P \setminus \{s, p\}$ be a point that maximizes the distance from s among all points in $P \setminus \{s, p\}$. The point q exists since $|P| > 2$. Note that $d(s,q) + d(q,p) = \frac{1}{2}$. Hence, setting $\rho(s) = d(s,q)$ and $\rho(q) = d(q,p)$ (and keeping all other ranges zero) gives a solution of cost $d(s,q)^\alpha + d(q,p)^\alpha$, which is less than $\left(\frac{1}{2}\right)^\alpha$ since $\alpha > 1$. Thus $\rho_{\text{opt}}(s) < \frac{1}{2}$, which finishes the proof. ∎

Before we proceed, we introduce some more notation.

Consider a directed edge (p,q) in a communication graph $\mathcal{G}_\rho(P)$. We say that (p,q) is a *clockwise edge* if $\rho(p) \geqslant d_{\text{cw}}(p,q)$, and we say that it is a *counterclockwise edge* if $\rho(p) \geqslant d_{\text{ccw}}(p,q)$. Lemma 4.3 implies that an edge cannot be both clockwise and counterclockwise in an optimal range assignment, assuming $|P| > 2$. Finally, we define the *covered region* of a subset $Q \subseteq P$ with respect to a range assignment ρ to be the set of all points $r \in \mathbb{S}^1$ such that there exists a point $p \in Q$ such that $\rho(p) \geqslant d(p,r)$. We denote this region by $\text{cov}(\rho, Q)$. Furthermore, the *counterclockwise covered region* of Q, denoted by $\text{cov}_{\text{ccw}}(\rho, Q)$, is the set of all points $r \in \mathbb{S}^1$ such that there exists a point $p \in Q$ such that $\rho(p) \geqslant d_{\text{ccw}}(p, r)$. The *clockwise covered region* of Q, denoted by $\text{cov}_{\text{cw}}(\rho, Q)$, is defined similarly.

We now have everything in place to prove Lemma 4.1, which we restate for the reader's convenience.

LEMMA 4.1. *Let P be a point set in $\mathbb{S}^1$ with $|P| > 2$ and let ρ_{opt} be an optimal range assignment for P. Then there exists a point $r \in \mathbb{S}^1$ such that $r \notin \text{cov}(\rho_{\text{opt}}, P)$.*

Proof. Let $d_{\text{hop}}(p, q)$ denote the hop distance from p to q in the communication graph $\mathcal{G}_{\rho_{\text{opt}}}(P)$. Let $\mathcal{B}$ be a broadcast tree rooted at s in $\mathcal{G}_{\rho_{\text{opt}}}(P)$ with the following properties.

- $\mathcal{B}$ is a shortest-path tree in terms of hop distance, that is, the hop distance from s to any point p in $\mathcal{B}$ is equal to $d_{\text{hop}}(s, p)$.
- Among all such shortest-path trees, $\mathcal{B}$ maximizes the number of clockwise edges.

For two points $p, q \in P$, let $\pi(p, q)$ denote the path from p to q in $\mathcal{B}$, and let $|\pi(p, q)|$ be its length, that is, the number of edges on the path. Note that $|\pi(s, p)| = d_{\text{hop}}(s, p)$ for any $p \in P$. Let $\text{pa}(p)$ denote the parent of a point p in $\mathcal{B}$ and define

$$S_{\text{cw}} = \{p \in P \setminus \{s\} : \ (\text{pa}(p), p) \text{ is a clockwise edge}\}$$

and

$$S_{\text{ccw}} = \{p \in P \setminus \{s\} : \ (\text{pa}(p), p) \text{ is a counterclockwise edge}\}.$$

Note that $S_{\text{cw}} \cup S_{\text{ccw}} = P \setminus \{s\}$. Now define

$$q_{\text{cw}} = \text{the point from } S_{\text{cw}} \text{ that maximizes } d_{\text{cw}}(s, p),$$

where $q_{\text{cw}} = s$ if $S_{\text{cw}} = \emptyset$. Similarly, define

$$q_{\text{ccw}} = \text{the point from } S_{\text{ccw}} \text{ that maximizes } d_{\text{ccw}}(s, p),$$

where $q_{\text{ccw}} = s$ if $S_{\text{ccw}} = \emptyset$. Let $\text{anc}(p)$ be the set of ancestors in $\mathcal{B}$ of a point $p \in P$, that is, $\text{anc}(p)$ contains the points of $\pi(s, p)$ excluding the point p. The following observation will be used repeatedly in the proof.

Claim. If $(\text{pa}(p), p)$ is a clockwise edge, then $[s, p]^{\text{cw}} \subset \text{cov}(\rho_{\text{opt}}, \text{anc}(p))$. Similarly, if $(\text{pa}(p), p)$ is a counterclockwise edge, then $[s, p]^{\text{ccw}} \subset \text{cov}(\rho_{\text{opt}}, \text{anc}(p))$.

Proof. Assume $(\text{pa}(p), p)$ is a clockwise edge; the proof for when $(\text{pa}(p), p)$ is a counterclockwise edge is similar. There are two cases, as illustrated in Figure 5.

If $s \in [\text{pa}(p), p)]^{\text{cw}}$—this includes the case where $\text{pa}(p) = s$—then the statement obviously holds, so assume $\text{pa}(p) \in [s, p]^{\text{cw}}$. Since $(\text{pa}(p), p)$ is a clockwise edge, it then suffices to prove that $[s, \text{pa}(p)]^{\text{cw}} \subset \text{cov}(\rho_{\text{opt}}, \text{anc}(p))$. Note that $\text{cov}(\rho_{\text{opt}}, \text{anc}(p))$ is connected, because the points in $\text{anc}(p)$ form a path, namely, $\pi(s, \text{pa}(p))$. Since $\pi(s, p)$ is the shortest path, $p \notin \text{cov}(\rho_{\text{opt}}, \text{anc}(\text{pa}(p))$, which implies that $[s, \text{pa}(p)]^{\text{cw}} \subset \text{cov}(\rho_{\text{opt}}, \text{anc}(\text{pa}(p))) \subset \text{cov}(\rho_{\text{opt}}, \text{anc}(p))$. ∎

We now proceed to show that q_{ccw} must lie clockwise from q_{cw}, as seen from s, that is, the situation shown in Figure 6(i) cannot happen.

Claim. $d_{\text{cw}}(s, q_{\text{cw}}) + d_{\text{ccw}}(s, q_{\text{ccw}}) < 1$.

Proof. Note that $d_{\text{cw}}(s, q_{\text{cw}}) + d_{\text{ccw}}(s, q_{\text{ccw}}) \neq 1$, since otherwise $q_{\text{cw}} = q_{\text{ccw}}$ which cannot happen since $S_{\text{cw}} \cap S_{\text{ccw}} = \emptyset$.

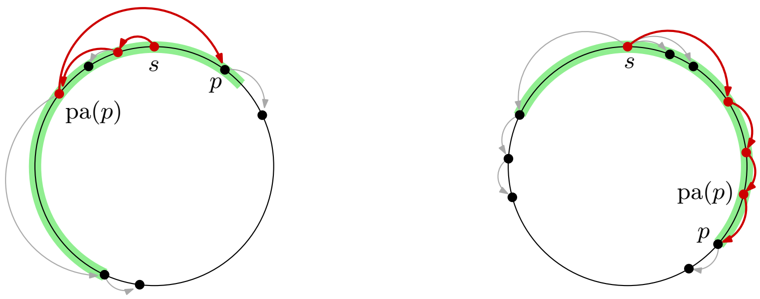

FIG. 5. *Two cases in the proof of the first claim in the proof of Lemma* 4.1. *The ancestors of* p *and the edges on the path* $\pi(s,p)$ *are shown in red, and* $\mathrm{cov}(\rho_{\mathrm{opt}}, \mathrm{anc}(p))$ *is shown in green. On the left we illustrate the case where* $s \in [\mathrm{pa}(p), p)]^{\mathrm{cw}}$, *and on the right the case where* $s \notin [\mathrm{pa}(p), p)]^{\mathrm{cw}}$. *Note: color appears only in the online article.*

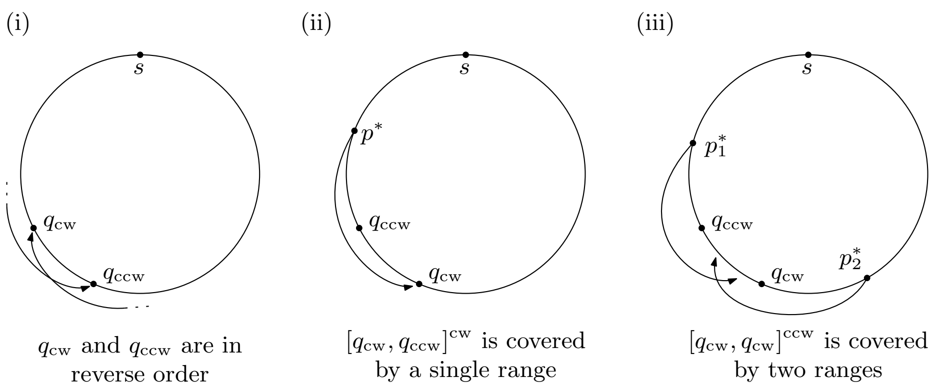

FIG. 6. *Illustration for the proof of Lemma* 4.1. *Note that the point* p^* *in part* (ii) *of the figure could also lie in* $[s, q_{\mathrm{cw}}]^{\mathrm{cw}}$. *Similarly, in part* (iii) *the points* p_1^* *and* p_2^* *could lie on "the other side" of* s.

Now assume for a contradiction that $d_{\mathrm{cw}}(s, q_{\mathrm{cw}}) + d_{\mathrm{ccw}}(s, q_{\mathrm{ccw}}) > 1$, which means that $q_{\mathrm{ccw}} \in [s, q_{\mathrm{cw}}]^{\mathrm{cw}}$. Since q_{cw} is reached from its parent by a clockwise edge, this implies that $q_{\mathrm{ccw}} \in \mathrm{cov}(\rho_{\mathrm{opt}}, \mathrm{anc}(q_{\mathrm{cw}}))$ by the observation above. Hence, $d_{\mathrm{hop}}(s, q_{\mathrm{cw}}) \geqslant d_{\mathrm{hop}}(s, q_{\mathrm{ccw}})$. An analogous argument shows that $d_{\mathrm{hop}}(s, q_{\mathrm{ccw}}) \geqslant d_{\mathrm{hop}}(s, q_{\mathrm{cw}})$. Hence, $d_{\mathrm{hop}}(s, q_{\mathrm{ccw}}) = d_{\mathrm{hop}}(s, q_{\mathrm{cw}})$. This implies that the edge $(\mathrm{pa}(q_{\mathrm{cw}}), q_{\mathrm{cw}})$ passes over q_{ccw}, otherwise some other edge of $\pi(s, q_{\mathrm{cw}})$ would pass over q_{ccw} and we would have $d_{\mathrm{hop}}(s, q_{\mathrm{ccw}}) < d_{\mathrm{hop}}(s, q_{\mathrm{cw}})$. But then we also have a shortest path from s to q_{ccw} whose last edge is a clockwise edge, contradicting the definition of $\mathcal{B}$. ∎

So we can assume that $d_{\mathrm{cw}}(s, q_{\mathrm{cw}}) + d_{\mathrm{ccw}}(s, q_{\mathrm{ccw}}) < 1$ or, in other words, that q_{ccw} lies clockwise from q_{cw}, as seen from s. Clearly no point from P lies in $(q_{\mathrm{cw}}, q_{\mathrm{ccw}})^{\mathrm{cw}}$. If we have $(q_{\mathrm{cw}}, q_{\mathrm{ccw}})^{\mathrm{cw}} \not\subset \mathrm{cov}(\rho_{\mathrm{opt}}, P)$, then we are done, so assume for a contradiction that $(q_{\mathrm{cw}}, q_{\mathrm{ccw}})^{\mathrm{cw}} \subset \mathrm{cov}(\rho_{\mathrm{opt}}, P)$. This can happen in three ways, each of which will lead to a contradiction.

Case I*: There exists a point* $p^* \in \mathcal{B}$ *such that* $q_{\mathrm{cw}} \in \mathrm{cov}_{\mathrm{ccw}}(\rho_{\mathrm{opt}}, \{p^*\})$.

See Figure 6(ii) for an illustration of the situation. If $p^* = s$, then $d_{\mathrm{hop}}(s, q_{\mathrm{cw}}) = 1$. Since $q_{\mathrm{cw}} \in S_{\mathrm{cw}}$ this means that q_{cw} must also have an incoming clockwise edge from s. But then $\rho_{\mathrm{opt}}(s) \geqslant \frac{1}{2}$, which contradicts Lemma 4.3. So $p^* \neq s$. Now note that p^* must have an outgoing clockwise edge in $\mathcal{B}$, else we can reduce the range of p^* to

$d_{\mathrm{ccw}}(p^*, q_{\mathrm{ccw}})$, which is smaller than $d_{\mathrm{ccw}}(p^*, q_{\mathrm{cw}})$, and still get a feasible solution. Observe that $p^* \notin \pi(s, q_{\mathrm{cw}})$, otherwise we must have $p^* = \mathrm{pa}(q_{\mathrm{cw}})$ (since q_{cw} lies in the range of p^*) which contradicts that $q_{\mathrm{cw}} \in S_{\mathrm{cw}}$. So for any point from P in the region $[s, q_{\mathrm{cw}}]^{\mathrm{cw}}$ there exists a path from s in the communication graph induced by ρ_{opt} that does not use p^*. We now have two subcases.

