1-11-2016

Changing of the Guards: Strip Cover with Duty Cycling ⁂

Amotz Bar-Noy
City University of New York

Benjamin Baumer
Smith College, bbaumer@smith.edu

Dror Rawitz
Bar-Ilan University, Israel

Follow this and additional works at: https://scholarworks.smith.edu/mth_facpubs

Part of the Statistics and Probability Commons

Changing of the Guards:
Strip Cover with Duty Cycling[*]

Amotz Bar-Noy[†]
amotz@sci.brooklyn.cuny.edu

Ben Baumer[‡]
bbaumer@smith.edu

Dror Rawitz[§]
dror.rawitz@biu.ac.il

October 9, 2014

Abstract

The notion of *duty cycling* is common in problems which seek to maximize the lifetime of a wireless sensor network. In the duty cycling model, sensors are grouped into *shifts* that take turns covering the region in question, and each sensor can belong to at most one shift. We consider the imposition of the duty cycling model upon the STRIP COVER problem, where we are given n sensors on a one-dimensional region, and each shift can contain at most $k \leq n$ sensors. We call the problem of finding the optimal set of shifts so as to maximize the length of time that the entire region can be covered by a wireless sensor network, k-DUTY CYCLE STRIP COVER (k-DUTYSC). In this paper, we present a polynomial-time algorithm for 2-DUTYSC. Furthermore, we show that this algorithm is a $\frac{35}{24}$-approximation algorithm for k-DUTYSC. We also give two lower bounds on the performance of our algorithm: $\frac{15}{11}$, for $k \geq 4$, and $\frac{6}{5}$, for $k = 3$, and provide experimental evidence suggesting that these lower bounds are tight. Finally, we propose a fault tolerance model and find thresholds on the sensor failure rate over which our algorithm has the highest expected performance.

Keywords: approximation algorithms, duty cycling, wireless sensor networks, adjustable ranges, strip cover.

1   Introduction

We consider the following problem: Suppose we have a one-dimensional region (or interval) that we wish to cover with a wireless sensor network. We are given the locations of n sensors located on that interval, and each sensor is equipped with an identical battery of finite charge. We have the ability to set the sensing radius of each sensor, but its battery charge drains in inverse proportion to the radius that we set. Our goal is to organize the sensors into disjoint coverage groups (or

---

1

shifts), that will take turns covering the entire region for as long as possible. We call this length of time the *lifetime* of the network.

More specifically, we consider the STRIP COVER problem with identical batteries under a *duty cycling* restriction. An instance consists of a set $X \subseteq [0,1]$ of n sensor locations, and a rational number B representing the initial battery charge of each sensor. Each battery discharges in inverse linear proportion to its radius, so that a sensor i whose radius is set to r_i survives for B/r_i time. In the *duty cycling* model, the sensors are partitioned into disjoint coverage groups, called *shifts*, which take turns covering the entire interval for as long as their batteries allow. The sum of these lengths of time is called the *lifetime* of the network and is denoted by T. For any fixed $k \leq n$, the k-DUTY CYCLE STRIP COVER (k-DUTYSC) problem seeks an optimal partitioning of the sensors such that the network lifetime T is maximized, yet no coverage group contains more than k sensors. In the fault tolerant variant, each sensor may fail to activate with some fixed probability $p \in [0,1]$, and we seek to maximize the *expected* lifetime of the network (i.e., the expected sum of lifetimes of surviving shifts).

Motivation. Applications of scheduling problems similar to STRIP COVER are increasingly common. One such application involves monitoring a fence, or supply line, that exists in inhospitable territory. In this case, it may be feasible (even cost-effective) to deploy a set of sensors along the fence, but unfeasible to position them at pre-determined locations. For example, it might be easy to drop the senors from an airplane, but impossible to dispatch human beings to place them. While the scheduler may have access to the location of each sensor via GPS, technical limitations may require that a single assignment be given. In such a scenario, we might be incentivized to organize the sensors into disjoint shifts, providing motivation for our duty cycling model. Finally, any physical device will have some nonzero failure rate, and thus a fault-tolerant solution will be more robust.

Solutions to the general STRIP COVER problem contain both the radial assignments and activation and de-activation times for each sensors. As a result, these solutions can be complicated to implement and understand. Moreover, interdependence among multiple sensors can make such solutions susceptible to catastrophic decline in network lifetime if there is a non-zero probability of sensor failure. Conversely, since in the duty cycling model each sensor can participate in at most one cover shift, the scheduling of the shifts is of little importance. Furthermore, by minimizing the number of sensors participating in each shift, duty cycling solutions become more resilient to sensor failure.

Related work. This line of research began with Buchsbaum, et al.'s [5] study of the RESTRICTED STRIP COVER (RSC) problem. In RSC, the locations and sensing radii of n sensors placed on an interval are given, and the problem is to compute an optimal set of activation times, so as to maximize the network lifetime. They showed that RSC is NP-hard, and presented an $O(\log \log n)$-approximation algorithm. Gibson and Varadarajan [11] later improved on this result by discovering a constant factor approximation algorithm.

