On Construction of Minimum Energy k-Fault Resistant Topologies

Michael Segal*
Communication Systems Engineering Dept.
Ben-Gurion University of the Negev
Beer-Sheva, Israel
segal@cse.bgu.ac.il

Hanan Shpungin†
Computer Sciences Dept.
Ben-Gurion University of the Negev
Beer-Sheva, Israel
shpungin@cs.bgu.ac.il

December 25, 2007

Abstract

This paper studies asymmetric power assignments for various network topologies under the k-resilience criterion in static and dynamic geometric settings. We aim to minimize the total energy consumption, which is usually NP-hard for a desired link topology. We develop a general approximation framework for various topology control problems under the k-fault resilience criterion in the plane. We use it to obtain an $O(k^2)$ approximation ratio for three k-fault resistant topology control problems: *multicast*, *broadcast* and *convergecast*. To the best of our knowledge, these are the first non-trivial results for these problems. In addition we present interesting results for the linear case of k-multicast, k-broadcast and k-convergecast. We also extend our static algorithms for k-strong connectivity in [15] and [41] to support dynamic node *insert/delete* operations in $O(\log n)$ time for the linear case and an expected $O(k \text{ poly} \log n)$ amortized time in the plane.

1 Introduction

A wireless ad-hoc network consists of several transceivers (nodes) $\mathcal{T} = \{t_1, t_2, \ldots, t_n\}$ located in the plane and communicating by radio. The underlying physical topology of the network is dependent on the distribution of the wireless nodes (location) as well as the transmission power (range) assignment of each node. The transmission range of node t is determined by the power assigned to that node, denoted by $p(t)$. This is customary to assume that the minimal transmission power required to transmit to distance d is d^α, where the *distance-power gradient* α is usually taken to be in the interval $[2, 4]$ (see [39]). Thus, node t receives transmissions from s if $p(s) \geq d(s,t)^\alpha$, where $d(s,t)$ is the Euclidean distance between s and t. A power assignment for $\mathcal{T}$ is a vector of transmission powers $A = \{p(t) \mid t \in \mathcal{T}\}$. The transmission possibilities resulting from a power assignment induce a directed communication graph $H_A = (\mathcal{T}, \mathcal{E}_A)$, where $\mathcal{E}_A = \{(s,t) \mid p(s) \geq d(s,t)^\alpha\}$ is the set of directed edges resulting from the power assignment. The cost of the power assignment is defined as the sum of all transmission powers, $C_A = \sum_{t \in \mathcal{T}} p(t)$.

There are two possible models: symmetric and asymmetric. In symmetric settings node s can reach node t if and only if node t can reach node s. That is, for any $s, t \in T$, $p(s) \geq d(s,t)^\alpha \Leftrightarrow p(t) \geq d(t,s)^\alpha$. We can also refer to it as the undirected model. The asymmetric variant allows directed links between two nodes. Krumke et al. [33] argued that the asymmetric version is harder than the symmetric one.

The most fundamental problem in wireless ad-hoc networks is to find a power assignment which induces a communication graph that satisfies some topology property, while minimizing the total cost. The *strong connectivity* (all-to-all) property has been the first studied problem in this area. This property is extremely useful in certain applications of wireless networks (e.g., battlefield or a rescue operation). Other key properties are *broadcast* (one-to-all) and *multicast* (one-to-many), which are very close from the designer point of view, while *broadcast* being a special case of *multicast*. The transmission is initiated by some source s and the message needs to be transmitted to all or some nodes (respectively). These two properties represent a major part of the activities in real life multi-hop radio networks. The *convergecast* (all-to-one) property has received less attention but might prove to be useful in networks where it is

*Supported by REMON (4G networking) consortium.
†Supported in part by the Lynn and William Frankel Center for Computer Sciences.

important to have transmissions from all nodes directed to one special node (e.g. temperature reading from multiple sensors). The network topology is in general dynamic, because the connectivity among the nodes may vary with time due to node departures and new node arrivals.

Let $P(G)$ be the set of all topology properties satisfied by G. A natural extension to the topology control problem is to impose the constraint of fault resistance. A graph property $\mathcal{P}$ is said to be k-fault resistant in G if for any subset of nodes X, where $|X| < k$, the condition $\mathcal{P} \in P(G \setminus X)$ holds. That is, the property $\mathcal{P}$ is satisfied by G even after the extraction of at most $k-1$ nodes. The benefits of a k-fault resistant topology is the multi-path redundancy for load balancing and higher transmission reliability.

This paper is organized as follows. In the rest of this section we give a formal problem statement, discuss the previous work and state our results. In section 2 we present our framework for planar k-fault resistance and present approximation algorithms for k-broadcast, k-multicast and k-convergecast, which, to the best of our knowledge, are the first non-trivial results for these problems. Next, in section 3 we deal with the linear case of these problems. The dynamic conversion of static algorithms for planar and linear k-strong connectivity is discussed in section 4. To the best of our knowledge this is a first attempt to present non-trivial bounds for maintenance of k-strong connectivity under dynamic updates. Finally, we conclude in section 5.

1.1 Problem Statement

We are given a set of transceivers $\mathcal{T}$, a desired topology property $\mathcal{P}$ and a natural number $k > 1$ (the *fault resistance parameter*). Our goal is to find a power assignment $A(\mathcal{T})$ so that the topology property $\mathcal{P}$ is k-fault resistant in the induced communication graph H_A and the power assignment cost C_A is minimized. Namely, we define the following optimization problem.

TOPOLOGY-CONTROL-PROBLEM($\mathcal{T}$,$\mathcal{P}$, k)
Input: A set $\mathcal{T}$ of transceivers, a topology property $\mathcal{P}$ and a fault resistance parameter $k > 1$.
Output: A power assignment $A(\mathcal{T})$ so that $\mathcal{P}$ is k-fault resistant in $P(H_A)$.
Target Function: Minimize C_A.

We address the following topology properties defined for a directed graph $G = (V, E)$.

- **Strong Connectivity** — A graph $G = (V, E)$ is *strongly connected* if for any two nodes $u, v \in V$ there exists a path from u to v.
- **Multicast from** r — A graph $G = (V, E)$ is *multicasting* from a root node $r \in V$ to a subset of nodes $M \subseteq V$, if for any $v \in M$ there exists a path from r to v.
- **Broadcast from** r — A graph $G = (V, E)$ is *broadcasting* from a root node $r \in V$, if for any $v \in V$ there exists a path from r to v. This is a special case of multicast with $M = V$.
- **Convergecast to** r — A graph $G = (V, E)$ is *convergecasting* to a root node $r \in V$, if for any $v \in V$ there exists a path from v to r.

Note that if graph G is strongly connected, then it is also *multicasting/broadcasting from* and *convergecasting to* any node in V. That is if graph G satisfies the strong connectivity property, it also satisfies the other three properties. Moreover, if the strong connectivity property is k-fault resistant in G then the other properties are also k-fault resistant in G. Given two topology properties $\mathcal{P}_1, \mathcal{P}_2$, if $\mathcal{P}_1 \in P(G) \Rightarrow \mathcal{P}_2 \in P(G)$ we say $\mathcal{P}_2$ is *weaker* than $\mathcal{P}_1$.

We use an abbreviate terminology of k-*<property-name>* problem, e.g. k-broadcast. We omit k, if $k = 1$. Note that providing higher order fault resistance requires a more expensive power assignment than is needed for a lower order fault resistance. That is $C_{A^*_{k_1}} \leq C_{A^*_{k_2}}$ iff $k_1 \leq k_2$.