If $p^* \in [s, q_{\mathrm{ccw}}]^{\mathrm{ccw}}$, then clearly $p^* \in S_{\mathrm{ccw}}$ (otherwise the definition of q_{cw} is contradicted). Hence, each point from P in the region $[s, p^*]^{\mathrm{ccw}}$ has a path from s that does not use p^*. This implies that can reduce the range of p^* to $d_{\mathrm{ccw}}(p^*, q_{\mathrm{ccw}})$ and still get a feasible solution.

If $p^* \in [s, q_{\mathrm{cw}}]^{\mathrm{cw}}$, then obviously we can also reduce the range of p^* to $d_{\mathrm{ccw}}(p^*, q_{\mathrm{ccw}})$ and still get a feasible solution.

So both subcases lead to the desired contradiction.

Case II*: There exists a point* $p^* \in \mathcal{B}$ *such that* $q_{\mathrm{ccw}} \in \mathrm{cov}_{\mathrm{cw}}(\rho_{\mathrm{opt}}, \{p^*\})$.

In the proof of Case I we never used that $\mathcal{B}$ maximizes the number of clockwise edges. Hence, a symmetric argument shows that Case II also leads to a contradiction.

Case III*: There are two points* $p_1^*, p_2^* \in P$ *such that* $[q_{\mathrm{cw}}, q_{\mathrm{ccw}}]^{\mathrm{cw}} \subseteq \mathrm{cov}_{\mathrm{ccw}}(\rho_{\mathrm{opt}}, \{p_1^*\}) \cup \mathrm{cov}_{\mathrm{cw}}(\rho_{\mathrm{opt}}, \{p_2^*\})$.

See Figure 6(iii) for an illustration. We can assume that $q_{\mathrm{cw}} \notin \mathrm{cov}_{\mathrm{ccw}}(\rho_{\mathrm{opt}}, \{p_1^*\})$ and $q_{\mathrm{ccw}} \notin \mathrm{cov}_{\mathrm{cw}}(\rho_{\mathrm{opt}}, \{p_2^*\})$, otherwise we are in Case I or Case II. Now either $p_2^* \notin \pi(s, p_1^*)$ or $p_1^* \notin \pi(s, p_2^*)$ or both. Without loss of generality, assume $p_2^* \notin \pi(s, p_1^*)$. Then $p_2^* \neq s$ and all points from P in the region $[s, q_{\mathrm{ccw}}]^{\mathrm{ccw}}$ have a path from s in the communication graph $\mathcal{G}_{\rho_{\mathrm{opt}}}(P)$ that does not use p_2^*. The point p_2^* must have an outgoing counterclockwise edge, else we can reduce the range of p_2^* to $d_{\mathrm{cw}}(p_2^*, q_{\mathrm{cw}})$ and still get a feasible solution. We have two subcases.

If $p_2^* \in [s, q_{\mathrm{ccw}}]^{\mathrm{ccw}}$, then by reducing the range of p_2^* to $d_{\mathrm{cw}}(p_2^*, q_{\mathrm{cw}})$ we still get a feasible solution.

If $p_2^* \in [s, q_{\mathrm{cw}}]^{\mathrm{cw}}$, then p_2^* must be reached by a clockwise edge from its parent in $\mathcal{B}$, otherwise the definition of q_{ccw} would be contradicted. Hence, for each point from P in the region $[s, p_2^*]^{\mathrm{cw}}$ there is a path from s that does not use p_2^*. So again we can reduce the range of p_2^* to $d_{\mathrm{cw}}(p_2^*, q_{\mathrm{cw}})$ and we still get a feasible solution.

Thus both subcases lead to a contradiction.

This finishes the proof of the lemma. □

4.2. Nonexistence of a SAS in $\mathbb{S}^1$. We have seen that an optimal solution for a set P in $\mathbb{S}^1$ can be obtained from an optimal solution in $\mathbb{R}^1$, when we cut $\mathbb{S}^1$ at an appropriate point r. However, the insertion of a new point into P may cause the location of the cutting point r to change drastically. Next we show that this means that the dynamic problem in $\mathbb{S}^1$ does not admit a SAS.

THEOREM 4.4. *The dynamic broadcast range-assignment problem in* $\mathbb{S}^1$ *with distance power gradient* $\alpha > 1$ *does not admit a SAS. In particular, there is a constant* $c_\alpha > 1$ *such that the following holds: for any* n *large enough, there is a set* $P := \{s, p_1, \ldots, p_{2n+1}\}$ *and a point* q *in* $\mathbb{S}^1$ *such that any update algorithm* ALG *that maintains a* c_α*-approximation must modify more than* $\frac{2n}{3} - 1$ *ranges upon the insertion of* q *into* P.

The rest of this section is dedicated to proving Theorem 4.4. We will prove the theorem for

$$c_\alpha := \min\left(1 + 2^{\alpha-4} - \frac{1}{8},\ 1 + \frac{2^{\alpha-1} - 1}{3 \cdot 2^\alpha + 2},\ 1 + \frac{\min\left(2^\alpha - 1, \frac{3^\alpha - 2^\alpha - 1}{2}, \frac{4^\alpha - 2^\alpha - 2}{3}\right)}{4(2^\alpha + 1)}\right).$$

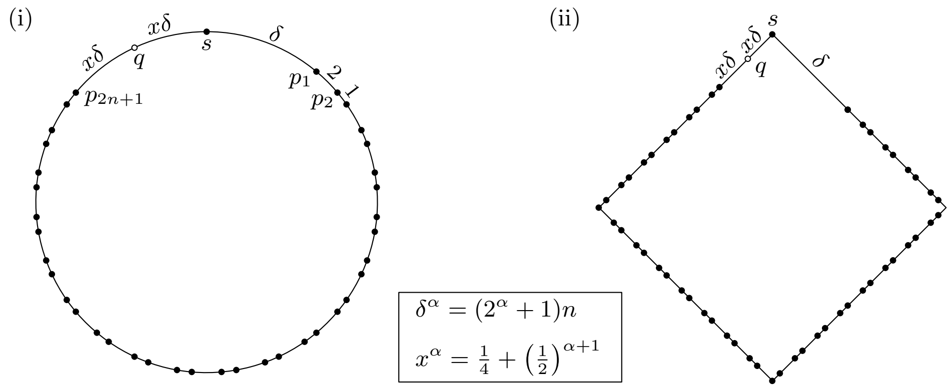

FIG. 7. (i) *The instance showing that there is no SAS in* $\mathbb{S}^1$. (ii) *The instance in* $\mathbb{R}^2$.

Note that each term is a constant strictly greater than 1 for any fixed constant $\alpha > 1$. In particular, for $\alpha = 2$ we have $c_\alpha = 1 + \frac{1}{14}$.

Let $P := \{s, p_1, \ldots, p_{2n+1}\}$, where $d_{\mathrm{cw}}(p_i, p_{i+1}) = 2$ for odd i and $d_{\mathrm{cw}}(p_i, p_{i+1}) = 1$ for even i; see Figure 7(i). Let $d_{\mathrm{cw}}(s, p_1) = \delta$, where $\delta^\alpha = (2^\alpha + 1)n$. Finally, let $d_{\mathrm{cw}}(p_{2n+1}, q) = d_{\mathrm{cw}}(q, s) = x\delta$, where $x^\alpha = \frac{1}{4} + \left(\frac{1}{2}\right)^{\alpha+1}$. Note that $(1/2)^\alpha < x^\alpha < 1/2$ for any $\alpha > 1$.

Let $\rho(p)$ denote the range given to a point p by the update algorithm ALG. Recall that a directed edge (p, p') in the communication graph induced by ρ is called a clockwise edge if $\rho(p) \geqslant d_{\mathrm{cw}}(p, p')$, and it is called a counterclockwise edge if $\rho(p) \geqslant d_{\mathrm{ccw}}(p, p')$. Observe that we may assume that no edge (p, p') is both clockwise and counterclockwise, because otherwise $\rho(p) \geqslant (\delta + 3n + 2x\delta)/2$, which is much too expensive for an approximation ratio of at most c_α. Define the range $\rho(p)$ of a point in P to be CW-*minimal* if $\rho(p)$ equals the distance from p to its clockwise neighbor in P. Similarly, $\rho(p)$ is CCW-*minimal* if $\rho(p)$ equals the distance from p to its counterclockwise neighbor. The idea of the proof is to show that before the insertion of q, most of the points $s, p_1, \ldots, p_{2n+1}$ must have a CW-minimal range, while after the insertion most points must have a CCW-minimal range. This will imply that many ranges must be modified from being CW-minimal to being CCW-minimal.

Before the insertion of q, giving every point a CW-minimal range leads to a feasible assignment of total cost $\delta^\alpha + (2^\alpha + 1)n = 2\delta^\alpha$. After the insertion of q, giving every point a CCW-minimal range leads to a feasible assignment of total cost $2(x\delta)^\alpha + (2^\alpha + 1)n = (2x^\alpha + 1)\delta^\alpha$. Hence, if OPT$(\cdot)$ denotes the cost of an optimal range assignment, we then have the following.

Observation 4.5. OPT$(P) \leqslant 2\delta^\alpha$ and OPT$(P \cup \{q\}) \leqslant (2x^\alpha + 1)\delta^\alpha < 2\delta^\alpha$.

We first prove a lower bound on the total cost of the points $p_1, \ldots, p_{2n+1}$. Intuitively, only $o(n)$ of those points can be reached from s or q (otherwise the range of s or q would be too expensive) and the cheapest way to reach the remaining points will be to use only CW-minimal or CCW-minimal ranges.

LEMMA 4.6. $\sum_{i=1}^{2n+1} \rho(p_i)^\alpha \geqslant (2^\alpha + 1)n - o(n)$, *both before and after the insertion of* q.

Proof. By Observation 4.5, we have $\rho(p)^\alpha \leqslant c_\alpha \cdot 2\delta^\alpha$ and, hence, $\rho(p) \leqslant (2c_\alpha)^{1/\alpha} \cdot \delta < 3\delta$ for any point p. Consider the interval $I = [y_1, y_2]^{\mathrm{cw}}$, where $d_{\mathrm{cw}}(s, y_1) = 3\delta$ and $d_{\mathrm{ccw}}(q, y_2) = 3\delta$. All the points in $I \cap P$ are at a distance more than 3δ from s or q

and hence $I \cap P \subseteq \mathrm{cov}(\rho_{\mathrm{opt}}, P \setminus \{s, q\})$. Let $p_i \in I \cap P$ be the point whose clockwise distance from s is minimum, and let $p_j \in I \cap P$ be the point whose counterclockwise distance from q is minimum. Then the cost of covering all the points in $I \cap P$ using the points in $P \setminus \{s, q\}$ is at least $\sum_{t=i}^{j-1} d_{\mathrm{cw}}(p_t, p_{t+1})^\alpha - 2^\alpha$, where the term -2^α is because the covered region may leave one interval $[p_t, p_{t+1}]^{\mathrm{cw}}$ uncovered. Recall that the cost of assigning all the points in $P \setminus \{s, q\}$ a CW-minimal range is $(2^\alpha + 1)n$. Note that $i = O(\delta)$ since $d_{\mathrm{cw}}(s, p_i) \leqslant 3\delta + 2$ and $(2n+1) - j = O(\delta)$ since $d_{\mathrm{cw}}(p_j, q) \leqslant 3\delta + 2$. Hence,

$$\sum_{i=1}^{2n+1} \rho(p_i)^\alpha \geqslant (2^\alpha + 1)n - O(\delta) \cdot 2^\alpha \geqslant (2^\alpha + 1)n - o(n),$$

since $\delta = ((2^\alpha + 1)n)^{1/\alpha} = o(n)$. ∎

The following lemma gives a key property of the construction.

Lemma 4.7. *The point p_{2n+1} cannot have an incoming counterclockwise edge before q is inserted, and the point p_1 cannot have an incoming clockwise edge after q has been inserted.*

Proof. Suppose before insertion of q the point p_{2n+1} has an incoming counterclockwise edge. The cheapest incoming counterclockwise edge would be from s and this is already too expensive. Indeed, if $\rho(s) \geqslant 2x\delta$, then by Lemma 4.6 the total cost of the range assignment by ALG is at least

$$\begin{aligned}
(2x\delta)^\alpha + (2^\alpha + 1)n - o(n) &= \left(2^\alpha \cdot \left(\frac{1}{4} + \left(\frac{1}{2}\right)^{\alpha+1}\right) + 1\right) \cdot \delta^\alpha - o(n) \\
&= \left(1 + \left(2^{\alpha-3} - \frac{1}{4}\right)\right) \cdot 2\delta^\alpha - o(n) \\
&\geqslant \left(1 + \frac{1}{2} \cdot \left(2^{\alpha-3} - \frac{1}{4}\right)\right) \cdot 2\delta^\alpha \quad \text{for } n \text{ sufficiently large} \\
&\geqslant c_\alpha \cdot \mathrm{OPT}(P) \quad \text{by the definition of } c_\alpha \text{ and Observation 4.5.}
\end{aligned}$$

This contradicts the approximation ratio of ALG, proving the first part of the lemma.