The problem of finding the optimal set of radial assignments for sensors deployed on an interval, rather than the activation times, is more tractable. Peleg and Lev-Tov [12] considered the problem of covering a finite set of m target points while minimizing the sum of the radii assigned, and found an optimal polynomial-time solution via dynamic programming. The situation wherein the whole

interval must be covered corresponds to a "one shift" version of n-DutySC, wherein the restriction is not upon the size of each shift, but upon the number of shifts. Bar-Noy, et al. [4] provided an optimal polynomial-time algorithm for this problem.

The interest in duty cycling developed in part from the introduction of the SET k-COVER problem by Slijepcevic and Potkonjak [16]. This problem, which they showed to be NP-hard, seeks to find at least k disjoint covers among a set of subsets of a base set. Perillo and Heinzelman [15] considered a variation in which each sensor has multiple modes. They translated the problem into a generalized maximum flow graph problem, and employed linear programming to find an optimal solution. Abrams et al. [1] provided approximation algorithms for a modification of the problem in which the objective was to maximize the total area covered by the sensors. Cardei et al. [6, 7, 8] considered adjustable range sensors, but also sought to maximize the number of non-disjoint set covers over a set of target coverage points.

The work of Pach and Tóth [13, 14] also has applications in this context. They showed that a k-fold cover of translates of a centrally-symmetric open convex polygon can be decomposed into $\Omega(\sqrt{k})$ covers. Aloupis, et al. [2] improved this to the optimal $\Omega(k)$ covers, and the centrally-symmetric restriction was later lifted by Gibson and Varadarajan [11]. In each of the above cases, the concept of finding many disjoint set covers, which can be seen as shifts, is used as a proxy for maximizing network lifetime.

Finally, the general STRIP COVER problem, in which each sensor has a *different* battery charge, was studied by Bar-Noy, et al. [4]. They also considered the SET ONCE STRIP COVER (OnceSC) problem, in which the radius and activation time of each sensor can be set only once. They showed that OnceSC is NP-hard, and that ROUNDROBIN (sensors take turns covering the entire interval) is a $\frac{3}{2}$-approximation algorithm for both OnceSC and STRIP COVER. Bar-Noy, et al. [4] also showed that the approximation ratio of any duty cycling algorithm is at least $\frac{3}{2}$ for both OnceSC and STRIP COVER. Bar-Noy and Baumer [3] also analyzed non-duty cycling algorithms for STRIP COVER with identical batteries. The CONNECTED RANGE ASSIGNMENT problem studied by Chambers, et al. [9], wherein the goal is to connect a series of points in the plane using circles, is also related. They presented approximation bounds for the case where solutions use a fixed number of circles, which is similar to limiting shift sizes.

Our results. In Section 2, we define the class of k-DutySC problems, and present the trivial solution to 1-DutySC. We present a polynomial-time algorithm, which we call MATCH, for 2-DutySC in Section 3. This algorithm is based on the a reduction to MAXIMUM WEIGHT MATCHING in bipartite graphs. In Section 4, we compare the performance of ROUNDROBIN to an algorithm that uses only a single shift. We prove that when the sensors are equi-spaced on the coverage interval, ROUNDROBIN performs most poorly in comparison to the one shift algorithm. Then we study the performance of ROUNDROBIN on these "perfect" deployments. This study is used to analyze MATCH in k-DutySC, but is of independent interest, since perfect deployments are the most natural. In Section 5 we show that MATCH is a $\frac{35}{24}$-approximation algorithm for k-DutySC. We also give two lower bounds on the performance of MATCH: $\frac{15}{11}$, for $k \geq 4$, and $\frac{6}{5}$, for $k = 3$, and provide experimental evidence suggesting that these lower bounds are tight. The question of whether k-DutySC is NP-hard, for $k \geq 3$, remains open. In Section 6, we consider a fault tolerance model, and show that if the failure rate of each sensor is sufficiently high, MATCH becomes optimal. We contend that even if the approximation ratio of k-DutySC for $k \geq 3$ is improved, MATCH will be of interest, due to its simplicity, performance, and fault tolerance.

2   Preliminaries

Duty cycles.   Let $U = [0, 1]$ be the interval that we wish to cover, and let $X = \{x_1, \ldots, x_n\} \in U^n$ be a set of n sensor locations. Henceforth we assume that $x_i \leq x_{i+1}$, for every $i \in \{1, \ldots, n-1\}$. We first assume that all sensors have unit capacity batteries (as opposed to uniform). We will justify this assumption later.