1.2 Previous Work

Topology control in wireless networks is a relatively new field of interest. Nevertheless a wide area of problems has already been studied. In [40] the authors initiated the formal study of controlling the network topology by adjusting the transmission power of the nodes. Most of the problems are aimed at computing a low energy power assignment that meets global topological constraints. Kirousis et al. [32] introduced the `MinRange(SC)` problem, which is the k-strong connectivity problem for $k = 1$. They proved it to be NP-hard for the three dimensional Euclidean space for any value of α. The same paper provided a 2-approximation algorithm for the planar case and an exact $O(n^4)$ time algorithm for the one dimensional case. In the planar case, the NP-hardness of the problem for every α has been proved in [22] and a simple 1.5-approximation algorithm for the case of $\alpha = 1$ has been provided in [4]. Some researchers add an additional constraint parameter to the problem, the bounded diameter h of the induced communication graph, see results in [21, 23, 25]. Ambühl et al. [3] presented some algorithms for the weighted power assignment, solving it optimally for the broadcast, multi-source broadcast and strong connectivity problems for the linear case (they achieved the same running time for the strong connectivity problem as in [32]). They also presented some approximation algorithms for the multi-dimensional case. An excellent survey covering many variations of the problem is given in [20].

Wieselthier et al. in [46, 47] were the first to study the broadcast problem in wireless ad-hoc networks for the 2-dimensional case and when $\alpha = 2$. In this work, the performances of three heuristics, namely the minimum spanning tree (MST), the shortest path tree (SPT) and the broadcasting incremental power (BIP) have been experimentally compared on the random uniform model without providing theoretical results. In addition, multicasting incremental power (MIP) was developed by adapting BIP. The approach taken in [46, 47] is to build a source rooted spanning tree by adjusting transmit powers of nodes, followed by a sweep operation to remove redundant transmissions. Wan et al. in [44] present the first analytical results for this problem by exploring geometric structures of Euclidean MSTs. In particular, they prove that the approximation ratio of an MST is between 6 and 12, for BIP it is between $\frac{1}{4}$ and 12 and for SPT it is at least $\frac{n}{2}$, where n is the number of receiving nodes given that there are no obstacles in the network and that the fixed energy cost for electronics is negligible. The authors of [46] adapt BIP to the multicast problem as well. Cagalj et al. [10] give a proof of NP-hardness of the minimum-energy broadcast problem in a Euclidean space. Many researchers provided analytic results of the minimum-energy broadcast algorithm based on computing an MST. In [2] Ambühl et al. proves an approximation factor of 6, which matches the lower bound previously known for this algorithm. Flammini et al. [27] establish improved approximation results on the performance of BIP. Cartigny et al. in [16] develop localized algorithms for minimum-energy broadcasting.

The multicast problem is also NP-hard since it is a more general case of the broadcast problem. For asymmetric graphs Liang in [34] shows an $O(|M|^\epsilon)$ (M is the multicast destination set) and an $O(n^\epsilon)$ approximation algorithms for the multicast and broadcast problems, respectively, for any constant ϵ. The same work also presents an $O(\ln^3 n)$-approximation algorithm for the multicast problem in symmetric graphs. In [35] Liang develops an approximation algorithm with energy consumption no more than $O(|M|^\epsilon)$ or $4 \ln |M|$ for the symmetric and asymmetric multicast tree problems, respectively. Wan et al. [45] show a constant factor approximation for the minimum power asymmetric multicast problem. Additional references and results may be found in [2, 6, 8, 11, 14, 35, 38].

It is believed that using a broadcast tree for convergecast is enough, since convergecast is usually preceded by broadcast tree construction. Upadhyayula et al. in [43] claim that solving the convergecast using broadcast trees is in fact inefficient in terms of energy consumption and latency of communication. The authors proposed a method which is measured using latency, energy and reliability. Simulation results showed that the proposed method performs better than [5]. In addition, they claim that their approach performs better than reusing a tree constructed by a broadcast approach in [19]. Additional work in the field can be found in [30, 36].

The hardness of topology control problems lacking the fault resistance constraint ($k = 1$) implies that topology control problems with a higher value of k are also NP-hard. A first non trivial result for planar asymmetric k-strong connectivity was presented by Shpungin and Segal in [41]. They derived an approximation factor of $O(k^2)$ for the planar case and some results for the linear case. Carmi et al. [15] improved the approximation ratio to $O(k)$. Another possible connectivity property is k-edge connectivity, which implies that there is a path from any node t to any node s even with the removal of at most $k - 1$ edges. In [13], Calinescu and Wan presented various aspects of symmetric/asymmetric k-strong connectivity and k-edge connectivity. They first proved NP-hardness of the symmetric two-edge and two-node strong connectivity and then provided a 4-approximation algorithm for both symmetric and asymmetric strong biconnectivity ($k = 2$) and a $2k$-approximation for both symmetric and asymmetric k-edge strong connectivity. Hajiaghayi et al. [28] give two algorithms for symmetric k-strong connectivity, with $O(k \log k)$ and $O(k)$-approximation factors and also some distributed approximation algorithms for $k = 2$ and $k = 3$ in geometric graphs. Jia et al. in [31] present various approximation factors (depending on k) for the symmetric k-strong connectivity, such as $3k$-approximation algorithm for any $k \geq 3$ and 6-approximation algorithm for $k = 3$. Additional results can be found in [1, 9, 12, 18, 24, 29, 37, 42].

1.3 Our Contribution

For the linear case, we solve optimally the k-convergecast problem and present approximation algorithms for the k-multicast and k-broadcast problems with energy consumption of at most 2 times the optimum. For the planar case, we develop a general approximation framework for various topology control problems and obtain an $O(k^2)$ approximation ratio for three k-fault resistant topology control problems: *multicast*, *broadcast* and *convergecast*. The framework is based on estimating the cost of the topology control problem instance in terms of the cost of MST. In addition we extend our static algorithms for k-strong connectivity in [15] and [41] to support node *insert/delete* operations in $O(\log n)$ time for the linear case and an expected $O(k \text{ poly} \log n)$ amortized time in the plane.

2 Planar k-Fault Resistant Topology Framework

We make use of the following two common definitions from the graph theory:
A square of graph $G = (V, E)$ – is defined as $G^2 = (V, E^2)$, where $(u, w) \in E^2$ if $(u, w) \in E$ or there exists $v \in V$ such that $(u, v), (v, w) \in E$.
A Hamiltonian cycle – is a graph cycle that visits each node exactly once. A graph possessing a Hamiltonian cycle is said to be a Hamiltonian graph or simply Hamiltonian. The cost of a Hamiltonian cycle $h = (t_1, t_2, \ldots, t_n, t_{n+1} = t_1)$ is:

$$C_h = \sum_{i=1}^{n} d(t_i, t_{i+1})^\alpha.$$

We have developed a framework which provides k-fault resistance to a family of topology control problems. In particular we show approximation algorithms for three problems k-multicast, k-broadcast and k-convergecast.

Let $G_\mathcal{T} = (\mathcal{T}, E_\mathcal{T}, c_\mathcal{T})$ be a complete directed graph with an edge set $E_\mathcal{T} = \{(s,t) : s, t \in \mathcal{T}\}$ and a cost function $c_\mathcal{T}(s,t) = d(s,t)^\alpha$. Let MST be a minimum spanning tree of $G_\mathcal{T}$. The cost of MST is defined as $C_{\text{MST}} = \Sigma_{(s,t)\in \text{MST}} d(s,t)^\alpha$. The cost of any undirected graph G which appears in this section is defined in a similar manner, i.e., $C_G = \Sigma_{(s,t)\in G} d(s,t)^\alpha$.