Now suppose after the insertion of q the point p_1 has an incoming clockwise edge. The cheapest way to achieve this is with $\rho(s) = \delta$, which is too expensive. Indeed, by Lemma 4.6 the total cost of the range assignment is then at least

$$\begin{aligned}
&\delta^\alpha + (2^\alpha + 1)n - o(n) \\
&\quad = \frac{2\delta^\alpha}{(2x^\alpha + 1)\delta^\alpha} \cdot (2x^\alpha + 1)\delta^\alpha - o(n) \\
&\quad \geqslant \left(1 + \frac{1}{2} \cdot \left(\frac{2\delta^\alpha}{(2x^\alpha + 1)\delta^\alpha} - 1\right)\right) \cdot \mathrm{OPT}(P \cup \{q\}) \quad \text{for } n \text{ sufficiently large} \\
&\quad = \left(1 + \frac{2 - (2x^\alpha + 1)}{2(2x^\alpha + 1)}\right) \cdot \mathrm{OPT}(P \cup \{q\}) \\
&\quad = \left(1 + \frac{1 - \left(\frac{1}{2} + \frac{1}{2^\alpha}\right)}{2\left(\frac{1}{2} + \frac{1}{2^\alpha} + 1\right)}\right) \cdot \mathrm{OPT}(P \cup \{q\}) \quad \text{since } 2x^\alpha = \tfrac{1}{2} + \tfrac{1}{2^\alpha} \\
&\quad = \left(1 + \frac{2^{\alpha-1} - 1}{3 \cdot 2^\alpha + 2}\right) \cdot \mathrm{OPT}(P \cup \{q\}) \\
&\quad \geqslant c_\alpha \cdot \mathrm{OPT}(P \cup \{q\}) \quad \text{by the definition of } c_\alpha \text{ and Observation 4.5.}
\end{aligned}$$

This contradicts the approximation ratio of ALG, proving the second part of the lemma. ∎

We are now ready to prove that many edges must change from being CW-minimal to being CCW-minimal when q is inserted. Observe that before and after the insertion of a point q, the distance between any two points is either 1, 2, or at least 3. Hence, in the following lemma we may assume that $\rho(p) \in \{0,1,2\} \cup [3,\infty)$ for any point $p \in P \cup \{q\}$.

LEMMA 4.8. *Before the insertion of q, at least $4n/3+1$ of the points from the set $\{s, p_1, \ldots, p_{2n}\}$ have a CW-minimal range. After the insertion of q, at least $4n/3+1$ of the points from the set $\{q, p_1, \ldots, p_{2n}\}$ have a CCW-minimal range.*

Proof. We prove the lemma for the situation before q is inserted; the proof for the situation after the insertion of q is similar. It will be convenient to define $p_0 := s$ (although we may still use s if we want to stress that we are talking about the source). Recall that p_{2n+1} does not have an incoming counterclockwise edge in the communication graph $\mathcal{G}_\rho(P)$ before the insertion of q. Let π^* be a minimum-hop path from s to p_{2n+1} in $\mathcal{G}_\rho(P)$. Since p_{2n+1} does not have an incoming counterclockwise edge and π^* is a minimum-hop path, all edges in π are clockwise. We assign each point p_j with $1 \leqslant j \leqslant 2n+1$ to the edge (p_i, p_t) in π^* such that $i+1 \leqslant j \leqslant t$, and we define $A(p_i,p_t) := \{p_{i+1}, \ldots, p_t\}$ to be the set of all points assigned to (p_i,p_t). We define the *excess* of a point $p_j \in A(p_i,p_t)$ to be

$$\mathrm{excess}(p_j) := \frac{1}{|A(p_i,p_t)|} \cdot \left(\rho(p_i)^\alpha - \sum_{p_\ell \in A(p_i,p_t)} d(p_{\ell-1}, p_\ell)^\alpha \right).$$

We say that an edge (p_i,p_t) in π^* is CW-minimal if p_i has a CW-minimal range. Note that if a point p_j is assigned to a CW-minimal edge, then this is the edge (p_{j-1}, p_j) and $\mathrm{excess}(p_j) = 0$. Intuitively, $\mathrm{excess}(p_j)$ denotes the additional cost we pay for reaching p_j compared to reaching it by a CW-minimal edge, if we distribute the additional cost of a non-CW-minimal edge over the points assigned to it. Because each of the points $p_1, \ldots, p_{2n+1}$ is assigned to exactly one edge on the path π^*, we have

$$\sum_{p_i \in \pi^*} \rho(p)^\alpha \geqslant \sum_{j=1}^{2n+1} d(p_{j-1},p_j)^\alpha + \sum_{j=1}^{2n+1} \mathrm{excess}(p_j) \geqslant \mathrm{OPT}(P) + \sum_{j=1}^{2n+1} \mathrm{excess}(p_j), \tag{4.1}$$

where the second inequality follows from Observation 4.5 and because $p_0 = s$. The following claim essentially states that the smallest possible excess is obtained when $|A(p_i,p_t)| \in \{1,2,3\}$. (The three terms in the claim correspond to these cases.) The reader may want to skip the proof of the claim on first reading, to avoid losing track of the overall proof.

Claim. If p_j is not assigned to a CW-minimal edge, then $\mathrm{excess}(p_j) \geqslant c'_\alpha$, where $c'_\alpha = \min\left(2^\alpha - 1, \frac{3^\alpha - 2^\alpha - 1}{2}, \frac{4^\alpha - 2^\alpha - 2}{3}\right)$.

Proof. Consider a non-CW-minimal edge (p_i,p_t). First suppose only a single point p_j is assigned to (p_i,p_t). Then $t = i+1$ and $p_j = p_t$. Hence, $\rho(p_i) \geqslant d(p_{j-1},p_j)+1$ because we assumed $\rho(p_i) \in \{0,1,2\} \cup [3,\infty)$. Thus when $|A(p_i,p_t)| = 1$ then

$$\mathrm{excess}(p_j) \geqslant (d(p_{j-1},p_j)+1)^\alpha - d(p_{j-1}p_j)^\alpha \geqslant 2^\alpha - 1 \geqslant c'_\alpha.$$

Now suppose $|A(p_i,p_t)| > 1$. Let z_1 be the number of points $p_j \in A(p_i,p_t)$ with $d(p_{j-1},p_j) = 1$, and let z_2 be the number of points $p_j \in A(p_i,p_t)$ with $d(p_{j-1},p_j) = 2$.

Since $|A(p_i,p_t)|>1$ we have $z_1 \geqslant 1$ and $z_2 \geqslant 1$ and $|z_1 - z_2| \leqslant 1$. When $|A(p_i,p_t)|=2$ then $z_1 = z_2 = 1$, and we are distributing the cost of an edge of length at least 3, minus the costs of edges of length 2 and 1, over two points. Thus in this case we have

$$\text{excess}(p_j) \geqslant \frac{3^\alpha - 2^\alpha - 1}{2}.$$

Similarly, when $|A(p_i,p_t)|=3$ then $z_1=2$ and $z_2=1$ (or vice versa, but that will only lead to a larger excess), and we have

$$\text{excess}(p_j) \geqslant \frac{4^\alpha - 2^\alpha - 2}{3}.$$

It remains to argue that we do not get a smaller excess when $|A(p_i,p_t)| \geqslant 4$. To see this, we compare the excess we get when (p_i,p_t) is an edge of π with the excesses we would get when, instead of (p_i,p_t), the edges (p_i,p_{i+2}) and (p_{i+2},p_t) would be in π^*. Note that

$$d(p_i,p_t)^\alpha = \Big(d(p_i,p_{i+2}) + d(p_{i+2},p_t)\Big)^\alpha > d(p_i,p_{i+2})^\alpha + d(p_{i+2},p_t)^\alpha$$

since $\alpha > 1$. Hence,

$$\begin{aligned}
&\frac{d(p_i,p_t)^\alpha - \sum_{\ell=i+1}^{t} d(p_{\ell-1},p_\ell)^\alpha}{t-i}\\
&\quad > \frac{\Big(d(p_i,p_{i+2})^\alpha - \sum_{\ell=i+1}^{i+2} d(p_{\ell-1},p_\ell)^\alpha\Big) + \Big(d(p_{i+2},p_t)^\alpha - \sum_{\ell=i+3}^{t} d(p_{\ell-1},p_\ell)^\alpha\Big)}{t-i}\\
&\quad \geqslant \frac{d(p_i,p_{i+2})^\alpha - \sum_{\ell=i+1}^{i+2} d(p_{\ell-1},p_\ell)^\alpha}{2} + \frac{d(p_{i+2},p_t)^\alpha - \sum_{\ell=i+3}^{t} d(p_{\ell-1},p_\ell)^\alpha}{t-i-2},
\end{aligned}$$

where the last step uses that $\frac{a_1+a_2}{b_1+b_2} \geqslant \min(\frac{a_1}{b_1},\frac{a_2}{b_2})$ for any $a_1,a_2,b_1,b_2>0$. Thus the excess we get for (p_i,p_t) is at least the minimum of the excesses we would get for (p_i,p_{i+2}) and (p_{i+3},p_t). More generally, when $|A(p_i,p_t)|>4$ then we can compare the excess for (p_i,p_t) with the excesses we get when we would replace (p_i,p_t) with a path of smaller edges, each being assigned two or three points. The excess for (p_i,p_{i+2}) is at least the minimum of the excesses for these shorter edges. (Reducing to edges that are assigned a single point is not useful, since these may be CW-minimal and have zero excess.) This finishes the proof of the claim. ∎

Now suppose for a contradiction that fewer than $4n/3+1$ points from $\{s,p_1,\dots,p_{2n+1}\}$ have a CW-minimal range. Then at least $2n/3+1$ points p_j have $\text{excess}(p_j) \geqslant c'_\alpha$ by the claim above. By Inequality (4.1) the total cost incurred by ALG is therefore more than

$$\text{OPT}(P) + c'_\alpha \cdot (2n/3) = \text{OPT}(P) + \frac{c'_\alpha}{3(2^\alpha+1)} \cdot 2(2^\alpha+1)n \tag{4.2}$$

$$> \left(1 + \frac{\min\left(2^\alpha - 1, \frac{3^\alpha-2^\alpha-1}{2}, \frac{4^\alpha-2^\alpha-2}{3}\right)}{4(2^\alpha+1)}\right) \cdot \text{OPT}(P) \tag{4.3}$$

$$\geqslant c_\alpha \cdot \text{OPT}(P) \tag{4.4}$$

which contradicts the approximation ratio achieved by ALG. ∎

Lemma 4.8 implies that at least $4n/3$ of the points $p_1,\dots,p_{2n+1}$ have a CW-minimal range before q is inserted, and at least $4n/3$ of those points have a CCW-minimal range after the insertion. Hence, at least $2n+1-2\cdot(2n/3+1) = 2n/3-1$

points must change from being CW-minimal to being CCW-minimal, thus finishing the proof of Theorem 4.4.

5. The 2-dimensional problem. The broadcast range-assignment problem is NP-hard in $\mathbb{R}^2$, so we cannot expect a characterization of the structure of an optimal solution similar to Theorem 2.1. By extending our lower-bound construction for $\mathbb{S}^1$ to $\mathbb{R}^2$, we show in section 5.1 that the problem in $\mathbb{R}^2$ does not admit a SAS. Also, in section 5.2 we give a relatively simple $O(1)$-stable $O(1)$-approximation algorithm for $\alpha \geqslant 2$.

5.1. Non-existence of SAS in $\mathbb{R}^2$.

THEOREM 5.1. *The dynamic broadcast range-assignment problem in $\mathbb{R}^2$ with distance power gradient $\alpha > 1$ does not admit a SAS. In particular, there is a constant $c_\alpha > 1$ such that the following holds: for any n large enough, there is a set $P := \{s, p_1, \dots, p_{2n+1}\}$ and a point q in $\mathbb{R}^2$ such that any update algorithm* ALG *that maintains a c_α-approximation must modify at least $2n/3-1$ ranges upon the insertion of q into P.*

Proof. We use the same construction as in $\mathbb{S}^1$, where we embed the points on a square and the distances used to define the instance are measured along the square; see Figure 7(ii). We now discuss the changes needed in the proof to deal with the fact that distances in $\mathbb{R}^2$ between points from $P \cup \{q\}$ may be smaller than when measured along the square. With a slight abuse of terminology, we will still refer to an edge (p, p') that was clockwise in $\mathbb{S}^1$ as a clockwise edge, and similarly for counterclockwise edges.

Note that Observation 4.5 still holds. Now consider Lemma 4.6. The proof uses the fact that the points p_i at a distance more than 3δ from s or q must be covered by the ranges of the points $p_1, \dots, p_{2n+1}$. We now restrict our attention to the points that are also at distance more than 3δ from a corner of the square. Each such point p_i must be covered by the range of some point p_j on the same edge of the square. Hence, the distance in $\mathbb{R}^2$ from p_j to p_i is the same as the distance in $\mathbb{S}^1$, so we can use the same reasoning as before. Thus the exclusion of the points that are at a distance at most 3δ from a corner of the square only influences the constant in the $o(n)$ term in the lemma. Hence, Lemma 4.6 still holds.

The proof of Lemma 4.7 still holds, since the cheapest counterclockwise edge to p_{2n+1} before the insertion of q is still from s (and the distance from s to p_{2n+1} did not change), and the cheapest clockwise edge to p_1 after the insertion of q is still from s (and the distance from s to p_1 did not change).

It remains to check Lemma 4.8. The proof still holds, except that the claim that $\text{excess}(p_j) \geqslant c'_\alpha$ may not be true for the given value of c'_α when p_j is near a corner of the square, because the distances between points on different edges of the square do not correspond to the distances in $\mathbb{S}^1$. To deal with this, we simply ignore the excess of any point within distance 3δ from a corner. This reduces the total excess by $o(n)$. It is easily verified that this does not invalidate the rest of the proof: we have to subtract $o(n)$ from the formulas in equality (4.2), but this is still larger than $c_\alpha \cdot \text{OPT}(P)$.