A pair (C, t), where $C \subseteq X$ is a subset of k sensor locations and $t \geq 0$, is called a *k-duty cycle* (or simply a *duty cycle*, or a *shift*). The sensors in C are activated at the same time and are deactivated together after t time units. A duty cycle (C, t) is feasible if the sensors in C can cover the interval $[0, 1]$ for the duration of t time units. More specifically, a sensor i such that $x_i \in C$ is assigned a radius $1/t$ and covers the range $[x_i - 1/t, x_i + 1/t]$, and the duty cycle is feasible if $[0, 1] \subseteq \bigcup_{i \in C}[x_i - 1/t, x_i + 1/t]$.

Let $\text{ALL}(C)$ denote the maximum t for which (C, t) is feasible. $\text{ALL}(C)$ is called the *lifetime* of C. Given a duty cycle $C = \{x_{i_1}, \ldots, x_{i_k}\}$ define

$$d_j \triangleq \begin{cases} 2x_{i_1} & j = 0, \\ 2(1 - x_{i_k}) & j = k, \\ x_{i_{j+1}} - x_{i_j} & \text{otherwise,} \end{cases}$$

and

$$\Delta \triangleq \max_j \{d_j\} .$$

Observation 1. $\text{ALL}(C) = \frac{2}{\Delta}$.

Proof. The maximum lifetime of C is at least $\frac{2}{\Delta}$, since the radial assignment $r_i = \frac{\Delta}{2}$ covers $[0, 1]$. However, if $\text{ALL}(C) > \frac{2}{\Delta}$, then $r_i < \frac{\Delta}{2}$ for every i. Hence, $[0, 1]$ is not covered.   □

In light of Observation 1, it suffices to refer to any subset $C \subseteq X$ as a shift, with a corresponding lifetime that is inferred from $\text{ALL}(C)$.

Problems. k-DUTY CYCLE STRIP COVER (abbreviated k-DUTYSC) is defined as follows. The input is a set $X = \{x_1, \ldots, x_n\} \in U^n$ of n sensor locations. A solution (or *schedule*) is a partition of X into m non-empty pairwise disjoint subsets $C_1, \ldots, C_m \subseteq X$ such that $|C_j| \leq k$, for every j. The goal is to find a solution that maximizes $\sum_j \text{ALL}(C_j)$. Thus, a solution to DUTYSC consists of a partition of X into shifts, where each shift employs ALL to achieve optimal lifetime.

Note that $\text{ALL}(C)$, for any shift C, and hence the maximum lifetime, are multiplied by a factor of B if all sensors have batteries with capacity B. Hence, throughout the paper we assume that all sensors have unit capacity batteries.

The optimal lifetime for k-DUTYSC is denoted by OPT_k. The best possible lifetime of a k-DUTYSC instance X, for any k, is $2n$.

Observation 2. $\text{OPT}_k(X) \leq 2n$, *for every* k.

Proof. Consider a schedule $C_1, \ldots, C_m$ and let Δ_i correspond to C_i. The minimum possible value of Δ_i is $1/|C_i|$, for every i. By Observation 1 we get that $\text{ALL}(C_i) \leq \frac{2}{\Delta_i} \leq 2|C_i|$. The observation follows from the fact that each of the n sensors is used in exactly one shift.   □

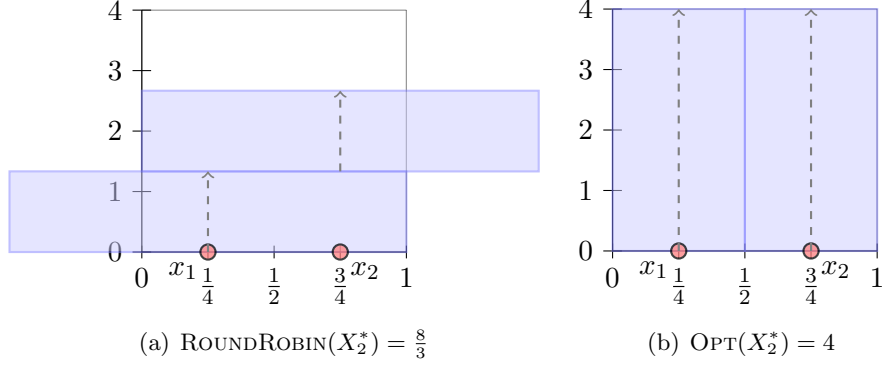(a) RoundRobin$(X_2^*) = \frac{8}{3}$ (b) Opt$(X_2^*) = 4$

Figure 1: Illustration of the $\frac{3}{2}$ upper bound on the approximation guarantee of RoundRobin. For the instance $X_2^* = \{\frac{1}{4}, \frac{3}{4}\}$, RoundRobin achieves a lifetime of only $\frac{8}{3}$ time units, while Opt achieves 4.

Perfect deployment. Define

$$X_n^* = \left\{ \frac{2i-1}{2n} : i \in \{1, \ldots, n\} \right\} = \left\{ \frac{1}{2n}, \frac{3}{2n}, \ldots, \frac{2n-1}{2n} \right\} .$$

We refer to X_n^* as the *perfect deployment* since the n-DutySC lifetime of X_n^* is $2n$, namely $\text{All}(X_n^*) = 2n$.