Due to the distance-power gradient α the edge costs do not hold the triangle inequality. However the *relaxed triangle inequality* is satisfied. In our cost model, for any $u, v, w \in \mathcal{T}$ and any $\alpha \geq 2$, it holds

$$c(u, w) < 2^{\alpha-1}(c(u, v) + c(v, w)).$$

The framework consists of two stages. First we construct a Hamiltonian cycle with a cost $O(C_{\text{MST}})$. Then each node is assigned a transmission range along the cycle with respect to the topology property $\mathcal{P}$ and the fault resistance parameter k. The resulting power assignment A_k has a cost $C_{A_k} \in O(k^2 C_{\text{MST}})$. For simplicity we assume $\alpha = 2$, but our results can be easily extended for any fixed α. Next, we explain the stages in detail.

2.1 The Framework

The Hamiltonian cycle construction is based on the **TSP-APPROX** algorithm in [7]. The authors use the fact that the square of every 2-node connected graph is Hamiltonian to find a low cost Hamiltonian cycle. We use the **MST-Augmentation** algorithm presented in [13] for constructing an undirected 2-node connected graph. For every non-leaf node $v \in \mathcal{T}$, the algorithm constructs a local Euclidean minimum spanning tree MST_v over all its neighbors. As a result we obtain an undirected graph H_2 with edges of both the initial MST and all the local minimum spanning trees MST_v.

Hamiltonian Cycle Stage (I)

1. Find MST.
2. Apply the **MST-Augmentation** algorithm to obtain a 2-node connected graph H_2.
3. Apply the **TSP-APPROX** algorithm over H_2 to obtain a Hamiltonian cycle h of the square of H_2.

Figure 1 demonstrates these steps. In Figure 1.1 we see the initial MST. Then in Figure 1.2 the edges in bold are added as a result of the **MST-Augmentation** algorithm. Finally in Figure 1.3 a Hamiltonian cycle is created according to the **TSP-APPROX** algorithm.

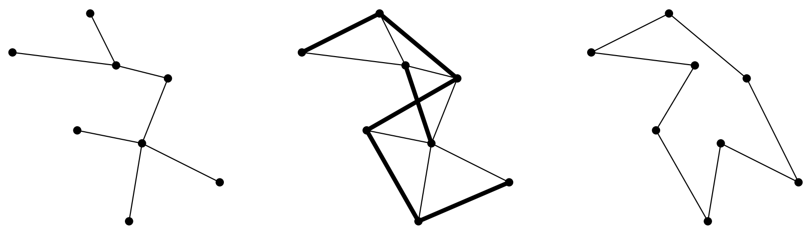
Power Assignment Stage (II)

For simplicity of notation, let us denote *clockwise* and *counterclockwise* traverses in the Hamiltonian cycle originating at node t by:

$$(t_{+i})_{i=0}^{n} = (t = t_{+0}, t_{+1}, \ldots, t_{+n} = t) \text{ and } (t_{-i})_{i=0}^{n} = (t = t_{-0}, t_{-1}, \ldots, t_{-n} = t)\,,$$

respectively. Note that $t_j = t_{n-j}$ for $0 \le j \le n$.

The same construction technique is used for every topology property. To form k node-disjoint paths between any two nodes we assign transmission powers in such a way so that there are $k/2$ node-disjoint paths in the clockwise and counterclockwise directions of the Hamiltonian cycle. This is achieved by assigning each node with sufficient power to reach $k/2$ neighboring nodes in both directions of the cycle. That is, node t_i is assigned $p(t_i) = (\Sigma_{j=0}^{k/2-1} d(t_{i+j}, t_{i+j+1}))^{\alpha}$. Let the resulting power assignment be A_k^h. Observe that in $H_{A_k^h}$ there are $k/2$ node-disjoint traverses in each direction of the cycle. For example, the j-th clockwise traverse originating in t_i is $(t_i, t_{i+j}, t_{i+j+1\cdot\frac{k}{2}}, t_{i+j+2\cdot\frac{k}{2}}, \ldots)$, where $1 \le j \le k/2$. In [41] we use a similar construction to provide k-strong connectivity.



1. MST

2. H_2 is 2-node connected

3. A Hamiltonian cycle h in the square of H_2

Figure 1: Hamiltonian Cycle Stage

2.2 Analysis

In our analysis we make use of several lemmas and theorems. The first two are defined in [7] and [13], respectively.[1]

Theorem 2.1 (Theorem 1 [7])**.** *For the relaxed triangle inequality parameter r, the algorithm* ***TSP-APPROX*** *finds a Hamiltonian cycle h of cost $C_h \le 4rC_{H_2}$.*

Lemma 2.2 (Lemma 11 [13])**.** *Let $\mathcal{E}_1$ be the set of all edges of* MST *incident on leaves. Let $\mathcal{E}_2$ be the set of all edges of the trees* MST_v *for all non-leaf nodes $v \in \mathcal{T}$. For $H_2' = (\mathcal{T}, \mathcal{E}_1 \cup \mathcal{E}_2)$ and $\alpha = 2$ it holds $C_{H_2'} \le 4C_{\text{MST}}$.*

Lemma 2.3. *For $\alpha = 2$, $C_h \in O(C_{\text{MST}})$.*

Proof. From the construction of H_2' we can see that $C_{H_2} \le C_{H_2'} + C_{\text{MST}}$. According to Lemma 2.2 $C_{H_2'} \le 4C_{\text{MST}}$ and therefore $C_{H_2} \le 5C_{\text{MST}}$. For $\alpha = 2$, the relaxed triangle inequality parameter is $r = 2$. According to Theorem 2.1 $C_h \le 8C_{H_2} \in O(C_{\text{MST}})$. ∎

In [41] we prove that the strong connectivity property is k-fault resistant in $H_{A_k^h}$ and derive the following cost bound.

Lemma 2.4 ([41])**.** $C_{A_k^h} \in O(k^2 C_h)$.

Now we are ready to present our main result.

Theorem 2.5. *Suppose $\mathcal{T}$ is a set of n nodes in the plane, $\mathcal{P}$ is a topology property and $k > 1$ is the fault resistance parameter. Let A_k^* be the optimal solution for the* Topology-Control-Problem*($\mathcal{T}$,$\mathcal{P}$,k). If $C_{A_k^*} \in \Omega(C_{\text{MST}})$ and $\mathcal{P}$ is weaker than the strong connectivity property then $C_{A_k^h} \in O(k^2 C_{A_k^*})$ and $\mathcal{P}$ is k-fault resistant in $H_{A_k^h}$.*

[1]We altered the description of every external lemma or theorem to match the model and definitions of the paper.

Proof. It is easy to see that $\mathcal{P}$ is k-fault resistant in $H_{A_k^h}$. This is due to the fact that the strong connectivity property is k-fault resistant in $H_{A_k^h}$. From Lemmas 2.3 and 2.4 we have $C_h \in O(C_{\text{MST}})$ and $C_{A_k^h} \in O(k^2 C_h)$, respectively. Given $C_{A_k^*} \in \Omega(C_{\text{MST}})$ we can conclude $C_{A_k} \in O(k^2 C_{A_k^*})$. ∎

We show that by using the framework described above we can derive an approximation bound of $O(k^2)$ for each of the three topology control problems, namely k-multicast, k-broadcast and k-convergecast. For each problem we first describe the power assignment and then prove the approximation ratio. Since all three problems have a root node $r \in \mathcal{T}$, we will use clockwise $((r_{+i})_{i=0}^{n})$ and counterclockwise $((r_{-i})_{i=0}^{n})$ traverses in the Hamiltonian cycle originating in r. Figure 2 demonstrates the power assignment stage (II) for the three topologies with $k = 2$. The arrows in the graph denote the transmission traffic we wish to obtain.

2.2.1 k-Broadcast

First we obtain the power assignment A_k^B which induces a broadcast tree rooted at node $r \in \mathcal{T}$ and there are k node-disjoint paths from r to any other node.