We conclude that all lemmas still hold, which proves Theorem 5.1. □

5.2. An $O(1)$-stable $O(1)$-approximation algorithm in $\mathbb{R}^2$. We describe an $O(1)$-stable $O(1)$-approximation algorithm for $\alpha \geqslant 2$ in $\mathbb{R}^2$. The algorithm is based on a result by Ambühl [2], who showed that a minimum spanning tree (MST) on P

gives a 6-approximation for the static broadcast range-assignment problem: turn the MST into a directed tree rooted at the source s, and assign as a range to each point $p \in P$ the maximum length of any of its outgoing edges. The key lemma underlying the result is the following.

LEMMA 5.2 ([2]). *Let P be any point set in $\mathbb{R}^2$. Let T_P be an MST on P, and let $E(T_P)$ be the set of edges of T_P. Then, for any distance-power gradient $\alpha \geqslant 2$, we have $\sum_{e \in T_P} |e|^\alpha \leqslant 6 \cdot \textsc{opt}$, where $\textsc{opt} = \mathrm{cost}_\alpha(\rho_{\mathrm{opt}}(P))$ is the cost of an optimal range assignment.*

In the static problem this immediately gives a 6-approximation algorithm: turn the MST into a directed tree rooted at the source s, and assign as a range to each point $p \in P$ the maximum length of any of its outgoing edges. To apply this in the dynamic setting, we need the following lemma, which implies that for any point set P and any additional point q, any MST T on P can be converted to an MST T' on $P \cup \{q\}$ that is very similar to T. The result is folklore [39].

LEMMA 5.3. *Let P be a set of points in a metric space X, and let $P' := P \cup \{q\}$ for some point $q \in X$. For any MST T on P, there exists an MST T' on P' that only contains edges of T and edges incident to q. Similarly, for any MST T' on P' there exists an MST T on P such that T' only contains edges of T and edges incident to q.*

We use this lemma in combination with the following well-known lemma.

LEMMA 5.4. *Let T be an MST of a point set in $\mathbb{R}^d$. Then the maximum vertex degree of T is bounded by the Hadwiger number of the corresponding unit ball in $\mathbb{R}^d$. In particular, the maximum vertex degree of an MST in $\mathbb{R}^2$ is at most 6.*

We can now prove the following theorem.

THEOREM 5.5. *There is a 17-stable 12-approximation algorithm for the dynamic broadcast range-assignment problem in $\mathbb{R}^2$, for any fixed power-distance gradient $\alpha \geqslant 2$. The algorithm can update the range assignment upon an insertion or deletion in $O(n\alpha(n))$ time, where n is the number of points in the current point set and $\alpha(n)$ is the inverse Ackermann function.*

Proof. Our algorithm will maintain an MST T on the current point set P, using Lemma 5.3. We set the range of each point to be the maximum length of any of its incident edges. Clearly, this defines a feasible solution. We denote the resulting range assignment by ρ_{mst}.

We now analyze the stability of ρ_{mst}. Consider the insertion of a point q. Let T be the MST before the insertion of q, and let T' be the MST T' after the insertion has been handled (and with the properties stated in Lemma 5.3). Observe that, apart from the point q itself, only the ranges of the neighbors of q in T' can increase. By Lemma 5.4 we have $\deg(q) \leqslant 6$, where $\deg(q)$ denotes the degree of q. Hence, the number of ranges that need to be increased is at most 7. Also observe that only the ranges of those points can decrease that had an edge belonging to the edge set $T \setminus T'$ incident to it. Since $\deg(q) \leqslant 6$, and T and T' have $|P|-1$ and $|P|$ edges, respectively, we have $|T \setminus T'| \leqslant 5$. Hence the ranges of at most ten points can decrease. It follows that the algorithm is 17-stable; even more, the algorithm is $(7,10)$ stable when only insertions are present, and $(10,7)$ stable when only deletions are allowed.

To analyze the approximation ratio, we use Lemma 5.2 and note that every edge in T is adjacent to at most two vertices. Hence, $\mathrm{cost}_\alpha(\rho_{\mathrm{mst}}(P)) \leqslant 2 \cdot \sum_{e \in T} |e|^\alpha \leqslant 12 \cdot \textsc{opt}$ for any set P.

It remains to argue that updating the range assignment can be done in $O(n\alpha(n))$ time. Assume for the moment that the MST, both before and after the update, is unique. It is well known that the MST of a planar point set is a subset of a Delaunay triangulation of that point set, and that a Delaunay triangulation has $O(n)$ edges [21]. Thus, if we maintain the Delaunay triangulation of P, then after each update we only need to compute the MST from the updated Delaunay triangulation. Maintaining a Delaunay triangulation can be done in $O(n)$ time per update [1] and computing an MST of a graph with $O(n)$ edges can be done in $O(n\alpha(n))$ time [11]. Hence, our update algorithm runs in $O(n\alpha(n))$ time in total.

So far we assumed that the MST is unique. In general this need not be the case, since edges may have exactly the same length. The Delaunay triangulation of a planar point set may not be unique either, when the point set contains four cocircular points. Lemma 5.3 guarantees that there still exists an MST for the updated point set that is sufficiently similar to the previous MST. An easy way to deal with these degeneracies is using *symbolic perturbation* [23]: symbolically perturb the points so that the degeneracies disappear and the Delaunay triangulation and the MST are unique. Note that in our application we need to ensure that the perturbations before and after an update are consistent—for instance, if a perturbed edge e is shorter than a perturbed edge e' before an insertion, then this should also be true after the perturbation. Fortunately this is easy to achieve, since the perturbation of a point is uniquely determined by its index (that is, its unique identifier). ∎

6. 1-Stable, 2-Stable, and 3-Stable Algorithms in $\mathbb{R}^1$. In section 3 we have presented a $(2k+6)$-stable algorithm with approximation ratio $1+2^\alpha/k^{\alpha-1}$, which provides us with a SAS. For small k the algorithm is not very good: the most stable algorithm we can get is 6-stable, by setting $k=0$. A careful analysis shows that the approximation ratio of this 6-stable algorithm is 3, for $\alpha=2$. In this section, we investigate the approximation ratios we can get for instances in $\mathbb{R}^1$ using a very small stability parameter. We give a 1-stable $O(1)$-approximation algorithm; obviously, this is the best we can do in terms of stability. This algorithm can only handle insertions. We also show that this is necessarily the case: a 1-stable algorithm that can handle insertions as well as deletions cannot have a bounded approximation ratio. We then present a straightforward 2-stable 2-approximation algorithm, which simply gives every point its standard range. Finally, we study 3-stable algorithms: we show that using a 3-stable algorithm it is possible to get an approximation ratio strictly below 2. See Table 1 for an overview of results.

6.1. A 1-stable insertion-only algorithm. We first describe our algorithm for the *one-sided* version of the problem, where all points in P lie to the same side of the source. Let $P=\{s,p_1,\dots,p_n\}$, where the points are numbered in order of increasing distance to the source. It will be convenient to define $p_0 := s$. Our algorithm maintains a range assignment ρ that satisfies the following invariant.

TABLE 1
An overview of the approximation ratio of 1-stable, 2-stable, and 3-stable algorithms.

ℓ-stable algorithm	Approximation ratio	Remarks
$\ell=1$	$6+2\sqrt{5}\approx 10.47$	$\alpha=2$, insertions only
$\ell=2$	2	for any $\alpha>1$
$\ell=3$	1.97	$\alpha=2$

Algorithm 6.1. 1-Stable-Insert(P,q).

$\triangleright$ By default $\rho(q)=0$, so we only set $\rho(q)$ when it receives a nonzero range.
if q is extreme **then**
  Set $\rho(\mathrm{pred}(q)) := |\,\mathrm{pred}(q)q|$, thus creating a new block.
else
  Let $B[i,j]$ be the block containing q, after the insertion of q.
  if $B[i,j]$ consists of at most four points **then**
    Do nothing.
  else if $B[i,j]$ consists of five points **then**
    Set $\rho(p_{\mathrm{mid}}) := |p_{\mathrm{mid}}p_j|$, where p_{mid} is the middle point from $B[i,j]$.
  else $\triangleright$ $B[i,j]$ consists of six points
    Let $p_{\mathrm{mid}} \notin \{p_i,p_j\}$ be the point in $B[i,j]$ with nonzero range.
    Split the block $B[i,j]$ by decreasing the range of p_i to $|p_ip_{\mathrm{mid}}|$.

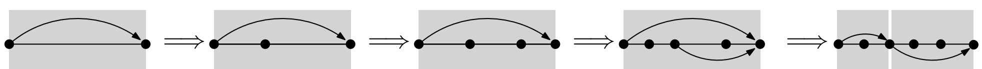

Fig. 8. *Life cycle of a block. At the last step, the block is split into two smaller blocks, which start in the middle of their life cycle: one block consists of three points, the other of four points.*

- There is a path π^* in $\mathcal{G}_\rho(P)$ from p_0 to p_n such that for each edge (p_i,p_j) on the path we have $\rho(p_i)=|p_ip_j|$ and $i<j\leqslant i+4$. For an edge (p_i,p_j) on π^*, we call the subsequence $p_i,\dots,p_j$ a *block,* and we denote it by $B[i,j]$.
- A point p_t in a block $B[i,j]$ is a zero-range point, unless $B[i,j]$ consists of five points (including p_i and p_j) of which p_t is the middle one. In the latter case $\rho(p_t)=|p_tp_j|$.

Algorithm 1-Stable-Insert, presented below, shows how to insert a point q into P. Figure 8 shows the life cycle of a block in the solution maintained by the algorithm.

It is readily verified that 1-Stable-Insert maintains the invariant, implying that the solution remains feasible, and that it is 1-stable. We now analyze its approximation ratio.

Lemma 6.1. *Algorithm* 1-Stable-Insert *maintains a c'_α-approximation of an optimal solution for the one-sided range-assignment problem in $\mathbb{R}^1$, where the approximation ratio c'_α depends on the distance-power gradient α. For $\alpha=2$ the approximation ratio is $c'_2=3+\sqrt{5}$.*

Proof. The unique optimal solution for the one-sided problem on the current point set $P=\{p_0,\dots,p_n\}$ is the chain from p_0 to p_n, which has cost $\sum_{i=0}^{n-1}|p_ip_{i+1}|^\alpha$. This implies that the approximation ratio of the current range assignment ρ is bounded by the maximum, over all blocks $B[i,j]$ in the current assignment, of the quantity $\sum_{t=i}^{j-1}\rho(p_t)^\alpha \;/\; \sum_{t=i}^{j-1}|p_tp_{t+1}|^\alpha$, where the numerator gives the cost incurred by the algorithm on $B[i,j]$ and the denominator gives the cost of the optimal solution on $B[i,j]$.

To analyze this maximum, consider a block $B[i,j]$ and assume without loss of generality that $p_i=0$ and $p_j=1$. Clearly, the maximum approximation ratio that can be achieved is when $B[i,j]$ consists of five points; see the fourth block in Figure 8. Let $p_{(i+j)/2}$, the middle point in $B[i,j]$, be located at position x, for some $0<x<1$. Then the cost of the algorithm incurred on $B[i,j]$ is $1+(1-x)^\alpha$. The cost of the optimal solution on $B[i,j]$ is minimized when the second point in the block is located

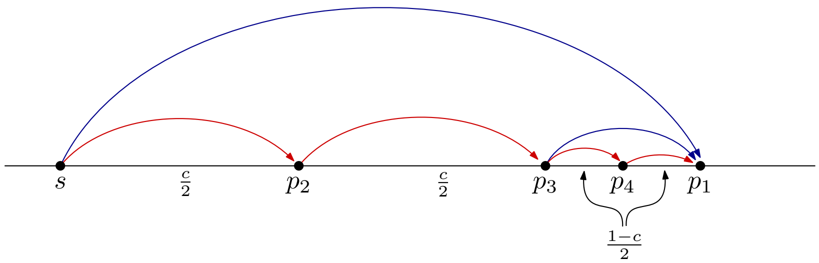

FIG. 9. *The lower-bound construction used in the proof of Observation* 6.2. *The red ranges correspond to the optimal solution, while the blue ranges correspond to the* 1*-stable algorithm. Note: color appears only in the online article.*

at position $x/2$ (this is true because $\alpha > 1$ and so for $0 < y < x$ the function $y^\alpha + (x-y)^\alpha$ is minimized at $y = x/2$) and, similarly, when the fourth point is located at $(x+1)/2$. Hence,

$$\frac{\sum_{t=i}^{j-1} \rho(p_t)^\alpha}{\sum_{t=i}^{j-1} |p_t p_{t+1}|^\alpha} = \frac{1 + (1-x)^\alpha}{2(x/2)^\alpha + 2((1-x)/2)^\alpha} = \frac{2^{\alpha-1} \cdot (1 + (1-x)^\alpha)}{x^\alpha + (1-x)^\alpha}.$$

Thus the approximation ratio is $c'_\alpha = \max_{0 \leqslant x \leqslant 1} \frac{2^{\alpha-1} \cdot (1+(1-x)^\alpha)}{x^\alpha + (1-x)^\alpha}$. For $\alpha = 2$ this is maximized at $x = (3 - \sqrt{5})/2$, giving an approximation ratio $c'_2 = 3 + \sqrt{5}$. ∎

The approximation ratio of Algorithm 1-STABLE-INSERT for the one-sided range assignment problem in $\mathbb{R}^1$ is actually tight for $\alpha = 2$, as the next observation shows.

Observation 6.2. For the one-sided range-assignment problem, the approximation ratio of $3 + \sqrt{5}$ is tight for Algorithm 1-STABLE-INSERT.