Round robin In 1-DutySC each sensor must work alone, therefore there is only one possible solution: $C_i = \{i\}$, for every i. Observe that this solution is valid for k-DutySC, for every k. We refer to the algorithm that generates this solution as RoundRobin. Observe that the RoundRobin lifetime is given by $\text{RR}(X) = \sum_i t_i$, where $t_i \triangleq \text{All}(C_i) = \min\{1/x_i, 1/(1-x_i)\}$ by Observation 1.

Bar-Noy, et al. [4] showed that RoundRobin is a $\frac{3}{2}$-approximation algorithm for Strip Cover. Since RoundRobin schedules are duty cycle schedules and any k-DutySC schedule is also a Strip Cover schedule, it follows that

Theorem 1. RoundRobin *is a $\frac{3}{2}$-approximation algorithm for k-DutySC, for every $k \geq 2$.*

The above ratio is tight due to the instance $X_2^* = \left\{\frac{1}{4}, \frac{3}{4}\right\}$ as shown in [3], and illustrated in Figure 1. We note that in our figures we depict schedules graphically as space-time diagrams. In each diagram, the horizontal axis represents the interval $[0, 1]$ that is to be covered, with the sensors in each instance illustrated by (red) circles at the appropriate locations. The vertical axis represents time, with the maximum possible lifetime of $2n$ visible at the top. The active coverage of each sensor is represented by a (blue) rectangle with a vertical arrow pointing upwards.

3   Strip Cover with Shifts of Size 2

We present a polynomial-time algorithm for solving 2-DutySC. The algorithm is based on a reduction to the Maximum Weight Matching problem in bipartite graphs that can be solved in $O(n^2 \log n + nm)$ in graphs with n vertices and m edges (see, e.g., [10]).
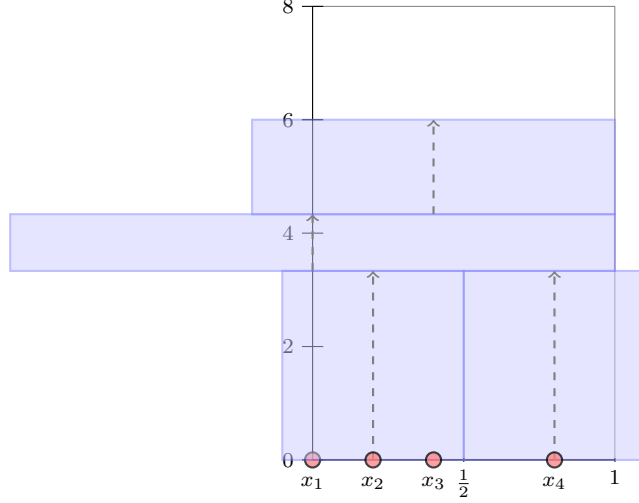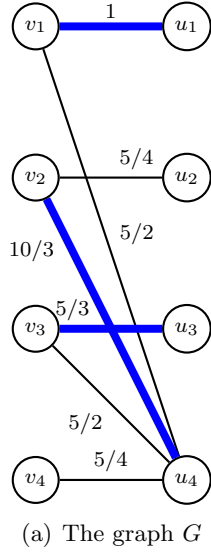
(a) The graph G (b) The optimal solution.

Figure 2: An example of 2-DUTYSC for $X = \{0, 0.2, 0.4, 0.8\}$. At left, the optimal matching is $\{(v_1, u_1), (v_2, u_4), (v_3, u_3)\}$ (in bold) and it corresponds to the optimal 2-DUTYSC solution $\{\{1\}, \{2, 4\}, \{3\}\}$ shown at right. The value of both solutions is 6.

Theorem 2. 2-DUTYSC *can be solved in polynomial time.*

Proof. Given a 2-DUTYSC instance X, with n sensors, we construct a bipartite graph $G = (L, R, E)$ as follows:

$$
\begin{aligned}
L &= \{v_i : i \in \{1, \ldots, n\}\} \\
R &= \{u_i : i \in \{1, \ldots, n\}\} \\
E &= \left\{(v_i, u_j) : x_i < \tfrac{1}{2} < x_j\right\} \cup \{(v_i, u_i) : i \in \{1, \ldots, n\}\} \ .
\end{aligned}
$$

The weight of an edge $e = (v_i, u_j)$ is defined as follows:

$$
w(e) = \begin{cases} \text{ALL}(\{x_i\}) & i = j, \\ \text{ALL}(\{x_i, x_j\}) & \text{otherwise.} \end{cases}
$$