> **k-Broadcast($\mathcal{T}$,k,r)** — *Take the Hamiltonian cycle h according to stage I of the framework. Then, follow the construction in stage II of the framework and form $k/2$ disjoint paths from r to $r_{+(n-1)}$ in the clockwise direction of the cycle and $k/2$ disjoint paths from r to $r_{-(n-1)}$ in the counterclockwise direction of the cycle.*

Next we analyze the approximation ratio of the algorithm. We make use of the following lemma from [44] to obtain a lower bound for the optimal solution A_1^* for the 1-broadcast problem.

Lemma 2.6 (Lemma 4 [44])**.** *For any node set $\mathcal{T}$ in the plane, the total energy required by broadcasting from any node in $\mathcal{T}$ is at least $\frac{1}{c} C_{\text{MST}}$, where $6 \le c \le 12$.*

The next theorem is now easily derived.

Theorem 2.7. *Given a set of nodes T in the plane, a root node $r \in \mathcal{T}$ and a fault resistance parameter k, the power assignment A_k^B induces a k-broadcasting communication graph from r and $C_{A_k^B} \in O(k^2 C_{A_k^*})$, where A_k^* is the optimal power assignment for the* Topology-Control-Problem*($\mathcal{T}$,***broadcast from** *r,k).*

Proof. Since there are $k/2$ node-disjoint paths from r to any other node in the clockwise direction of the cycle and $k/2$ node-disjoint paths in the counterclockwise direction we can conclude that the induced communication graph is k-broadcasting (for example, in Figure 2.1, there is one path from r to any other node in clockwise and counterclockwise direction). From Lemma 2.6 we immediately conclude $C_{A_1^*} \in \Omega(C_{\text{MST}})$ and therefore, $C_{A_k^*} \in \Omega(C_{\text{MST}})$. Since A_k^B is cheaper than A_k^h, the approximation ratio follows immediately Theorem 2.5. ∎

2.2.2 k-Multicast

Let us describe the power assignment A_k^M which induces a multicast tree rooted at node $r \in \mathcal{T}$ and there are k node-disjoint paths from r to any node in M. We will use an approximation algorithm presented in [45]. The authors developed a heuristic called **SPF** (shortest-path first), which finds a multicast tree in an asymmetric network model such that the total transmission power is no more than 24 times the optimum.

> **k-Multicast($\mathcal{T}$,k,r,M)** — Let $\mathcal{S} \subseteq \mathcal{T}$ be a set of nodes reachable from r as a result of using **SPF**. We construct a Hamiltonian cycle $h_{\mathcal{S}}$ traversing all nodes in $\mathcal{S}$ according to stage I of the framework. Let f and l, where $1 \le f \le l \le |\mathcal{S}| + 1$, be the indexes of the first and last nodes in M as they appear in the Hamiltonian cycle traverse $\big(r = r_{+0}, r_{+1}, \ldots, r_{+(n-1)}, r_{+n} = r\big)$. That is $r_{+f}, r_{+l} \in M$ and $\forall i, 0 < i < f$ or $l < i < n : r_{+i} \notin M$. Next we follow the construction in stage II of the framework for the node set $\mathcal{S}$ and form $k/2$ disjoint paths from r to r_{+l} in the clockwise direction of the cycle and $k/2$ disjoint paths from r to $r_{-(n-f)}$ in the opposite direction.

The next theorem proves the correctness of the assignment A_k^M and the approximation ratio of $O(k^2)$ times the optimum for the k-multicast problem.

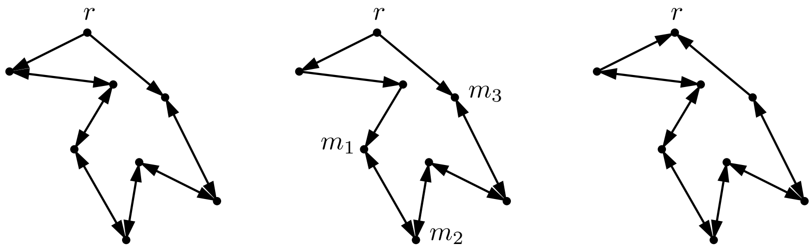


Figure 2: Power Assignment Stage ($k = 2$)

Theorem 2.8. *Given a set of nodes T in the plane, root node $r \in \mathcal{T}$, a multicast destination set M and a fault resistance parameter k, the power assignment A_k^M induces a k-multicasting communication graph from r to M and $C_{A_k^M} \in O(k^2 C_{A_k^*})$, where A_k^* is the optimal power assignment for the* Topology-Control-Problem*($\mathcal{T}$,**multicast from** r,k).*

Proof. In a similar way, like the broadcast case, there are $k/2$ node-disjoint paths in both the clockwise and counterclockwise direction from r to every node in M (for example, in Figure 2.2 there is one path in both directions from r to all nodes in M). Let A_1 be the power assignment produced by **SPF** ([45]) so that $C_{A_1} \leq 24 C_{A_1^*}$, where A_1^* is the optimal power assignment for 1-multicasting from r. Let $\text{MST}_\mathcal{S}$ be the minimum spanning tree of $\mathcal{S}$ and $B_1^*(\mathcal{S})$ be the optimal power assignment for 1-broadcasting from r to nodes in $\mathcal{S}$. Obviously $C_{A_1} \geq C_{B_1^*(\mathcal{S})}$. From Lemma 2.6 it is easy to see that $C_{B_1^*(\mathcal{S})} \in \Omega(C_{\text{MST}_\mathcal{S}})$ and therefore, $C_{A_k^*} \in \Omega(C_{\text{MST}_\mathcal{S}})$. Let $A_k^h(\mathcal{S})$ be the power assignment after applying our framework on the node set $\mathcal{S}$. Clearly, $C_{A_k^M} \leq C_{A_k^h(\mathcal{S})}$. Due to Theorem 2.5 $C_{A_k^M(\mathcal{S})} \in O(k^2 C_{A_k^*})$. ∎

2.2.3 k-Convergecast

A similar technique is proposed in the case of k-convergecast. The power assignment A_k^C, which forms a k-convergecast tree to some root node r will have k node-disjoint paths from every node to r. We first describe the construction and then analyze the approximation ratio.

k-Convergecast($\mathcal{T}$,k,r) — Take the Hamiltonian cycle h according to stage I of the framework. We follow the construction in stage II of the framework and form $k/2$ disjoint paths from every r_{+i} to r in the clockwise direction of the cycle and $k/2$ disjoint paths in the counterclockwise direction.

In order to obtain the approximation ratio of $O(k^2)$ we first prove the following bound.

Lemma 2.9. *Given a set of nodes $\mathcal{T}$ in the plane and a root node $r \in \mathcal{T}$, the total energy required by convergecasting from every node to r is at least C_{MST}.*

Proof. Let A_1^* be the optimal power assignment so that $H_{A_1^*}$ is 1-convergecasting to r. Remove edges from $H_{A_1^*}$, leaving only paths towards the root, so there is a unique path from every node to r. Let the resulting directed tree be ST. Clearly, $C_{ST} \leq C_{A^*}$, where $C_{ST} = \Sigma_{(s,t)\in ST} d(s,t)^\alpha$. It is easy to see that $C_{\text{MST}} \leq C_{ST}$, since ST is a tree. As a result, $C_{\text{MST}} \leq C_{A^*}$. ∎

The following theorem easily follows.