Proof. Let $c = (3 - \sqrt{5})/2$, and consider the instance $P = \{s, p_1, p_2, p_3, p_4\}$ with $s = 0$, $p_1 = 1$, $p_2 = \frac{c}{2}$, and $p_3 = c, p_4 = \frac{(1+c)}{2}$; see Figure 9. The insertion order of the points is p_1, p_2, p_3, p_4. Clearly, by setting $\rho(s) = \rho(p_2) = \frac{c}{2}$ and $\rho(p_3) = \rho(p_4) = \frac{1-c}{2}$, an optimum solution with $\text{cost}_2(\rho(P)) = \frac{5-2\sqrt{5}}{2}$ is found. Algorithm 1-STABLE-INSERT sets $\rho(s) := 1$ and $\rho(p_3) := 1 - c$, and all other ranges to 0. The resulting cost equals $\frac{5-\sqrt{5}}{2}$, and the ratio follows. See Figure 9 for an illustration. ∎

To handle the case with points to both sides of s, we proceed as follows. Let $P = L \cup \{s\} \cup R$, where L and R contain the points to the left and to the right of s, respectively. We simply run the above algorithm separately on $L \cup \{s\}$ and $\{s\} \cup R$. This way the points in $R \cup L$ are assigned one range, while s gets assigned two ranges; the actual range of s is the largest of these two ranges.

THEOREM 6.3. *There exists a* 1*-stable* c_α*-approximation algorithm for the broadcast range-assignment problem in* $\mathbb{R}^1$*, where the approximation ratio* c_α *depends on the distance-power gradient* α*. For* $\alpha = 2$ *the approximation ratio is* $c_2 = 2(3 + \sqrt{5}) \approx 10.47$*.*

Proof. Recall that our algorithm simply runs the one-sided algorithm separately on $L \cup \{s\}$ and $\{s\} \cup R$, where the actual range of s is defined to be the largest of the two ranges it receives.

To analyze the approximation ratio of this algorithm we use that for any $\alpha > 1$ we have $\text{OPT}(L \cup \{s\} \cup R) \geqslant \max(\text{OPT}(L \cup \{s\}), \text{OPT}(\{s\} \cup R))$, where $\text{OPT}(\cdot)$ denotes the cost of an optimal range assignment [22]. Hence, the cost of the range assignment ρ that we maintain is

$$\begin{aligned}\mathrm{cost}_\alpha(\rho(L\cup\{s\}\cup R)) &\leqslant \mathrm{cost}_\alpha(\rho(L\cup\{s\})) + \mathrm{cost}_\alpha(\rho(\{s\}\cup R))\\ &\leqslant c'_\alpha\cdot(\textsc{opt}(L\cup\{s\}) + \textsc{opt}(\{s\}\cup R))\\ &\leqslant 2c'_\alpha\cdot\max(\textsc{opt}(L\cup\{s\}),\textsc{opt}(\{s\}\cup R))\\ &\leqslant 2c'_\alpha\cdot\textsc{opt}(L\cup\{s\}\cup R).\end{aligned}$$

Lemma 6.1 thus implies the theorem. ∎

We note that a 1-stable algorithm ALG that handles deletions cannot have a bounded approximation ratio, as we show next for $\alpha=2$. Suppose for a contradiction that ALG has approximation ratio c, where we assume for simplicity that c is an integer. Let $P:=\{s,r_1,\dots,r_{c+1}\}$, where $s=0$ and $r_i=i/(c+1)$. Then OPT $=(c+1)\cdot(1/(c+1))^2=1/(c+1)$, so ALG cannot give the source a range of 1. But if we then delete all nonzero points in $P\setminus\{s\}$, the algorithm is stuck: the deletion of a nonzero point already causes a modification, so the algorithm is not allowed to increase any range; hence, the solution is invalid after all nonzero-range points from $P\setminus\{s\}$ have been deleted. This is a consequence of our choice of the definition of an algorithm's stability, and one may consider alternative definitions of the stability in the broadcast range-assignment problem avoiding this consequence. However, with the application in mind, our current definition (where the algorithm has to pay for both for removing a point with nonzero range, as well as for assigning a nonzero range to a new point) is appropriate. It is then actually interesting to observe that for insertions it is possible to obtain a 1-stable algorithm with $O(1)$-approximation ratio.

A lower bound for 1*-stable insertion-only algorithms.* The next theorem shows that any 1-stable algorithm in $\mathbb{R}^1$ has an approximation ratio greater than 2.61 for $\alpha=2$.

THEOREM 6.4. *Any* 1*-stable algorithm for the insertion-only broadcast range-assignment problem in* $\mathbb{R}^1$ *has an approximation ratio that is greater than or equal to* $\frac{1}{2}\cdot(3+\sqrt{5})\approx 2.61$ *for* $\alpha=2$, *and any* 1*-stable algorithm has an approximation ratio greater than* $\frac{2^\alpha+1}{2}$ *for* $\alpha>2$.

Proof. Let ALG be a 1-stable algorithm, and let ρ_{alg} be the range assignment it maintains.

Consider the point set $P:=\{s,r_1,r_2,p\}$, where $s=0$, and $r_i=x_i$ for $i=1,2$, and $p=1$. Assume $0<x_1<1$ and let $x_2=x_1+\frac{(1-x_1)}{2}$. Also assume after the source, the point p arrives first, then the point r_1, and finally r_2 arrives. Let $P':=\{s,r_1,p\}$. Trivially, after the arrival of the point p, we must have $\rho_{\mathrm{alg}}\geqslant 1$ in order to have a feasible solution. After the arrival of r_1, ALG is forced to keep $\rho_{\mathrm{alg}}(s)\geqslant 1$ since ALG is 1-stable.

We consider two cases.

Case I*: After the arrival of* r_1, ALG *gives a range of at least* $1-x_1$ *to* r_1.

In this case ALG cannot decrease any range. So,

$$\mathrm{cost}_\alpha(\rho_{\mathrm{alg}}(P'))\geqslant 1+(1-x_1)^\alpha.$$

An optimal solution for P' has cost $x_1^\alpha+(1-x_1)^\alpha$, and so the approximation ratio of ALG in Case I is at least $\frac{1+(1-x_1)^\alpha}{x_1^\alpha+(1-x_1)^\alpha}$.

Case II*: After the arrival of* r_1, ALG *gives a range less than* $1-x_1$ *to* r_1.

Now the point r_2 arrives. Since ALG is 1-stable it cannot decrease the range of the source. Hence,

$$\mathrm{cost}_\alpha(\rho_{\mathrm{alg}}(P)) \geqslant 1.$$

An optimal solution for P has cost $x_1^\alpha + 2 \cdot (\frac{1-x_1}{2})^\alpha$, and so the approximation ratio of ALG in Case II is at least $\frac{1}{x_1^\alpha + 2\cdot(\frac{1-x_1}{2})^\alpha}$.

We conclude that the approximation ratio of any 1-stable algorithm is greater than or equal to at least

$$\min\left(\frac{1+(1-x_1)^\alpha}{x_1^\alpha+(1-x_1)^\alpha}, \frac{1}{x_1^\alpha + 2\cdot(\frac{1-x_1}{2})^\alpha}\right).$$

For $\alpha = 2$ we see that by substituting $x_1 = \frac{3}{2} - \frac{\sqrt{5}}{2}$ we get the approximation ratio is at least $\frac{1}{2}\cdot(3+\sqrt{5})$. Moreover, for any $\alpha > 2$ we get an approximation ratio greater than $\frac{2^\alpha+1}{2}$, for instance, by substituting $x_1 = \frac{1}{2}$. ∎

6.2. A 2-stable algorithm. Obtaining a 2-stable 2-approximation algorithm is straightforward: simply give every point in P its standard range, where the source s receives the largest of its (at most) two standard ranges. This induces a broadcast tree consisting of (at most) two chains: a chain from s to the rightmost point and a chain from s to the leftmost point. This algorithm is 2-stable: if we insert an extreme point, then we increase the range of at most one point, and if we insert a nonextreme point q we increase the range of q and decrease the range of its predecessor. (Deletions are symmetrical.) We call this algorithm the *standard-range algorithm.* It is easy show that the standard-range algorithm gives a 2-approximation [22].

Observation 6.5 ([22]). The standard-range algorithm for the dynamic broadcast range-assignment problem in $\mathbb{R}^1$ is 2-stable and gives a 2-approximation, for any power-distance gradient $\alpha > 1$. Moreover, the approximation ratio 2 is tight for this algorithm.

Proof. The fact that the approximation ratio is at most 2 was observed in Theorem 2.2 in [22]. We sketch the key idea for completeness. Divide the point set $P \setminus \{s\}$ into P^+ and P^-. Let the source be at $x = 0$ on the real line. Let P^+ be the set of points whose x-coordinate is positive and P^- be the set of points whose x-coordinate is negative. Then

$$\mathrm{cost}_\alpha(\rho_{\mathrm{opt}}(P)) \geqslant \max(\mathrm{cost}_\alpha(\rho_{\mathrm{opt}}(P^+ \cup \{s\})), \mathrm{cost}_\alpha(\rho_{\mathrm{opt}}(P^- \cup \{s\}))).$$

Moreover, the increase in the cost of ALG after arrival of a point q with positive x-coordinate is at most the increase in the cost of $\mathrm{cost}_\alpha(\rho_{\mathrm{opt}}(P^+ \cup \{s\}))$. Similarly, the increase in the cost of ALG after arrival of a point q with negative x-coordinate is at most the increase in the cost of $\mathrm{cost}_\alpha(\rho_{\mathrm{opt}}(P^- \cup \{s\}))$. Hence,

$$\mathrm{cost}_\alpha(\rho_{\mathrm{alg}}(P)) \leqslant \mathrm{cost}_\alpha(\rho_{\mathrm{opt}}(P^+ \cup \{s\})) + \mathrm{cost}_\alpha(\rho_{\mathrm{opt}}(P^- \cup \{s\})).$$

Since ALG is optimal for P^+ and P^-, we have $\mathrm{cost}_\alpha(\rho_{\mathrm{opt}}(P^+ \cup \{s\})) = \mathrm{cost}_\alpha(\rho_{\mathrm{alg}}(P^+ \cup \{s\}))$ and $\mathrm{cost}_\alpha(\rho_{\mathrm{opt}}(P^- \cup \{s\}) = \mathrm{cost}_\alpha(\rho_{\mathrm{alg}}(P^- \cup \{s\}))$. We can conclude that ALG is a 2-approximation.

It remains to give an instance showing this bound is tight. Define $P := P(\varepsilon) \cup \{s\}$, where $s = 0$ is the source, and $P(\varepsilon) := \{p_1, p_2, p_3, p_4\}$, where $p_1 = \varepsilon$, $p_2 = -\varepsilon$, $p_3 = 1$, and $p_4 = -1$ for some small $\varepsilon > 0$. The insertion order of the points in $P(\varepsilon)$ is p_1, p_2, p_3, p_4. Clearly, by setting $\rho(s) = 1$ and $\rho(p_i) = 0$ for $i = 1, \ldots, 4$, we obtain an optimal solution with $\text{cost}_\alpha(\rho(P)) = 1$. However, the standard-range algorithm will set $\rho(s) = \varepsilon$, $\rho(p_1) = \rho(p_2) = 1 - \varepsilon$, and $\rho(p_3) = \rho(p_4) = 0$, leading to a solution with cost $f(\varepsilon) := 2(1-\varepsilon)^\alpha + \varepsilon^\alpha$. Since $\lim_{\varepsilon \to 0} f(\varepsilon) = 2$, this proves we have tightness for any $\alpha > 1$. ∎

6.3. A 3-stable algorithm with approximation ratio less than 2. Given the simplicity of our 2-stable 2-approximation algorithm, it is surprisingly difficult to obtain an approximation ratio strictly smaller than 2. In fact, we have not been able to do this with a 2-stable algorithm. Below we show this is possible with a 3-stable algorithm, at least for the case $\alpha = 2$, which we assume from now on.

Recall that for any set P with points on both sides of the source point s, there is an optimal range assignment inducing a broadcast tree with a single root-crossing point; see Figure 1. Unfortunately the root-crossing point may change when P is updated. This may cause many changes if we maintain a solution with a good approximation ratio and the same root-crossing point as the optimal solution. We therefore restrict ourselves to *source-based range assignments,* where s is the root-crossing point. The main question is then how large the range of s should be, and which points within range of s should be zero-range points.

We now define our source-based range assignment, which we denote by ρ_{sb}, more precisely. It will be uniquely defined by the set P; it does not depend on the order in which points have been inserted or deleted. Let δ be a parameter with $1/2 < \delta < 1$; later we will choose δ such that the approximation ratio of our algorithm is optimized. We call a point $p \in P \setminus \{s\}$ *expensive* if $\text{succ}(p) \neq \text{NIL}$ and $|p\,\text{succ}(p)| > \delta \cdot |s\,\text{succ}(p)|$, and we call it *cheap* otherwise. The source s is defined to be always expensive. (This is consistent in the sense that for $p = s$ the condition $|p\,\text{succ}(p)| > \delta \cdot |s\,\text{succ}(p)|$ holds for both successors, since $\delta < 1$.) We denote the set of all expensive points in P by P_{exp} and the set of all cheap points by P_{cheap}. Define $d_{\max} := \max\{|s\,\text{succ}(p)| : p \in P_{\text{exp}}\}$, that is, $d_{\max}$ is the maximum distance from s to the successor of any expensive point. We say that a point $p \in P_{\text{exp}}$ is *crucial* if $|s\,\text{succ}(p)| = d_{\max}$. Typically there is a single crucial point, but there can also be two: one on the left and one on the right of s. Our source-based range assignment ρ_{sb} is now defined as follows:

- $\rho_{\text{sb}}(s) := d_{\max}$;
- $\rho_{\text{sb}}(p) := 0$ for all $p \in P_{\text{exp}} \setminus \{s\}$; and
- $\rho_{\text{sb}}(p) := \rho_{\text{st}}(p)$ for all $p \in P_{\text{cheap}}$, where $\rho_{\text{st}}(p)$ denotes the standard range of a point.