Let $C_1, \ldots, C_m$ be a 2-DUTYSC solution for X. Notice that if there exists a shift $C_k = \{i, j\}$ such that $x_i, x_j \in [0, \tfrac{1}{2}]$, then a better solution may be obtained by splitting C_k into the two shifts $\{\{i\}, \{j\}\}$. The same argument applies to $x_i, x_j \in [\tfrac{1}{2}, 1]$. Hence, if $C_1, \ldots, C_m$ is an optimal 2-DUTYSC solution, then each shift contains either a singleton or a pair of sensors: one in $[0, \tfrac{1}{2})$ and the other in $(\tfrac{1}{2}, 1]$. It follows that an optimal 2-DUTYSC solution $C_1, \ldots, C_m$ induces a matching whose weight is the lifetime of the solution. Also, a matching $M \subseteq E$ induces a 2-DUTYSC solution whose lifetime is the weight of the matching. Hence, the weight of a maximum weight matching in G is the optimal 2-DUTYSC lifetime of X. An example is depicted in Figure 2 $\square$

The algorithm that is described in the theorem is henceforth referred to as Algorithm MATCH.

6

4 Round Robin vs. All

Assume we are given a set X of k sensors. In this section we compare $\text{RR}(X)$ to $\text{ALL}(X)$. This comparison will be used in the next section to analyze Algorithm MATCH for k-DUTYSC.

Define

$$\gamma(X) \triangleq \frac{\text{RR}(X)}{\text{ALL}(X)} \ .$$

In this section we look for a lower bound on $\min_{X:|X|=k} \gamma(X)$. Due to Theorem 1 it follows that $\gamma(X) \geq \frac{2}{3}$, for any set X of k sensors. In what follows, we prove the stronger result that the placement that minimizes the ratio is the perfect deployment, namely X_k^*. Notice that this is true for $k = 2$, since $\gamma(X_2^*) = \frac{2}{3}$.

4.1 Stretching the Instance

Our first step is to transform X into an instance X' for which $\gamma(X') \leq \gamma(X)$. This is done by pushing sensors away from $\frac{1}{2}$ so that all internal gaps are of size Δ. (See Section 2 for the definition of Δ.) If a sensor needs to be moved to the left of 0, it is placed at 0, and if it needs to move to the right of 1, it is placed at 1.

Definition 1. *For a given instance X, let j be the sensor whose location is closest to $\frac{1}{2}$. Then we define the* stretched *instance X' of X as follows:*

$$x_i' = \begin{cases} \max\{0, x_j - (j-i)\Delta\} & i < j, \\ x_j & i = j, \\ \min\{1, x_j + (i-j)\Delta\} & i > j. \end{cases}$$

We show that γ may only decrease.

Lemma 3. *Let X' be the stretched instance of X. Then, $\gamma(X') \leq \gamma(X)$.*

Proof. Sensors only get pushed away from $\frac{1}{2}$, and thus their ROUNDROBIN lifetime only decreases. Thus, $\text{RR}(X') \leq \text{RR}(X)$. By definition, Δ must equal either $2d_0$, $2d_k$ or the length of the largest internal gap in X. However neither d_0 nor d_k can be larger in X' than it was in X, since no sensors move closer to $\frac{1}{2}$. Moreover, by construction the length of the largest internal gap in X' is Δ. Hence $\Delta' \leq \Delta$, and $\text{ALL}(X') \geq \text{ALL}(X)$. □

Notice that it may be that $\Delta' < \Delta$. For example, consider $X = \{0.4, 0.5, 0.6\}$. In this case we have that $\Delta = 0.8$, $X' = \{0, 0.5, 1\}$, and $\Delta' = 0.5$.

4.2 Perfect Deployment is the Worst

By Lemma 3, it suffices to consider only stretched instances. The next step is to show that the worst stretched instance is in fact the perfect deployment. We use the following approach: we show that for every stretched non-perfect deployment X there is another deployment Y such that $\gamma(X) > \gamma(Y)$.

7

A stretched instance X with k sensors can be described as follows:

$$X = \left\{ 0^{k_0}, a, a + \Delta, \ldots, a + (k_{in} - 1)\Delta = 1 - b, 1^{k_1} \right\}$$

where $k_{in} \geq 1$, $k_0 \geq 0$, $k_1 \geq 0$, and $k = k_{in} + k_0 + k_1$. We denote $k_{out} \triangleq k_0 + k_1$. Notice that it may be the case that $a = 0$ and/or $b = 0$. In what follows we use both $a = 0$ and $a = \Delta$ (and a different k_0) to describe the same instance. The same applies to $b = 0$ and $b = \Delta$.

If $k = 1$, then $\mathrm{RR}(X) = \mathrm{ALL}(X)$. Recall that $\gamma(X) \geq \frac{2}{3} = \gamma(X_2^*)$ due to Theorem 1, and therefore we may assume that $k = k_{in} + k_{out} \geq 3$. Nevertheless, we prove that the perfect deployment is the worst for $k \geq 2$. (We consider the case of $k = 2$ for completeness.)