Theorem 2.10. *Given a set of nodes T in the plane, a root node $r \in \mathcal{T}$ and a fault resistance parameter k, the power assignment A_k^C induces a k-convergecasting communication graph to r and $C_{A_k^C} \in O(k^2 C_{A_k^*})$, where A_k^* is the optimal power assignment for the* Topology-Control-Problem*($\mathcal{T}$,**convergecast to** r,k).*

Proof. There are $k/2$ node-disjoint paths in both directions of the cycle, from any node to r (for example, in Figure 2.3 there is one path in both directions from every node to r). Due to Lemma 2.9 it is easy to see that $C_{A_k^C} \in \Omega(C_{\text{MST}})$. Since A_k^C is cheaper than A_k^h, the approximation ratio follows immediately Theorem 2.5. ∎

3 Linear k-Fault Resistance

In this section we present some interesting results for k-fault resistance in various topology control problems for the linear case. In this case we only need to consider the distances between adjacent transceivers rather than between any pair of transceivers. Let $d_i = d(t_i, t_{i+1})$, $1 \leq i \leq n-1$.

We will use our definitions from [41]. Let $d^L_{i,k} = d(t_i, t_{i-k})$ and $d^R_{i,k} = d(t_i, t_{i+k})$. In case that $i \leq k$ or $i > n-k$, $d^L_{i,k} = d(t_i, t_1)$ and $d^R_{i,k} = d(t_i, t_n)$, respectively. In [41] we obtained k-strong connectivity by assigning each node $t_i \in \mathcal{T}$ the power value $p(t_i) = \max\{(d^L_{i,k})^\alpha, (d^R_{i,k})^\alpha\}$. We showed that this power assignment forms k node-disjoint paths from each node in both directions of the line (right and left).

We start by constructing a linear k-broadcast. Then we explain how this construction can be used for the linear k-multicast problem. In the end we present an optimal power assignment construction for the linear k-convergecast problem.

3.1 Multicast and Broadcast

First, note that for $r = t_1$ or $r = t_n$ the following simple algorithm results in an optimal power assignment for the linear k-broadcast problem.

One-Sided-Linear-k-Broadcast($\{t_1, t_2, \ldots, t_n\}$,$k$,$d$) — *Let d be the direction of the desired broadcast. For broadcasting from t_1 or t_n it would be right or left, respectively. Without loss of generality suppose d is right, that is $r = t_1$. Let A^j_k, for $k < j < n$ be a power assignment so that: $p(t_1) = d(t_1, t_j)^\alpha$ and $p(t_i) = (d^R_{i,k})^\alpha$, for $j - k + 1 \leq i \leq n$. Let A^n_k be a power assignment so that $p(t_1) = d(t_1, t_n)^\alpha$. The desired power assignment is the cheapest A^j_k, for $k < j \leq n$.*

This algorithm first decides on the power assignment of t_1. If t_1 does not reach t_n, then the algorithm assigns to each node, starting from t_{j-k+1}, enough power to reach k nodes to its right. Otherwise, t_1 is assigned to reach all nodes in a single hop and the rest of the nodes are not assigned any power. In both cases the induced communication graph is k-broadcasting from t_1. Note that if t_1 is assigned $p(t_1) < d(t_1, t_{k+1})^\alpha$, then k-broadcast is impossible.

The cheapest of the possible $n-k$ power assignments, $A^{k+1}_n, \ldots, A^n_k$, is optimal since once t_j is determined, the rest of the assignment is straightforward. We can immediately derive an approximation ratio of 2 for the linear k-broadcast problem by using **One-Sided-Linear-k-Broadcast** in both directions from $r = t_i$.

Linear-k-Broadcast($\{t_1, t_2, \ldots, t_n\}$,$k$,$i$) — *Let $r = t_i$ and let A_L and A_R be the power assignments as a result of applying* **One-Sided-Linear-k-Broadcast**($\{t_1, \ldots, t_i\}, k, left$) *and* **One-Sided-Linear-k-Broadcast**($\{t_i, \ldots, t_n\}, k, right$)*, respectively. Let p^L_j and p^R_j be the power assignment of t_j in A_L and A_R, respectively. To form a power assignment A which produces a k-broadcasting graph from t_i, assign each node t_j with $p(t_j) = \max\{p^L_j, p^R_j\}$.*

It is easy to see that the algorithm is correct (we preserve left and right k-broadcast from t_i). The next Lemma proves the approximation ratio of 2 times the optimum.

Lemma 3.1. *Given a set of nodes $\mathcal{T}$ positioned on a line, a root node $r = t_i$ and a fault resistance parameter $k > 1$. The* **Linear-k-Broadcast** *algorithm produces a power assignment with an approximation ratio 2 times the optimum.*

Proof. Let A^* be the optimal power assignment for the linear k-broadcast problem. Clearly $C_{A^*} \geq C_{A_L}$ and $C_{A^*} \geq C_{A_R}$. Since $C_A \leq C_{A_L} + C_{A_R}$ we conclude $C_A \leq 2C_{A^*}$. ∎

For the case of linear k-multicast from some root node $r = t_i$ to a set of nodes $M \subseteq \mathcal{T}$, let t_l and t_r be the leftmost the rightmost nodes in M, respectively. The problem of linear k-multicast can be reduced to the problem of linear k-broadcast from r to $\{t_{\min\{l,i\}}, \ldots, t_i, \ldots, t_{\max\{i,r\}}\}$. This is because in the linear case, constructing k node-disjoint paths from node s to node t automatically forms k node-disjoint paths from s to any node between s and t. As a result we obtain the approximation factor of 2.

3.2 Convergecast

The linear k-convergecast problem is solved optimally by the next simple algorithm, which is followed by a Lemma proving its correctness and optimality.

Linear-k-Convergecast($\{t_1, t_2, \ldots, t_n\}$,$k$,$i$) — *Each node $t_j \neq t_i$ is assigned,*

$$p(t_j) = \begin{cases} d(t_j, t_{j+k})^\alpha & \text{for } 1 \leq j < i - k \\ d(t_j, t_i)^\alpha & \text{for } i - k \leq j \leq i + k \\ d(t_j, t_{j-k})^\alpha & \text{for } i + k < j \leq n \end{cases}$$

Lemma 3.2. *Given a set of nodes $\mathcal{T}$ positioned on a line, a root node $r = t_i$ and a fault resistance parameter $k > 1$. The* **Linear-k-Convergecast** *algorithm produces an optimal power assignment for the linear k-convergecast.*

Proof. The algorithm is correct since each node either reaches r using one hop, or there are k nodes it can reach in a single hop in the direction of t_i. As a result the power assignment creates a k-convergecasting communication graph to r. Next we prove the optimality of the proposed power assignment. It is easy to see that the algorithm is optimal if $r = t_1$ or $r = t_n$. We treat both sections of the line (to the left and to the right of t_i) as it were a separate line. Each section is solved independently and has no influence on the other section. Therefore combining two sections yields a complete and optimal solution for the linear k-convergecast problem. ■

4 Dynamic k-strong connectivity

In this section we present an efficient dynamic scheme for maintaining k-strong connectivity in both $\mathbb{R}^1$ and $\mathbb{R}^2$. We extend our static algorithms in [15] and [41], for linear and planar cases, respectively, and make them dynamic.

4.1 Linear dynamic k-strong connectivity

In [41], the static algorithm for linear k-strong connectivity assigns each node $t \in \mathcal{T}$ with sufficient power to reach k nodes to its right and k nodes to its left. Therefore, in order to compute the power assignment of node t_i we need to know the distance to k-th node to its left and k-th node to its right, that is $d^L_{i,k}$ and $d^R_{i,k}$, respectively. We maintain a dynamic data structure that supports insert/delete/query operations in $O(\log n)$ time. In other words, we dynamically maintain the set of points and efficiently answer the following query: <*Given point index i, determine the power of t_i in the near optimal power assignment described in [15]*>. Moreover, only relative location is needed and not the exact coordinates of each node.

We use a standard balanced binary search tree implemented as a red-black tree [26]. Each node t is represented by a node in the red-black tree and holds the following information: (a) the number n_t of nodes in a subtree rooted at t and (b) for each child t_c, the distance $d(t, t_c)$. The nodes in the tree are ordered in accordance to their relative positions. The nodes are added or deleted according to the red-black tree operations. We start by describing the insert/delete operations and then address the query operation.