It is easily checked that we can maintain this range assignment with a 3-stable algorithm. The challenge is to analyze its approximation ratio; we show that, for a suitable choice of δ, the approximation ratio is strictly smaller than 2.

Now we prove the stability and approximation ratio of our proposed 3-stable algorithm. The lemma below analyzes the stability of ρ_{sb}. The lemma implies that insertions are $(2,1)$-stable and deletions are $(1,2)$-stable.

LEMMA 6.6. *Consider a point set P and a point $q \notin P$. Let $\rho_{\text{old}}(p)$ be the range of a point p in $\rho_{\text{sb}}(P)$ and let $\rho_{\text{new}}(p)$ be the range of p in $\rho_{\text{sb}}(P \cup \{q\})$. Then*

$$|\{p \in P \cup \{q\} : \rho_{\text{old}}(p) < \rho_{\text{new}}(p)\}| \leqslant 2 \text{ and } |\{p \in P \cup \{q\} : \rho_{\text{old}}(p) > \rho_{\text{new}}(p))|\} \leqslant 1.$$

Proof. Due to the insertion of q, five types of range modifications can happen.

(i) The point q may get a nonzero range because it is cheap and nonextreme.

(ii) A point p may move from P_{cheap} to P_{exp} and become a zero-range point. This can only happen when $p = \text{pred}(q)$ and p was extreme before the insertion of q. Hence, q will be extreme after its insertion, so this cannot occur together with type (i).

(iii) A point $p \in P_{\text{cheap}}$ may get a smaller range because its standard range decreases. This can only happen when $p = \text{pred}(q)$, and so it cannot happen together with type (ii).

(iv) A point p may move from P_{exp} to P_{cheap} and get a nonzero range. Again, this can only happen when $p = \text{pred}(q)$, so this cannot happen together with types (ii) or (iii).

(v) The source s may get a different range because $d_{\max}$ changes. If $d_{\max}$ decreases, then $\text{pred}(q)$ must have been crucial, and so this cannot occur together with types (ii) or (iii). If $d_{\max}$ increases, then a type (ii) modification must have occurred, which means that types (i), (iii), and (iv) did not occur.

Overall, we have at most one range increase of type (i), at most one range change from any of the types (ii), (iii), (iv), and at most one change of type (v). There can be at most one decrease among these three changes, because if type (v) is a decrease, then types (ii) and (iii) did not occur. Finally, there can be at most two increases, because if type (v) is an increase, then types (i), (iii), and (iv) did not occur. ∎

From now on we assume without loss of generality that the source s is located at $x = 0$. We will need the following lemma before we can proceed to prove the performance guarantee.

LEMMA 6.7. *Let $I \subset \mathbb{R}^1$ be an interval of length Δ_1 at distance Δ_2 from the source, that is, $I = [\Delta_2, \Delta_2 + \Delta_1]$ or $I = [-\Delta_1 - \Delta_2, -\Delta_2]$ for some $\Delta_2 > 0$. Let $P_{\text{cheap}}(I)$ be the set of all cheap points that are in I and whose successor lies in I as well. Then $\sum_{p \in P_{\text{cheap}}(I)} \rho_{\text{st}}(p)^2 \leqslant \delta(\Delta_1 + \Delta_2)\Delta_1$.*

Proof. Let $p \in P_{\text{cheap}}(I)$. Since p is cheap we have

$$\rho_{\text{st}}(p) = |p\,\text{succ}(p)| \leqslant \delta \cdot |s\,\text{succ}(p)|.$$

Since $\text{succ}(p) \in I$, we have $|s\,\text{succ}(p)| \leqslant \Delta_1 + \Delta_2$, and so $\rho_{\text{st}}(p) \leqslant \delta(\Delta_2 + \Delta_1)$. Hence,

$$\sum_{p \in P_{\text{cheap}}(I)} \rho_{\text{st}}(p)^2 \quad \leqslant \quad \delta(\Delta_2 + \Delta_1) \cdot \sum_{p \in P_{\text{cheap}}(I)} \rho_{\text{st}}(p) \quad \leqslant \quad \delta(\Delta_2 + \Delta_1)\Delta_1.$$
∎

We now prove the approximation ratio of our source-based range assignment. Let $\text{OPT} = \text{cost}_2(\rho_{\text{opt}}(P))$ denote the cost of the optimal range assignment on P.

LEMMA 6.8. *For any point set P in $\mathbb{R}^1$ and any $1/2 < \delta < 1$ we have*

$$\text{cost}_2(\rho_{\text{sb}}(P)) \leqslant c_\delta \cdot \textit{OPT}, \text{ where } c_\delta := \max\left(1 + \delta + \tfrac{(1+5\delta)(1-\delta)^2}{\delta^2}, \tfrac{1}{\delta^2} + \tfrac{1}{2}\right).$$

Proof. The worst approximation ratio is achieved by a set P with points to both sides of the source—indeed, if we only have points to the right of s, say, then adding an additional point slightly to the left of s will change neither the cost of an optimal solution nor the cost of ρ_{sb}. So from now on we assume that P has points to both sides of s. In the following, with a slight abuse of notation, we will use the notation p both for the point $p \in P$ and its value (that is, its x-coordinate when identifying $\mathbb{R}^1$ with the x-axis). For example, to indicate that a point p lies to the left of another point p' we may write $p < p'$. We will assume without loss of generality that $s = 0$.

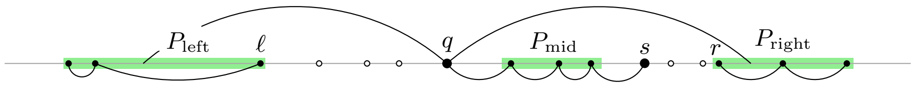

FIG. 10. *The cost of ρ_{opt} on $P_{\text{left}} \cup P_{\text{mid}} \cup P_{\text{right}}$ is at least the cost of ρ_{sb} on this set. Note that ℓ and/or r can lie exactly at the end of the range of q, that is, $|q\ell| = \rho_{\text{opt}}(q)$ and/or $|qr| = \rho_{\text{opt}}(q)$—in fact, one of this cases must happen. Note: color appears only in the online article.*

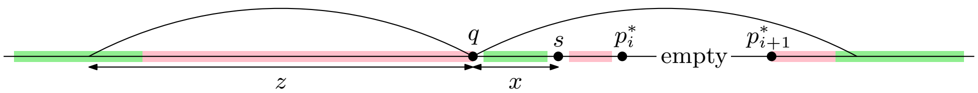

FIG. 11. *Relative position of the points s, q, p_i^*, p_{i+1}^* in Case 1. Costs of the points in the green regions are included in C_{sb} and C_{opt}. The cost for the other regions is analyzed in the text. Note: color appears only in the online article.*

Let $\rho_{\text{opt}}(P)$ be an optimal range assignment that induces a broadcast tree $\mathcal{B}$ with the structure of Theorem 2.1, and let q denote the root-crossing point in $\mathcal{B}$. Let p_i^* denote a crucial point in P, and let $p_{i+1}^* = \text{succ}(p_i^*)$. If $p_i^* = s$, then we define p_{i+1}^* to be a successor of s at maximum distance from s, so that in this case we also have $d_{\max} = |sp_{i+1}^*|$. Thus $\rho_{\text{sb}}(s) = d_{\max} = |sp_{i+1}^*|$.

Let ℓ and r be the leftmost and the rightmost points that are within range of q in the optimal solution, respectively. Let $P_{\text{left}} := \{p \in P : p \leqslant \ell\}$ be the set of all points to the left of ℓ plus ℓ itself, and let $P_{\text{right}} := \{p \in P : p \geqslant r\}$. Finally, let P_{mid} be the set of points in-between s and q, excluding both s and q; see Figure 10. We now define

$$C_{\text{sb}} := \sum_{p \in P_{\text{left}} \cup P_{\text{mid}} \cup P_{\text{right}}} \rho_{\text{sb}}(p)^2 \text{ and } C_{\text{opt}} := \sum_{p \in P_{\text{left}} \cup P_{\text{mid}} \cup P_{\text{right}}} \rho_{\text{opt}}(p)^2$$

as the costs incurred by ρ_{sb} and ρ_{opt} on the sets just defined. Observe that $C_{\text{opt}} \geqslant C_{\text{sb}}$, because $\rho_{\text{opt}}(p) = \rho_{\text{st}}(p)$ for all $p \in P_{\text{left}} \cup P_{\text{mid}} \cup P_{\text{right}}$, and $\rho_{\text{sb}}(p) \leqslant \rho_{\text{st}}(p)$ for all $p \in P \setminus \{s\}$.

We now analyze the costs incurred by ρ_{sb} and ρ_{opt} on the remaining points. We assume without loss of generality that $q \leqslant s$, and we let $x := |qs|$ denote the distance from q to s. Furthermore, we define $z := \rho_{\text{opt}}(q)$. Note that $z \geqslant x$. We divide the analysis into several cases, depending on the relative position of s, q, p_i^*, p_{i+1}^*.

Case 1*: p_i^* and p_{i+1}^* lie to the right of s (possibly $p_i^* = s$) and inside the range of q in the optimal solution.*

See Figure 11 for an illustration. Since p_i^* is crucial we have $|p_i^* p_{i+1}^*| > \delta \cdot |sp_{i+1}^*|$ and so

$$|sp_i^*| = |sp_{i+1}^*| - |p_i^* p_{i+1}^*| < \left(\frac{1}{\delta} - 1\right) |p_i^* p_{i+1}^*| = \left(\frac{1-\delta}{\delta}\right) |p_i^* p_{i+1}^*| \leqslant \left(\frac{1-\delta}{\delta}\right)(z-x).$$

We now bound the cost of the points $p \notin P_{\text{left}} \cup P_{\text{mid}} \cup P_{\text{right}}$. Theses are the points s, q, p_i^*, p_{i+1}^* plus the points in the red regions in Figure 11. Note that $\rho_{\text{sb}}(p_i^*) = 0$.

By applying Lemma 6.7 with $\Delta_1 \leqslant (\frac{1-\delta}{\delta})(z-x)$ and $\Delta_2 = 0$, we see that the cost incurred by ρ_{sb} due to the points strictly in-between s and p_i^* is less than or equal to $\frac{(1-\delta)^2}{\delta}(z-x)^2$. By applying Lemma 6.7 with $\Delta_1 = z$ and $\Delta_2 = x$, the cost incurred

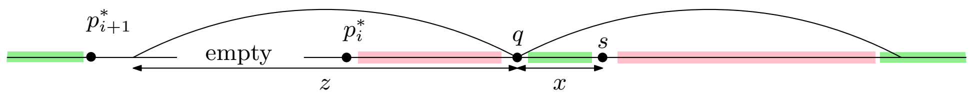

FIG. 12. *Relative position of the points s, q, p^*_i, p^*_{i+1} in Case 2. Note: color appears only in the online article.*

by $\rho_{\rm sb}$ due to the points in the region left of and including q and within the range of q is at most $\delta z(z+x)$. Finally, the cost incurred by $\rho_{\rm sb}$ due to the points to the right of and including p^*_{i+1} and within the range of q is at most $(z-x-|sp^*_{i+1}|)^2$. Since $\rho_{\rm sb}(s) = |sp^*_{i+1}|$ we obtain

$$\begin{aligned}\mathrm{cost}_2(\rho_{\rm sb}(P)) &\leqslant |sp^*_{i+1}|^2 + \delta z(z+x) + \frac{(1-\delta)^2}{\delta}(z-x)^2 \\ &\quad + (z-x-|sp^*_{i+1}|)^2 + C_{\rm sb}.\end{aligned}$$

Obviously, $\mathrm{cost}_2(\rho_{\rm opt}(P)) > z^2 + C_{\rm opt}$. Since $C_{\rm sb} \leqslant C_{\rm opt}$ and $x \leqslant z$ we conclude

$$\begin{aligned}\frac{\mathrm{cost}_2(\rho_{\rm sb}(P))}{\mathrm{cost}_2(\rho_{\rm opt}(P))} &\leqslant \frac{(z-x-|sp^*_{i+1}|)^2 + |sp^*_{i+1}|^2}{z^2} + \delta\frac{x}{z} + \frac{(1-\delta)^2}{\delta} \\ &\leqslant \frac{(z-x)^2}{z^2} + \delta\frac{x}{z} + \frac{(1-\delta)^2}{\delta} \\ &\leqslant 1 + \delta + \frac{(1-\delta)^2}{\delta} \\ &< c_\delta.\end{aligned}$$

Case 2: p^*_i lies to the left of q but within the range of q and p^*_{i+1} lies outside the range of q.

See Figure 12 for an illustration. As before, since p^*_i is crucial we have $|p^*_i p^*_{i+1}| > \delta \cdot |sp^*_{i+1}|$ and so

$$|sp^*_i| = |sp^*_{i+1}| - |p^*_i p^*_{i+1}| < \left(\frac{1-\delta}{\delta}\right)|p^*_i p^*_{i+1}|.$$

Applying Lemma 6.7 with $\Delta_1 + \Delta_2 = |sp^*_i|$ and $\Delta_1 = |qp^*_i| \leqslant z$, we see that the cost incurred by $\rho_{\rm sb}$ due to the points in the region between and including q and p^*_i is at most $(1-\delta)|p^*_i p^*_{i+1}|z$. Applying Lemma 6.7 with $\Delta_1 = z - x$ and $\Delta_2 = 0$, we see that the cost incurred by $\rho_{\rm sb}$ due to the points in the region to the right of s and within the range of q is at most $\delta(z-x)^2$. Furthermore, the cost incurred by $\rho_{\rm sb}$ due to the range of s is equal to $|sp^*_{i+1}|^2$.