For reasons of symmetry we assume, without loss of generality, that $0 \leq a \leq b$. Hence, $\lceil k_{in}/2 \rceil$ sensors are located in $(0, \frac{1}{2}]$ and $\lfloor k_{in}/2 \rfloor$ sensors are located in $(\frac{1}{2}, 1)$. Also, observe that a sensor located at 0 contributes to RR exactly the same as a sensor located at 1. On the other hand, if $k_{out} = 1$, $\mathrm{ALL}(X')$ is maximized with a sensor at 1. If $k_{out} \geq 2$, $\mathrm{ALL}(X')$ is maximized if there is at least one sensor at 0 and at least one at 1. Hence, we may assume without loss of generality that there are $k_0 = \lfloor k_{out}/2 \rfloor$ sensors at 0, and $k_1 = \lceil k_{out}/2 \rceil$ sensors at 1.

It follows that it is sufficient to consider stretched deployments X that can be described as follows:

$$X = \left\{ 0^{\lfloor k_{out}/2 \rfloor}, a, a + \Delta, \ldots, a + (k_{in} - 1)\Delta = 1 - b, 1^{\lceil k_{out}/2 \rceil} \right\}$$

where $0 \leq a \leq b$. Since the k_{out} sensors located at 0 and 1 contribute a lifetime of exactly 1, the ROUNDROBIN lifetime of X is:

$$\mathrm{RR}(X) = k_{out} + \sum_{i=0}^{\lceil k_{in}/2 \rceil - 1} \frac{1}{1 - (a + i\Delta)} + \sum_{i=0}^{\lfloor k_{in}/2 \rfloor - 1} \frac{1}{1 - (b + i\Delta)} . \tag{1}$$

Due to (1) and Observation 1 we have that

$$\gamma(X) = \frac{\Delta}{2} \left[k_{out} + \sum_{i=0}^{\lceil k_{in}/2 \rceil - 1} \frac{1}{1 - (a + i\Delta)} + \sum_{i=0}^{\lfloor k_{in}/2 \rfloor - 1} \frac{1}{1 - (b + i\Delta)} \right]$$

$$= \frac{k_{out}}{2} \cdot \Delta + \sum_{i=0}^{\lceil k_{in}/2 \rceil - 1} \frac{\Delta}{2 - 2(a + i\Delta)} + \sum_{i=0}^{\lfloor k_{in}/2 \rfloor - 1} \frac{\Delta}{2 - 2(b + i\Delta)} . \tag{2}$$

Let $\Omega_{k_{out}}$ denote the set of stretched instances with k_{out} sensors on 0 and 1. We distinguish between three cases as illustrated in Figure 3:

1. $X \in \Omega_0$. In this case, $a \in [0, \Delta/2]$ and $b \in [a, \Delta/2]$.

2. $X \in \Omega_1$. In this case, $a \in [0, \Delta/2]$ and $b \in [a, \Delta]$.

3. $X \in \Omega_{k_{out}}$, for $k_{out} \geq 2$. In this case, $a \in [0, \Delta]$ and $b \in [a, \Delta]$.

We first explore γ in $\Omega_{k_{out}}$, for $k_{out} \geq 2$.

Lemma 4. γ has no local minima in $\Omega_{k_{out}}$, for any $k_{out} \geq 2$. Furthermore, the minimum is obtained when $a = b = \Delta$.

8

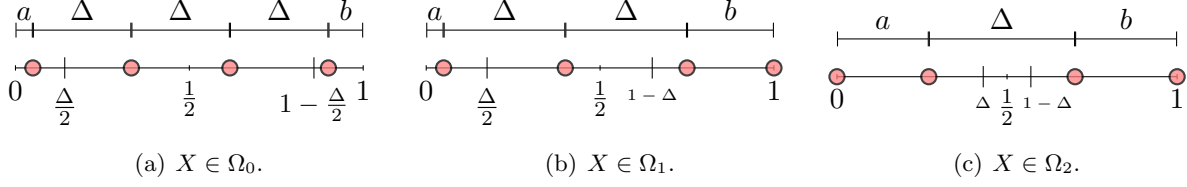(a) $X \in \Omega_0$.   (b) $X \in \Omega_1$.   (c) $X \in \Omega_2$.

Figure 3: Illustration of $\Omega_{k_{out}}$.