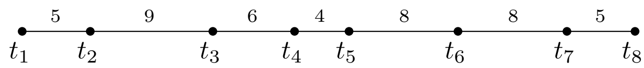
Insert and Delete — A new node t is inserted or deleted according to the red-black insert/delete operations which take $O(\log n)$ time. The update of node internal information takes place along with the insert/delete of the node along the path from root to t.

Query — Given some node t_i, we need to compute its power assignment by computing $d^L_{i,k}$ and $d^R_{i,k}$. Without loss of generality we describe the computation of $d^R_{i,k}$. First we locate the node t_i which takes $O(\log n)$ time. Then we traverse the tree starting at t_i in order to find a node k places distant from t_i to the right. Since every node t holds the number of nodes in a subtree rooted at t we can find t_{i+k} in $O(\log n)$ time by a simple binary search starting at t_i. Once we have located t_{i+k} we compute the value $d^R_{i,k}$ as follows[2]. Let u be the least common ancestor of t_i and t_{i+k}. We compute the distance between t_i and t_{i+k} by following the path from u to t_i and t_{i+k}. Recall that each node stores the distance to every child. When following a path from u to t_i, for every left child we add the distance and for every

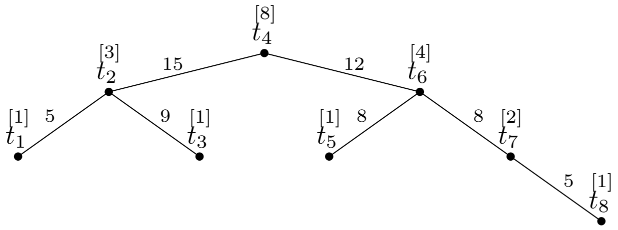
[2]Note that if $i + k > n$ then $d^R_{i,k} = d(t_i, t_n)$, which matches the definition.

right child we subtract the distance. When following the path from u to t_{i+k} we do the same, only this time adding on a right child and subtracting on left. The value $d^R_{i,k}$ is obtained by adding the distances computed for both paths. As a result the total time it takes to answer a power assignment query for any node is $O(\log n)$.

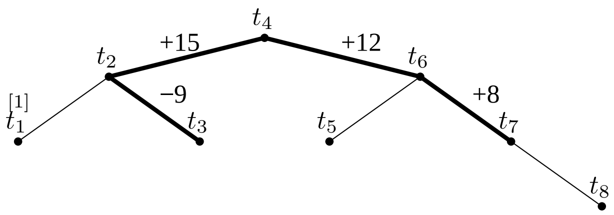
For example, in Figure 3.1 there are 8 nodes positioned on a single line. The corresponding red-black tree is presented in Figure 3.2. Each node on the line is represented by a node in the tree. For each node t_i the value in square brackets is n_{t_i}. For every edge (t_i, t_j) the distance $d(t_i, t_j)$ is represented by a value along that edge. In Figure 3.3 the calculation of $d^R_{3,4}$ is presented. That is we look for the distance from t_3 to t_7. The least common ancestor of t_3 and t_7 is t_4. We compute as described earlier, $d(t_3, t_4) = 15 - 9 = 6$ and $d(t_4, t_7) = 12 + 8 = 20$. As a result $d^R_{3,4} = d(t_3, t_4) + d(t_4, t_7) = 6 + 20 = 26$.



1. Linear layout



2. Red-Black tree



3. Calculation of $d^R_{3,4}$

Figure 3: Dynamic linear k-connectivity maintenance

4.2 Planar dynamic k-strong connectivity

We first present the static algorithm used to obtain planar k-strong connectivity and then describe the dynamic scheme built on top of it.

Planar k-strong connectivity (Carmi et al. [15])

Let $\mathcal{T}$ be a set of n points in the plane (representing n transceivers). For each node $t \in \mathcal{T}$, let $N_t \subseteq \mathcal{T}$ be a set of k-closest nodes to t, and let $r^*_t = \max_{t' \in N_t} d(t, t')$. We now describe the range assignment algorithm. Compute a minimum spanning tree MST of the Euclidean graph induced by $\mathcal{T}$. Assign to each node $t \in \mathcal{T}$ the range r^*_t. Denote this initial range assignment by A'. For each edge $e = (t, s)$ of MST, increase the range of the nodes in $N_t \cup N_s$ (if necessary), such that each node $t' \in N_t$ can reach all nodes in $N_s \cup \{s\}$, and vice versa. Let A_k denote the resulting range assignment.

The idea is rather simple, we construct k node-disjoint paths along the edges of the MST. Think about each N_t as a large intersections containing k intersection points, and there are k symmetric links between N_t and N_s iff (t, s) is an edge in the MST. The range assignment of each node t must be at least r^*_t (otherwise k-strong connectivity is impossible), and in addition sufficient enough to create the intersections mentioned above. We obtain an approximation ratio of $O(k)$ times the optimum under the L_2 metric in $O(n \log n)$ time.

Dynamic k-closest nodes

The algorithm heavily depends on computing N_t for every node t. That is, for every node we want to be able to find its k-closest nodes in the plane. In order to maintain the topology property of k-strong connectivity in dynamic settings, where nodes can be added or removed, we need to maintain a dynamic data structure which allows a cheap k-closest nodes query with every update operation. Next we describe such a data structure with a query in $O(\log^2 n)$ time under the L_∞ norm. The preprocessing time is $O(n \log n)$. Later we discuss the implication of using the L_∞ norm and its effect on the overall approximation factor.

Below we present a scheme that computes for a given point t, the k nearest points in the plane under the L_∞ metric. In other words we aim to find the smallest axis-parallel square centered at k that contains exactly k points. We perform a binary search over the sorted values of x-and y-coordinates of points, in order to determine the radius of square centered at p. Once we have fixed the radius of the square, we ask how many points are inside of it. The idea is to apply standard orthogonal range searching with fractional cascading technique (see Willard and Lueker [48]), that allows one to count the number of points in a given query axis-parallel rectangle in $O(\log n)$ time after $O(n \log n)$ preprocessing. This can be generalized to deal with updates in $O(\log^2 n)$ time and reporting points in time $O(k + \log^2 n)$. If the number is exactly k, we found all the k-nearest neighbors of p. Otherwise, we increase or decrease the radius of square depending on the number of points inside of square. Finally, after we find a square with k points we can actually report them using a dynamic orthogonal range tree augmented for reporting (not counting) the points inside of given axis-parallel rectangle. To answer a range reporting query we simply traverse the subtrees determined by the searching procedure, reporting the points of their leaves. Each subtree can be traversed in time proportional to the number of leaves it contains. The query time is $O(k + \log^2 n)$.

Using the L_∞ metric worsens the original $O(k)$ approximation ratio of static k-strong connectivity by $\sqrt{2}$ factor as stated in the next observation.

Observation 4.1. *For any node $t \in \mathcal{T}$, let N_t^∞ and N_t^2 be the sets of its k-closest nodes computed in the L_∞ and L_2 metrics, respectively. Then $\max_{t' \in N_t^\infty} d(t, t') \leq \sqrt{2} \max_{t' \in N_t^2} d(t, t')$.*

We will now show that node t' can be one of the k-closest nodes for at most $8k$ other nodes in L_∞. This will mean that adding or removing a single node will not change more than $O(k)$ k-closest node sets, that is the number of all sets N_t so that $t' \in N_t$ is $O(k)$. Let $RN_{t'}^\infty$ be a set of nodes which have t' as their k-closest neighbor in L_∞. That is, if $t' \in N_t^\infty$ then $t \in RN_{t'}^\infty$, where N_t^∞ is as defined in Observation 4.1. We prove that $|RN_{t'}^\infty| \leq 8k$, for any $t' \in \mathcal{T}$. We will need the following technical lemma.