Let $C^*_{\rm opt} := C_{\rm opt} - |p^*_i p^*_{i+1}|^2$. Note that in Case 2, p^*_i is the leftmost point in the range of q (the point ℓ in Figure 10) and so it is included in $P_{\rm left}$. Since $\rho_{\rm sb}(p^*_i) = 0$ this implies $C^*_{\rm opt} \geqslant C_{\rm sb}$. Hence,

$$\mathrm{cost}_2(\rho_{\rm opt}(P)) > z^2 + |p^*_i p^*_{i+1}|^2 + C^*_{\rm opt}.$$

Moreover,

$$\mathrm{cost}_2(\rho_{\rm sb}(P)) = |sp^*_{i+1}|^2 + (1-\delta)|p^*_i p^*_{i+1}|z + \delta(z-x)^2 + C_{\rm sb}.$$

Since $C_{\text{sb}} \leqslant C^*_{\text{opt}}$ we thus get

$$\frac{\text{cost}_2(\rho_{\text{sb}}(P))}{\text{cost}_2(\rho_{\text{opt}}(P))} \leqslant \frac{|sp^*_{i+1}|^2 + (1-\delta)|p^*_i p^*_{i+1}|z + \delta(z-x)^2}{z^2 + |p^*_i p^*_{i+1}|^2}.$$

Now if $|p^*_i p^*_{i+1}| \geqslant z$, and using that $x \leqslant z$, we find

$$\begin{aligned}\frac{\text{cost}_2(\rho_{\text{sb}}(P))}{\text{cost}_2(\rho_{\text{opt}}(P))} &\leqslant \frac{|sp^*_{i+1}|^2}{|p^*_i p^*_{i+1}|^2} + \frac{(1-\delta)|p^*_i p^*_{i+1}|z}{z^2 + |p^*_i p^*_{i+1}|^2} + \frac{\delta(z-x)^2}{2z^2} \\ &\leqslant \frac{1}{\delta^2} + \frac{1-\delta}{2} + \frac{\delta}{2} \\ &= \frac{1}{2} + \frac{1}{\delta^2}.\end{aligned}$$

On the other hand, if $|p^*_i p^*_{i+1}| < z$ we have $|sp^*_{i+1}| < |p^*_i p^*_{i+1}|/\delta < z/\delta$ and so we get

$$\begin{aligned}\frac{\text{cost}_2(\rho_{\text{sb}}(P))}{\text{cost}_2(\rho_{\text{opt}}(P))} &\leqslant \frac{|sp^*_{i+1}|^2}{z^2 + |p^*_i p^*_{i+1}|^2} + \frac{(1-\delta)|p^*_i p^*_{i+1}|z}{z^2 + |p^*_i p^*_{i+1}|^2} + \frac{\delta(z-x)^2}{z^2} \\ &< \frac{|p^*_i p^*_{i+1}|z}{(z^2 + |p^*_i p^*_{i+1}|^2)\delta^2} + \frac{(1-\delta)|p^*_i p^*_{i+1}|z}{z^2 + |p^*_i p^*_{i+1}|^2} + \frac{\delta(z-x)^2}{z^2} \\ &\leqslant \frac{1}{2\delta^2} + \frac{1-\delta}{2} + \delta.\end{aligned}$$

So we have,

$$\frac{\text{cost}_2(\rho_{\text{sb}}(P))}{\text{cost}_2(\rho_{\text{opt}}(P))} \leqslant \max\left(\frac{1}{2} + \frac{1}{\delta^2}, \frac{1}{2\delta^2} + \frac{1-\delta}{2} + \delta\right) = \frac{1}{\delta^2} + \frac{1}{2} \leqslant c_\delta.$$

Case 3: *p^*_i, p^*_{i+1} lie to the left of q and inside the range of q.*

See Figure 13 for an illustration. As before, since p^*_i is crucial we have $|sp^*_i| < \left(\frac{1-\delta}{\delta}\right)|p^*_i p^*_{i+1}|$, which in Case 3 implies $x \leqslant \frac{(1-\delta)}{\delta}z$ and $|qp^*_i| \leqslant \frac{(1-\delta)}{\delta}z$. Applying Lemma 6.7 with $\Delta_1 = |qp^*_i|$ and $\Delta_2 = x$, we see that the cost incurred by ρ_{sb} due to the points in the region between and including q and p^*_i is at most $\delta \cdot z(x + \frac{(1-\delta)}{\delta}z) \cdot \frac{(1-\delta)}{\delta}$. Applying Lemma 6.7 with $\Delta_1 = z - x$ and $\Delta_2 = 0$, we see that the cost incurred by ρ_{sb} due to the points in the region to the right of s and within the range of q is at most $\delta(z-x)^2$. Finally, the cost incurred by ρ_{sb} due to the points in the region left of p^*_{i+1} and within the range of q is at most $(z + x - |sp^*_{i+1}|)^2$. Hence,

$$\text{cost}_2(\rho_{\text{opt}}(P)) > z^2 + C_{\text{opt}}$$

and

$$\begin{aligned}\text{cost}_2(\rho_{\text{sb}}(P)) \leqslant{}& |sp^*_{i+1}|^2 + \delta(z-x)^2 + \delta \cdot \frac{(1-\delta)}{\delta} \cdot z\left(x + \frac{(1-\delta)}{\delta}z\right) \\ &+ (z + x - |sp^*_{i+1}|)^2 + C_{\text{sb}}.\end{aligned}$$

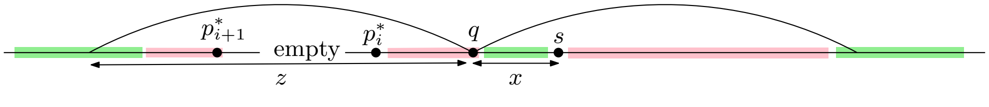

FIG. 13. *Relative position of the points s, q, p^*_i, p^*_{i+1} in Case* 3. *Costs of the points in the green regions are included in C_{sb} and C_{opt}. The cost for the other regions is analyzed in the text. Note: color appears only in the online article.*

Since $C_{\text{sb}} \leqslant C_{\text{opt}}$ this implies

$$\begin{aligned}\frac{\text{cost}_2(\rho_{\text{sb}}(P))}{\text{cost}_2(\rho_{\text{opt}}(P))} &\leqslant \frac{|sp^*_{i+1}|^2 + (z+x-|sp^*_{i+1}|)^2 + \delta(z-x)^2 + \delta\frac{(1-\delta)}{\delta}z(x+\frac{(1-\delta)}{\delta}z)}{z^2}\\ &\leqslant \frac{(z+x)^2 + \delta(z-x)^2 + (1-\delta)xz + \frac{(1-\delta)^2}{\delta}z^2}{z^2}\\ &\quad\times (\text{note that } |sp^*_{i+1}|^2 + (z+x-|sp^*_{i+1}|)^2 < (z+x)^2)\\ &= \frac{(1+\delta+\frac{(1-\delta)^2}{\delta})z^2 + (3-3\delta)zx + (1+\delta)x^2}{z^2}\\ &= (1+\delta)\frac{x^2}{z^2} + (3-3\delta)\frac{x}{z} + 1 + \delta + \frac{(1-\delta)^2}{\delta}.\end{aligned}$$

Now define $f(y) := (1+\delta)y^2 + (3-3\delta)y + 1 + \delta + \frac{(1-\delta)^2}{\delta}$, then the last term is equal to $f(x/z)$. Observe that f is quadratic in y and that $0 \leqslant x/z \leqslant (1-\delta)/\delta$. Hence,

$$\begin{aligned}\frac{\text{cost}_2(\rho_{\text{sb}}(P))}{\text{cost}_2(\rho_{\text{opt}}(P))} &\leqslant \max(f(0), f((1-\delta)/\delta))\\ &= \max\left(1+\delta+\frac{(1-\delta)^2}{\delta}, 1+\delta+(1+5\delta)\frac{(1-\delta)^2}{\delta^2}\right)\\ &= 1+\delta+(1+5\delta)\frac{(1-\delta)^2}{\delta^2}\\ &\leqslant c_\delta.\end{aligned}$$

Case 4: p^*_i, p^*_{i+1} lie to the left of q but outside the range of q; or p^*_i, p^*_{i+1} lie in the region $[s,q]$.

See Figure 14 for an illustration. Since p^*_i is crucial we have $|sp^*_{i+1}| < |p^*_i p^*_{i+1}|/\delta$. Clearly the cost incurred by ρ_{sb} due to the points in the region to the left of q and within the range of q is at most z^2. Applying Lemma 6.7 with $\Delta_1 = z - x$ and $\Delta_2 = 0$, the cost incurred by ρ_{sb} due to the points in the region to the right of s and within the range of q is at most $\delta(z-x)^2$. Finally, the cost incurred by ρ_{sb} due to the range of s is $|sp^*_{i+1}|^2$. Define $C^*_{\text{opt}} := C_{\text{opt}} - |p^*_i p^*_{i+1}|^2$ and observe that $C^*_{\text{opt}} \geqslant C_{\text{sb}}$. Then

$$\text{cost}_2(\rho_{\text{opt}}(P)) > z^2 + |p^*_i p^*_{i+1}|^2 + C^*_{\text{opt}}$$

and

$$\text{cost}_2(\rho_{\text{sb}}(P)) \leqslant |sp^*_{i+1}|^2 + z^2 + \delta(z-x)^2 + C_{\text{sb}}$$

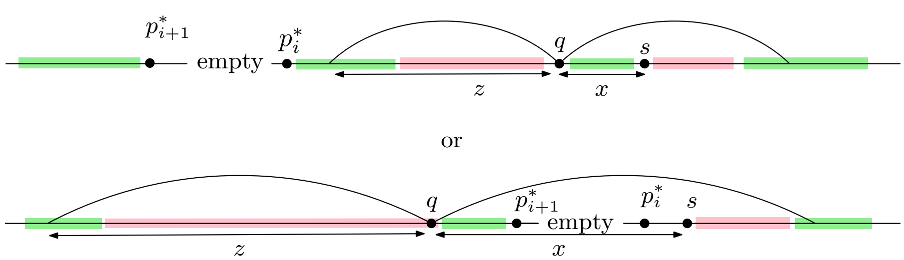

FIG. 14. *The two options for relative position of the points s, q, p^*_i, and p^*_{i+1} in Case 4. Note: color appears only in the online article.*

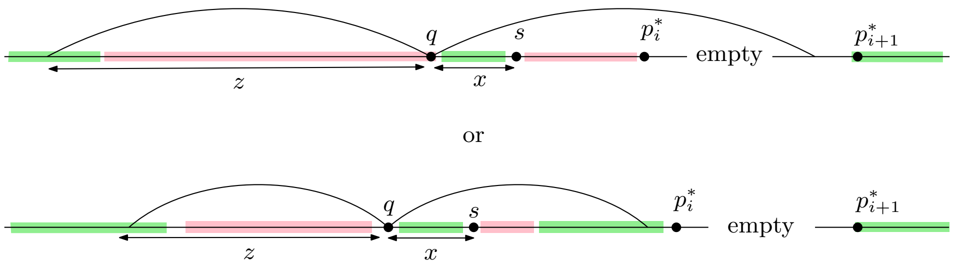

FIG. 15. *The two options for the relative position of the points s, q, p_i^*, and p_{i+1}^* in Case 5. Note: color appears only in the online article.*

with $C_{\mathrm{sb}} \leqslant C^*_{\mathrm{opt}}$. Since $x \leqslant z$ we thus obtain

$$\begin{aligned}\frac{\mathrm{cost}_2(\rho_{\mathrm{sb}}(P))}{\mathrm{cost}_2(\rho_{\mathrm{opt}}(P))} &\leqslant frac|sp^*_{i+1}|^2 + z^2 + \delta(z-x)^2 z^2 + |p^*_i p^*_{i+1}|^2 \\ &\leqslant \frac{\frac{1}{\delta^2}|p^*_i p^*_{i+1}|^2| + (1+\delta)z^2}{z^2 + |p^*_i p^*_{i+1}|^2} \\ &\leqslant \max\left(1+\delta, \frac{1}{\delta^2}\right) \\ &< c_\delta.\end{aligned}$$

Case 5*: p_i^* lies to the right of s and within the range of q and p^*_{i+1} lies to the right of s and outside the range of q; or p_i^*, p^*_{i+1} lie to the right of s and outside the range of q.*

See Figure 15 for an illustration. As before, we have $|sp_i^*| < \frac{(1-\delta)}{\delta}|p_i^* p^*_{i+1}|$ and $|sp^*_{i+1}| < \frac{1}{\delta}|p^*_i p^*_{i+1}|$. Clearly the cost incurred by ρ_{sb} due to the points in the region to the left of q and within the range of q is at most z^2, and the cost incurred by ρ_{sb} due to the points in the region right of s and within the range of q is at most $|sp_i^*|^2 < \frac{(1-\delta)^2}{\delta}|p^*_i p^*_{i+1}|^2$. Finally, the cost incurred by ρ_{sb} due to source s is $|sp^*_{i+1}|^2$ which is at most $\frac{1}{\delta^2}|p^*_i p^*_{i+1}|^2$. Define $C^*_{\mathrm{opt}} := C_{\mathrm{opt}} - |p^*_i p^*_{i+1}|^2$ and observe that $C^*_{\mathrm{opt}} \geqslant C_{\mathrm{sb}}$. Then

$$\mathrm{cost}_2(\rho_{\mathrm{opt}}(P)) > z^2 + |p^*_i p^*_{i+1}|^2 + C^*_{\mathrm{opt}}$$

and

$$\mathrm{cost}_2(\rho_{\mathrm{sb}}(P)) \leqslant |sp^*_{i+1}|^2 + z^2 + \frac{(1-\delta)^2}{\delta}|p^*_i p^*_{i+1}|^2 + C_{\mathrm{sb}}.$$

Since $C_{\mathrm{sb}} \leqslant C^*_{\mathrm{opt}}$ and $\delta < 1$ we conclude

$$\begin{aligned}\frac{\mathrm{cost}_2(\rho_{\mathrm{sb}}(P))}{\mathrm{cost}_2(\rho_{\mathrm{opt}}(P))} &\leqslant \frac{|sp^*_{i+1}|^2 + z^2 + \frac{(1-\delta)^2}{\delta}|p^*_i p^*_{i+1}|^2}{z^2 + |p^*_i p^*_{i+1}|^2} \\ &\leqslant \frac{(\frac{1}{\delta^2} + \frac{(1-\delta)^2}{\delta})|p^*_i p^*_{i+1}|^2 + z^2}{z^2 + |p^*_i p^*_{i+1}|^2} \\ &\leqslant \frac{(1-\delta)^2}{\delta} + \frac{1}{\delta^2} \\ &< c_\delta.\end{aligned}$$

We conclude that the approximation ratio is bounded by c_δ in all cases. ∎

We now want to choose δ so as to minimize $c_\delta = \max(1+\delta+\frac{(1+5\delta)(1-\delta)^2}{\delta^2}, \frac{1}{\delta^2}+\frac{1}{2})$. The first term is minimized at the real root of the polynomial $6\delta^3-3\delta-2$, whose approximate value is 0.92711; this gives a value that is approximately 1.97. For this value of δ the first term dominates the second one, leading to the following theorem.