Proof. First, assume that $k_{in} \geq 2$. In this case, $\Delta = \frac{1-a-b}{k_{in}-1}$. We compute $\frac{\partial \gamma}{\partial a}$ using (2). Since $\frac{\partial \Delta}{\partial a} = -\frac{1}{k_{in}-1}$, it follows that

$$\frac{\partial}{\partial a}\left(\frac{\Delta}{2-2(a+i\Delta)}\right) = \frac{(2-2(a+i\Delta))(\frac{\partial \Delta}{\partial a}) - \Delta(-2-2i\frac{\partial \Delta}{\partial a})}{(2-2(a+i\Delta))^2} = \frac{2\left(\frac{\partial \Delta}{\partial a}(1-a)+\Delta\right)}{4(1-(a+i\Delta))^2}$$

$$= \frac{\left(-\frac{1}{k_{in}-1}(1-a)+\Delta\right)}{2(1-(a+i\Delta))^2}$$

$$= \frac{-b}{2(k_{in}-1)\cdot(1-(a+i\Delta))^2} \, ,$$

and

$$\frac{\partial}{\partial a}\left(\frac{\Delta}{2-2(b+i\Delta)}\right) = \frac{(2-2(b+i\Delta))(\frac{\partial \Delta}{\partial a}) + \Delta 2i\frac{\partial \Delta}{\partial a}}{(2-2(b+i\Delta))^2} = \frac{2(1-b)\frac{\partial \Delta}{\partial a}}{4(1-(b+i\Delta))^2}$$

$$= \frac{(1-b)\frac{-1}{k_{in}-1}}{2(1-(b+i\Delta))^2}$$

$$= \frac{b-1}{2(k_{in}-1)\cdot(1-(b+i\Delta))^2} \, .$$

Thus,

$$\frac{\partial \gamma(X)}{\partial a} = \frac{-k_{out}}{2(k_{in}-1)} + \frac{1}{2(k_{in}-1)}\left[\sum_{i=0}^{\lceil k_{in}/2\rceil - 1}\frac{-b}{(1-(a+i\Delta))^2} + \sum_{i=0}^{\lfloor k_{in}/2\rfloor - 1}\frac{b-1}{(1-(b+i\Delta))^2}\right] < 0 \, .$$

Hence $\gamma(X)$ decreases as a increases. An analogous calculation shows that the same is true for b, namely that $\frac{\partial \gamma(X)}{\partial b} < 0$. Since neither $\frac{\partial \gamma(X)}{\partial a}$ nor $\frac{\partial \gamma(X)}{\partial b}$ can be zero at any point in the interior of the domain $\Omega_{k_{out}}$, $\gamma(X)$ has no local minima. Finally, any minima must occur when both a and b are as large as possible within the domain $\Omega_{k_{out}}$, namely when $a = b = \Delta$.

It remains to consider the case where $k_{in} = 1$. Since $k_{out} \geq 2$, we have that $\Delta = b$. Hence,

$$\gamma(X) = \frac{b}{2}\left(k_{out} + \frac{1}{b}\right) = \frac{1}{2}(b \cdot k_{out} + 1) \, ,$$

which means that $\frac{\partial \gamma(X')}{\partial b} > 0$. Hence, $\gamma(X')$ decreases as b decreases. It follows that the minimum occurs when $a = b = \frac{1}{2}$. $\square$

Next we consider Ω_1.

9

Lemma 5. γ *has no local minima in* Ω_1. *Furthermore, the minimum is obtained when* $a = \Delta/2$ *and* $b = \Delta$.

Proof. First, assume that $k_{in} \geq 2$. In this case we have that $\Delta = \frac{1-a-b}{k_{in}-1}$, since otherwise this instance is not stretched. It follows that $\frac{\partial \gamma(X)}{\partial a} < 0$ and $\frac{\partial \gamma(X)}{\partial b} < 0$ as shown in Lemma 4 for the case where $k_{in} \geq 2$. Hence the minimum occurs when both a and b are as large as possible within the domain Ω_1, namely when $a = \Delta/2$ and $b = \Delta$.

If $k_{in} = 1$, we have that $\Delta = \max\{2a, b\}$. If $b > 2a$, $\gamma(X)$ decreases as b decreases as shown in Lemma 4 for the case where $k_{in} = 1$. However, if $b < 2a$, we have that

$$\gamma(X) = \frac{2a}{2}\left(k_{out} + \frac{1}{1-a}\right) = a + \frac{a}{1-a} \; ,$$

which means that

$$\frac{\partial \gamma(X)}{\partial a} = 1 + \frac{1-a+a}{(1-a)^2} = 1 + \frac{1}{(1-a)^2} > 0 \; .$$

Hence, $\gamma(X)$ decreases as a decreases. It follows that the minima occurs then $2a = b = \frac{2}{3}$. $\quad\square$

Finally, we deal with Ω_0.

Lemma 6. $X_k^* = argmin_{X \in \Omega_0} \gamma(X)$.