Lemma 4.2. *Let P be a set of points in the plane. For any point q in the plane, let $X \subseteq P$ be a set of points that have point q as their nearest neighbor in $P \cup \{q\}$ under the L_∞ metric. Then $|X| \leq 8$.*

Proof. Without loss of generality, assume that q is the origin, and partition the plane into eight wedges by the four lines $y = 0$, $x = 0$, $y = x$ and $y = -x$, so that each wedge is open on its clockwise side and close on its counterclockwise side. We claim that each of these wedges can contain at most one point $a \in P$ whose nearest $L_\infty-$neighbor in $P \cup \{q\}$ is q. Without loss of generality, consider the wedge W_1, given by

$$W_1 = \{(x, y) \mid x \geq 0 \text{ and } 0 < y \leq x\}.$$

Suppose that $|P \cap W_1| \geq 2$ (otherwise the claim is obvious), and let a be the leftmost point in $P \cap W_1$ (see Figure 4). If there is more than one such point, choose a to be the lowest among them. Let b be any other point of $S \cap W_1$ and $d_\infty(a, b)$ be the distance between points a and b under the L_∞ metric.[3] Then $d_\infty(b, a) < d_\infty(b, q)$. Indeed, $d_\infty(b, q) = b_x$ and $d_\infty(b, a) = \max\{b_x - a_x, |b_y - a_y|\}$ Clearly, $b_x - a_x < b_x$. If $b_y < a_y$ then $|b_y - a_y| = b_y - a_y < b_y \leq b_x$; otherwise $|b_y - a_y| = a_y - b_y < a_y \leq a_x \leq b_x$. Hence, in both cases $d_\infty(b, a) < d_\infty(b, q)$, as claimed. That is, a is the only one point of $P \cap W_1$ in X. Since we have eight wedges, $|X| \leq 8$. ∎

The same argument in the proof can be applied when analyzing the maximal number of points for which point q is one of the k-nearest points. We show that it is possible to have at most k points in each wedge. The following observation can be easily derived then.

[3]The x and y coordinates of point p are given by p_x and p_y, respectively. The distance under the L_∞ metric is defined as $d_\infty(a, b) = \max\{a_x - b_x, b_x - b_y\}$.

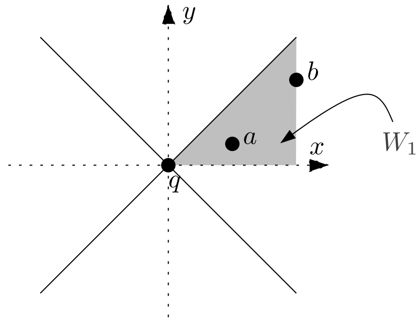


Figure 4: Dividing the plane into eight wedges.

Observation 4.3. *Let P be a set of points in the plane. For any point q in the plane, let $X_k \subseteq P$ be a set of points that have point q as their one of their k-nearest neighbors in $P \cup \{q\}$ under the L_∞ metric. Then $|X_k| \leq 8k$.*

The following lemma is easily derived from Observation 4.3.

Lemma 4.4. $\forall t' \in \mathcal{T} : |RN_{t'}^{\infty}| \leq 8k$.

Dynamic MST

We maintain the MST under insert and delete operations by using the results provided by Chan in [17]. The author uses a reduction to bichromatic closest pairs in order to maintain the Euclidean MST of a dynamic 2-d point set in $O(\log^{10} n)$ expected amortized time.

Observation 4.5. *Let $\mathcal{T}$ be a set of n points in the plane. Let $\text{MST}_{\mathcal{T}} = (\mathcal{T}, E)$ be a Euclidean minimum spanning tree over $\mathcal{T}$, with E being the set of edges. A point p removal or addition to $\mathcal{T}$ changes at most $O(1)$ edges in E.*

Proof. Suppose a point is deleted from $\mathcal{T}$. Since in $\text{MST}_{\mathcal{T}}$ the degree of each node is at most 6, the deletion of a point may result in at most 6 unconnected components. To form a new MST at most 5 new edges are needed. As a result there are at most 6 edges removed and 5 added to E when forming a new MST. Now suppose a point is added to $\mathcal{T}$. In the resulting MST this point might have at most 6 neighbors. We cannot simply connect the point because we might have some cycles. There might be at most $\binom{6}{2}$ cycles created as a result of adding a point. What we need to do is connect the point and then resolve the cycles by deleting the edge with the largest cost from each cycle. If we compare the original MST with the resulting one, at most 6 edges might be added and at most $\binom{6}{2}$ removed. ∎

The observation above shows that an addition or removal of a node affects only a constant number of nodes with respect to their MST neighbors. Recall that the power assignment for k-strong connectivity is to create k symmetric links between N_t and N_s for every edge $(s, t) \in$ MST. Therefore, each addition or removal of a node will force power assignment changes according to the changes in the MST. Next we focus on insert/delete operations.

Insert — When a new node t is introduced into the system we need to: (a) compute its k-closest nodes, (b) recompute r_w^* of every node w, for which $t \in N_w$, (c) update MST and (d) update the assignment alongside the MST edges. We focus on each of the steps:

(a) Given some point p it takes $O(\log^2 n)$ time to find its k-closest nodes under the L_1 norm.

(b) Due to Lemma 4.4 there are at most $8k$ nodes for which we need to recalculate their k-closest nodes, which takes $O(k \log^2 n)$ time.

(c) The MST update takes $O(\log^{10} n)$ expected amortized time according to [17].

(d) When a new node is added, at most $O(1)$ edges are altered. Each MST edge addition/removal affects $O(k)$ of nodes that need to change their range assignment.

To summarize we have an expected amortized $O(k \log^2 n + \log^{10} n)$ update time in case of adding a new node to the system.

Delete — When we remove a node $t \in \mathcal{T}$ we need to do the same as for the insert operation except (a). We recalculate k-closest nodes of all nodes w for which $t \in N_w$, update the MST and the assignment alongside its edges. As before, the expected amortized node removal time would be $O(k \log^2 n + \log^{10} n)$.

5 Conclusions

In this paper we have presented a number of results regarding k-connectivity power assignment problems in wireless networks. In particular we give a general framework for producing provable approximation factor solutions for various topologies as well as their dynamic counterparts.

One of possible interesting directions is to generate the obtained results for bounded-h-hop topologies, i.e when each formed path should contain no more than h hops.

References

[1] E. Althaus, G. Calinescu, I. Mandoiu, S. Prasad, N. Tchervenski, and A. Zelikovsky. Power efficient range assignment in ad-hoc wireless networks. In *WCNC'03*, pages 1889–1894, 2003.

[2] C. Ambühl. An optimal bound for the MST algorithm to compute energy efficient broadcast trees in wireless networks. In *ICALP'05*, volume 3580, pages 1139–1150, 2005.

[3] C. Ambühl, A. Clementi, M. D. Ianni, A. Monti, G. Rossi, and R. Silvestri. The range assignment problem in non-homogeneous static ad-hoc networks. In *IPDPS'04*, 2004.

[4] C. Ambühl, A. Clementi, P. Penna, G. Rossi, and R. Silvestri. Energy consumption in radio networks: Selfish agents and rewarding mechanisms. *Theoretical Computer Science*, 343(1-2):27–41, 2004.

[5] V. Annamalai, S. K. Gupta, and L. Schwiebert. On tree-based convergecasting in wireless sensor networks. In *WCNC'03*, volume 3, pages 1942–1947, 2003.

[6] S. Banerjee, A. Misra, J. Yeo, and A. Agarwala. Energy-efficient broadcast and multicast trees for reliable wireless communication. In *WCNC'03*, volume 1, pages 660–667, 2003.