THEOREM 6.9. *There exists a* 3*-stable* 1.97*-approximation algorithm for the dynamic broadcast range-assignment problem in* $\mathbb{R}^1$ *for* $\alpha = 2$.

7. Concluding remarks.

We studied the dynamic broadcast range-assignment problem from a stability perspective, introducing the notions of k-stable algorithms and SASs. Our results provide a fairly complete picture of the problem in $\mathbb{R}^1$, in $\mathbb{S}^1$, and in $\mathbb{R}^2$. In particular, we presented a SAS in $\mathbb{R}^1$ that has an asymptotically optimal stability parameter, and showed that the problem does not admit a SAS in $\mathbb{S}^1$ and $\mathbb{R}^2$. Future work can focus on improving the (upper and/or lower bounds for) approximation ratios that we have obtained for algorithms with a constant stability parameter. While generalizing the structure theorem that exists for $\mathbb{R}^1$ to $\mathbb{R}^2$ will be difficult, weaker properties of an optimal solution in $\mathbb{R}^2$ (or in $\mathbb{R}^d$, $d \geqslant 2$) could be obtained. Also one can work on good approximation algorithms with bounded stability parameter on the circle.

Acknowledgment. We thank the reviewers of an earlier version of the paper for pointing us to some important references and for other helpful comments.

REFERENCES

[1] A. Aggarwal, L. J. Guibas, J. B. Saxe, and P. W. Shor, *A linear-time algorithm for computing the Voronoi diagram of a convex polygon*, Discrete Comput. Geom., 4 (1989), pp. 591–604, https://doi.org/10.1007/BF02187749.

[2] C. Ambühl, *An optimal bound for the MST algorithm to compute energy efficient broadcast trees in wireless networks*, in Proceedings 32nd International Colloquium on Automata, Languages and Programming (ICALP 2005), Lecture Notes in Comput. Sci. 3580, Springer, Berlin, 2005, pp. 1139–1150, https://doi.org/10.1007/11523468_92.

[3] S. Angelopoulos, C. Dürr, and S. Jin, *Online maximum matching with recourse*, in Proceedings of the 43rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2018), LIPIcs Leibniz Int. Proc. Inform. 117, 2018, 8, https://doi.org/10.4230/LIPIcs.MFCS.2018.8.

[4] M. R. Ataei, A. H. Banihashemi, and T. Kunz, *Low-complexity energy-efficient broadcasting in one-dimensional wireless networks*, IEEE Trans. Veh. Technol., 61 (2012), pp. 3276–3282, https://doi.org/10.1109/TVT.2012.2204077.

[5] T. Avitabile, C. Mathieu, and L. H. Parkinson, *Online constrained optimization with recourse*, Inform. Process. Lett., 113 (2013), pp. 81–86, https://doi.org/10.1016/j.ipl.2012.09.011.

[6] M. A. Bassiouni and C. Fang, *Dynamic channel allocation for linear macrocellular topology*, in Proceedings of the 1999 ACM Symposium on Applied Computing, ACM, New York, 1999, pp. 382–388, https://doi.org/10.1145/298151.298391.

[7] S. Behnezhad, J. Lacki, and V. S. Mirrokni, *Fully dynamic matching: Beating* 2*-approximation in* Δ^ϵ *update time*, in Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadelphia, 2020, pp. 2492–2508, https://doi.org/10.1137/1.9781611975994.152.

[8] A. Bernstein, J. Holm, and E. Rotenberg, *Online bipartite matching with amortized* $O(\log^2 n)$ *replacements*, J. ACM, 66 (2019), pp. 37, https://doi.org/10.1145/3344999.

[9] I. Caragiannis, M. Flammini, and L. Moscardelli, *An exponential improvement on the MST heuristic for minimum energy broadcasting in ad hoc wireless networks*, IEEE/ACM Trans. Netw., 21 (2013), pp. 1322–1331, https://doi.org/10.1109/TNET.2012.2223483.

[10] I. Caragiannis, C. Kaklamanis, and P. Kanellopoulos, *New results for energy-efficient broadcasting in wireless networks*, in Proceedings of the 13th International Symposium on Algorithms and Computation (ISAAC 2002), Lecture Notes in Comput. Sci. 2518, Springer, Berlin, 2002, pp. 332–343, https://doi.org/10.1007/3-540-36136-7_30.

[11] B. Chazelle, *A minimum spanning tree algorithm with inverse-Ackermann type complexity*, J. ACM, 47 (2000), pp. 1028–1047, https://doi.org/10.1145/355541.355562.
[12] A. E. F. Clementi, G. Huiban, P. Penna, G. Rossi, and Y. C. Verhoeven, *Some recent theoretical advances and open questions on energy consumption in ad-hoc wireless networks*, in Proceedings of the 3rd Workshop on Approximation and Randomization Algorithms in Communication Networks (ARACNE 2002), Carleton, Scientific, Waterloo, Canada, 2002.
[13] A. E. F. Clementi, M. D. Ianni, and R. Silvestri, *The minimum broadcast range assignment problem on linear multi-hop wireless networks*, Theoret. Comput. Sci., 299 (2003), pp. 751–761, https://doi.org/10.1016/S0304-3975(02)00538-8.
[14] A. E. F. Clementi, P. Penna, A. Ferreira, S. Perennes, and R. Silvestri, *The minimum range assignment problem on linear radio networks*, Algorithmica, 35 (2003), pp. 95–110, https://doi.org/10.1007/s00453-002-0985-2.
[15] A. E. F. Clementi, P. Penna, and R. Silvestri, *Hardness results for the power range assignment problem in packet radio networks*, in Proceedings of the 3rd International Workshop on Randomization and Approximation Techniques in Computer Science, and 2nd International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (RANDOM-APPROX'99), Lecture Notes in Comput. Sci. 1671, Springer, Berlin, 1999, pp. 197–208, https://doi.org/10.1007/978-3-540-48413-4_21.
[16] A. E. F. Clementi, P. Penna, and R. Silvestri, *The power range assignment problem in radio networks on the plane*, in Proceedings of the 17th Annual Symposium on Theoretical Aspects of Computer Science (STACS), Lecture Notes in Comput. Sci. 1770, Springer, Berlin, 2000, pp. 651–660, https://doi.org/10.1007/3-540-46541-3_54.
[17] V. Cohen-Addad, N. Hjuler, N. Parotsidis, D. Saulpic, and C. Schwiegelshohn, *Fully dynamic consistent facility location*, in Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Curran Associates, Red Hook, NY, 2019, pp. 3250–3260, https://proceedings.neurips.cc/paper/2019/hash/fface8385abbf94b4593a0ed53a0c70f-Abstract.html.
[18] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed., MIT Press, Cambridge, 2009, http://mitpress.mit.edu/books/introduction-algorithms.
[19] G. K. Das, S. Das, and S. C. Nandy, *Range assignment for energy efficient broadcasting in linear radio networks*, Theoret. Comput. Sci., 352 (2006), pp. 332–341, https://doi.org/10.1016/j.tcs.2005.11.046.
[20] G. K. Das and S. C. Nandy, *Weighted broadcast in linear radio networks*, Inform. Process. Lett., 106 (2008), pp. 136–143, https://doi.org/10.1016/j.ipl.2007.10.016.
[21] M. de Berg, O. Cheong, M. J. van Kreveld, and M. H. Overmars, *Computational Geometry: Algorithms and Applications*, 3rd ed., Springer, Berlin, 2008.
[22] M. de Berg, A. Markovic, and S. W. Umboh, *The online broadcast range-assignment problem*, in Proceedings of the 31st International Symposium on Algorithms and Computation (ISAAC), LIPIcs Leibniz Int. Proc. Inform. 181, 2020, 60, https://doi.org/10.4230/LIPIcs.ISAAC.2020.60.
[23] H. Edelsbrunner and E. P. Mücke, *Simulation of simplicity: A technique to cope with degenerate cases in geometric algorithms*, ACM Trans. Graph., 9 (1990), pp. 66–104, https://doi.org/10.1145/77635.77639.
[24] H. Fichtenberger, S. Lattanzi, A. Norouzi-Fard, and O. Svensson, *Consistent k-clustering for general metrics*, in Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA 2021), SIAM, Philadelphia, 2021, pp. 2660–2678, https://doi.org/10.1137/1.9781611976465.158.
[25] B. Fuchs, *On the hardness of range assignment problems*, Networks, 52 (2008), pp. 183–195, https://doi.org/10.1002/net.20227.
[26] A. Gu, A. Gupta, and A. Kumar, *The power of deferral: Maintaining a constant-competitive Steiner tree online*, SIAM J. Comput., 45 (2016), pp. 1–28, https://doi.org/10.1137/140955276.
[27] A. Gupta and A. Kumar, *Online steiner tree with deletions*, in Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2014), SIAM, Philadelphia, 2014, pp. 455–467, https://doi.org/10.1137/1.9781611973402.34.
[28] A. Gupta, A. Kumar, and C. Stein, *Maintaining assignments online: Matching, scheduling, and flows*, in Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2014), SIAM, Philadelphia, 2014, pp. 468–479, https://doi.org/10.1137/1.9781611973402.35.
[29] X. Han and K. Makino, *Online minimization knapsack problem*, Theoret. Comput. Sci., 609 (2016), pp. 185–196, https://doi.org/10.1016/j.tcs.2015.09.021.

[30] M. Imase and B. M. Waxman, *Dynamic steiner tree problem*, SIAM J. Discrete Math., 4 (1991), pp. 369–384, https://doi.org/10.1137/0404033.

[31] K. Iwama and S. Taketomi, *Removable online knapsack problems*, in Proceedings of the 29th International Colloquium Automata, Languages and Programming (ICALP 2002), Lecture Notes in Comput. Sci. 2380, Springer, Berlin, 2002, pp. 293–305, https://doi.org/10.1007/3-540-45465-9_26.

[32] L. M. Kirousis, E. Kranakis, D. Krizanc, and A. Pelc, *Power consumption in packet radio networks*, Theoret. Comput. Sci., 243 (2000), pp. 289–305, https://doi.org/10.1016/S0304-3975(98)00223-0.

[33] S. Lattanzi and S. Vassilvitskii, *Consistent k-clustering*, in Proceedings of the 34th International Conference on Machine Learning (ICML 2017), Proc. Mach. Learn. Res. 70, 2017, pp. 1975–1984, http://proceedings.mlr.press/v70/lattanzi17a.html.

[34] R. Mathar and J. Mattfeldt, *Optimal transmission ranges for mobile communication in linear multihop packet radio networks*, Wirel. Netw., 2 (1996), pp. 329–342, https://doi.org/10.1007/BF01262051.

[35] N. Megow, M. Skutella, J. Verschae, and A. Wiese, *The power of recourse for online MST and TSP*, SIAM J. Comput., 45 (2016), pp. 859–880, https://doi.org/10.1137/130917703.

[36] K. Pahlavan and A. H. Levesque, *Wireless Information Networks*, 2nd ed., Wiley Ser. Telecomm. Signal Process., Wiley, Hoboken, NJ, 2005.

[37] P. Sanders, N. Sivadasan, and M. Skutella, *Online scheduling with bounded migration*, Math. Oper. Res., 34 (2009), pp. 481–498, https://doi.org/10.1287/moor.1090.0381.

[38] M. Skutella and J. Verschae, *A robust PTAS for machine covering and packing*, in Proceedings of the 18th Annual European Symposium on Algorithms (ESA 2010), Lecture Notes in Comput. Sci. 6346, Springer, Berlin, 2010, pp. 36–47, https://doi.org/10.1007/978-3-642-15775-2_4.

[39] P. M. Spira and A. Pan, *On finding and updating spanning trees and shortest paths*, SIAM J. Comput., 4 (1975), pp. 375–380, https://doi.org/10.1137/0204032.

[40] J. Wulms, *Stability of Geometric Algorithms*, Ph.D. thesis, Eindhoven University of Technology, Eindhoven, The Netherlands, 2020.