[7] M. A. Bender and C. Chekuri. Performance guarantees for the TSP with a parameterized triangle inequality. *Information Processing Letters*, 73(1–2):17–21, 2000.

[8] F. Bian, A. Goel, C. S. Raghavendra, and X. Li. Energy-efficient broadcasting in wireless ad hoc networks lower bounds and algorithms. *Journal of Interconnection Networks*, 3(3-4):149–166, 2002.

[9] D. M. Blough, M. Leoncini, G. Resta, and P. Santi. On the symmetric range assignment problem in wireless ad hoc networks. In *IFIP TCS'02*, pages 71–82, 2002.

[10] M. Cagalj, J.-P. Hubaux, and C. C. Enz. Minimum-energy broadcast in all-wireless networks: : Np-completeness and distribution issues. In *MobiCom'02*, pages 172–182, 2002.

[11] G. Calinescu, S. Kapoor, A. Olhevsky, and A. Zelikovsky. Network lifetime and power assignment in ad-hoc wireless networks. In *ESA'03*, pages 114–126, 2003.

[12] G. Calinescu, I. I. Mandoiu, and A. Zelikovsky. Symmetric connectivity with minimum power consumption in radio networks. In *IFIP TCS'02*, pages 119–130, 2002.

[13] G. Calinescu and P.-J. Wan. Range assignment for high connectivity in wireless ad hoc networks. In *AdHoc-NOW'03*, pages 235–246, 2003.

[14] I. Caragiannis, C. Kaklamanis, and P. Kanellopoulos. New results for energy-efficient broadcasting in wireless networks. In *ISAAC'02*, pages 332–343, 2002.

[15] P. Carmi, M. Segal, M. J. Katz, and H. Shpungin. Fault-tolerant power assignment and backbone in wireless networks. In *FAWN'06*, pages 80–84, 2006.

[16] J. Cartigny, D. Simplot, and I. Stojmenovic. Localized minimum-energy broadcasting in ad-hoc networks. In *INFOCOM'03*, 2003.

[17] T. M. Chan. A dynamic data structure for 3-d convex hulls and 2-d nearest neighbor queries. In *SODA'06*, pages 1196–1202, 2006.

[18] J. Cheriyan, S. Vempala, and A. Vetta. Approximation algorithms for minimum-cost k-vertex connected subgraphs. In *STOC'02*, pages 306–312, 2002.

[19] I. Chlamtac and S. Kutten. Tree-based broadcasting in multihop radio networks. *IEEE Transactions on Computers*, 36(10):1209–1223, 1987.

[20] A. Clementi, G. Huiban, P. Penna, G. Rossi, and Y. Verhoeven. Some recent theoretical advances and open questions on energy consumption in ad-hoc wireless networks. In *ARACNE'02*, pages 23–38, 2002.

[21] A. E. F. Clementi, A. Ferreira, P. Penna, S. Perennes, and R. Silvestri. The minimum range assignment problem on linear radio networks. In *ESA'00*, pages 143–154, 2000.

[22] A. E. F. Clementi, P. Penna, and R. Silvestri. Hardness results for the power range assignmet problem in packet radio networks. In *RANDOM-APPROX'99*, pages 197–208, 1999.

[23] A. E. F. Clementi, P. Penna, and R. Silvestri. On the power assignment problem in radio networks. *Electronic Colloquium on Computational Complexity*, 7(054), 2000.

[24] A. E. F. Clementi, P. Penna, and R. Silvestri. On the power assignment problem in radio networks. *Electronic Colloquium on Computational Complexity*, 7(54), 2000.

[25] A. E. F. Clementi, P. Penna, and R. Silvestri. The power range assignment problem in radio networks on the plane. In *STACS'00*, pages 651–660, 2000.

[26] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. The MIT Press, 2nd edition, 2001.

[27] M. Flammini, A. Navarra, R. Klasing, and S. Pérennes. Improved approximation results for the minimum energy broadcasting problem. In *DIALM-POMC'04*, pages 85–91, 2004.

[28] M. T. Hajiaghayi, N. Immorlica, and V. S. Mirrokni. Power optimization in fault-tolerant topology control algorithms for wireless multi-hop networks. In *MobiCom'03*, pages 300–312, 2003.

[29] M. T. Hajiaghayi, G. Kortsarz, V. S. Mirrokni, and Z. Nutov. Power optimization for connectivity problems. In *IPCO'05*, pages 349–361, 2005.

[30] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *HICSS'00*, 2000.

[31] X. Jia, D. Kim, S. Makki, P.-J. Wan, and C.-W. Yi. Power assignment for k-connectivity in wireless ad hoc networks. In *INFOCOM'05*, pages 2206–2211, 2005.

[32] L. M. Kirousis, E. Kranakis, D. Krizanc, and A. Pelc. Power consumption in packet radio networks. *Theoretical Computer Science*, 243(1–2):289–305, 2000.

[33] S. O. Krumke, R. Liu, E. L. Lloyd, M. V. Marathe, R. Ramanathan, and S. S. Ravi. Topology control problems under symmetric and asymmetric power thresholds. In *AdHoc-NOW'03*, pages 187–198, 2003.

[34] W. Liang. Constructing minimum-energy broadcast trees in wireless ad hoc networks. In *MobiHoc'02*, pages 112–122, 2002.

[35] W. Liang. Minimizing energy and maximizing network lifetime multicasting in wireless ad hoc networks. In *ICC'05*, volume 5, pages 3375–3379, 2005.

[36] S. Lindsey, C. S. Raghavendra, and K. M. Sivalingam. Data gathering algorithms in sensor networks using energy metrics. *IEEE Transactions on Parallel and Distributed Systems*, 13(9):924–935, 2002.

[37] E. L. Lloyd, R. Liu, M. V. Marathe, R. Ramanathan, and S. S. Ravi. Algorithmic aspects of topology control problems for ad hoc networks. *MONET*, 10(1-2):19–34, 2005.

[38] Z. Nutov. Approximating minimum power covers of intersecting families and directed connectivity problems. In *APPROX-RANDOM'06*, pages 236–247, 2006.

[39] K. Pahlavan and A. H. Levesque. *Wireless information networks*. Wiley-Interscience, 1995.

[40] R. Ramanathan and R. Hain. Topology control of multihop wireless networks using transmit power adjustment. In *INFOCOM'00*, pages 404–413, 2000.

[41] H. Shpungin and M. Segal. k-fault resistance in wireless ad-hoc networks. In *DIALM-POMC'05*, pages 89–96, 2005.

[42] A. Srinivas and E. Modiano. Minimum energy disjoint path routing in wireless ad-hoc networks. In *MobiCom'03*, pages 122–133, 2003.

[43] S. Upadhyayula, V. Annamalai, and S. K. S. Gupta. A low-latency and energy-efficient algorithm for convergecast in wireless sensor networks. In *GLOBECOM'03*, volume 22, pages 3525–3530, 2003.

[44] P.-J. Wan, G. Calinescu, X. Li, and O. Frieder. Minimum-energy broadcast routing in static ad hoc wireless networks. In *INFOCOM'01*, pages 1162–1171, 2001.

[45] P.-J. Wan, G. Călinescu, and C.-W. Yi. Minimum-power multicast routing in static ad hoc wireless networks. *IEEE/ACM Trans. Netw.*, 12(3):507–514, 2004.

[46] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides. On the construction of energy-efficient broadcast and multicast trees in wireless networks. In *INFOCOM'00*, pages 585–594, 2000.

[47] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides. Energy-efficient broadcast and multicast trees in wireless networks. *MONET*, 7(6):481–492, 2002.

[48] D. E. Willard and G. S. Lueker. Adding range restriction capability to dynamic data structures. *Journal of the ACM*, 32(3):597–617, 1985.