Proof. First, note that $k_{in} \geq 2$ if $X \in \Omega_0$. Observe that if $\Delta = 2b > \frac{1-a-b}{k_{in}-1}$, then X is not stretched. Hence, if $X \in \Omega_0$, then $\Delta = \frac{1-a-b}{k_{in}-1}$. It follows that $\frac{\partial \gamma(X)}{\partial a} < 0$ and $\frac{\partial \gamma(X)}{\partial b} < 0$ as shown in Lemma 4 for the case where $k_{in} \geq 2$. Hence the minimum occurs when both a and b are as large as possible within the domain Ω_0, namely when $a = \Delta/2$ and $b = \Delta/2$. Namely when $X = X_k^*$. $\quad\square$

Let $\gamma_k^* = \gamma(X_k^*)$. We show that, for any fixed k, $\gamma(X)$ reaches its minimum at $X = X_k^*$. We do so by showing that, for every $k_{out} > 0$, the deployment X in $\Omega_{k_{out}}$ that minimizes γ corresponds to a deployment from $\Omega_{k_{out}-1}$.

Theorem 3. $\min_{X:|X|=k} \gamma(X) = \gamma_k^*$.

Proof. We prove that $\min_{X:|X|=k} \gamma(X) = \gamma_k^*$ by induction on k_{out}. For the base case, if $X \in \Omega_0$, then by Lemma 6, $\gamma(X)$ achieves its minimum in X_k^*. For the inductive step, let $X \in \Omega_{k_{out}}$, for $k_{out} \geq 1$, and assume that $\min_{X \in \Omega_{k_{out}-1}} \gamma(X) \geq \gamma_k^*$. If $k_{out} = 1$ (and $k_{in} \geq 1$), then by Lemma 5 the minimum in Ω_1 is reached when $a = \Delta/2$ and $b = \Delta$ (if $k_{in} = 1$, then the minimum is when $2a = b = \frac{2}{3}$), namely for $X = \{\frac{\Delta}{2}, \frac{3\Delta}{2}, \ldots, 1 - \Delta, 1\}$. By symmetry, this instance has the same ratio as the instance $X' = \{1 - x : x \in X\}$, which is in Ω_0 with parameters $a = 0$ and $b = \frac{\Delta}{2}$. If $k_{out} > 1$, then Lemma 4 the minimum in $\Omega_{k_{out}}$ is obtained when $a = \Delta$ and $b = \Delta$. In this case $X' \in \Omega_{k_{out}-1}$ with parameters $a = 0$ and $b = \Delta$. Hence by the induction hypothesis we have that $\gamma(X) \geq \gamma_k^*$. $\quad\square$

4.3 Properties of γ_k^*

In this section we explore γ_k^* as a function of k. Observe that for even k we have that

$$\gamma_k^* = \frac{1}{2k} \cdot 2 \sum_{i=1}^{k/2} \frac{2k}{2k+1-2i} = 2 \sum_{i=1}^{k/2} \frac{1}{2k+1-2i} = 2 \sum_{i=k/2+1}^{k} \frac{1}{2i-1} \ ,$$

and for odd k we have that

$$\gamma_k^* = \frac{1}{2k} \left[2 + 2 \sum_{i=1}^{(k-1)/2} \frac{2k}{2k+1-2i} \right] = \frac{1}{k} + 2 \sum_{i=(k+1)/2}^{k-1} \frac{1}{2i+1} \ .$$

Lemma 7. γ_k^* satisfies the following: (i) $\gamma_k^* \leq \gamma_{k+2}^*$, for every even k. (ii) $\gamma_k^* \geq \gamma_{k+2}^*$, for every odd k. (iii) $\gamma_k^* \geq \gamma_{k+1}^*$, for every odd k.

Proof. Due to the convexity of the function $f(z) = \frac{1}{z}$, we have that for even k,

$$\gamma_{k+2}^* - \gamma_k^* = 2 \sum_{i=k/2+2}^{k+2} \frac{1}{2i-1} - 2 \sum_{i=k/2+1}^{k} \frac{1}{2i-1} = \frac{2}{2k+3} + \frac{2}{2k+1} - \frac{2}{k+1} > 0 \ .$$

By the same rationale, for odd k,

$$\begin{aligned}
\gamma_k^* - \gamma_{k+2}^* &= \left(\frac{1}{k} + 2 \sum_{i=(k+1)/2}^{k-1} \frac{1}{2i+1} \right) - \left(\frac{1}{k+2} + 2 \sum_{i=(k+1)/2+1}^{k+1} \frac{1}{2i+1} \right) \\
&= \frac{1}{k} + \frac{2}{k+2} - \frac{1}{k+2} - \frac{2}{2k+1} - \frac{2}{2k+3} \\
&= \frac{1}{k} + \frac{1}{k+2} - \frac{2}{2k+1} - \frac{2}{2k+3} \\
&> 0 \ .
\end{aligned}$$

Finally, for odd k,

$$\gamma_k^* - \gamma_{k+1}^* = \frac{1}{k} + 2 \sum_{i=(k+1)/2}^{k-1} \frac{1}{2i+1} - 2 \sum_{i=(k+1)/2+1}^{k+1} \frac{1}{2i-1} = \frac{1}{k} - \frac{2}{2k+1} > 0 \ ,$$

as required. $\qquad\square$

Lemma 7 implies that

Corollary 4. $\gamma(X) \geq \frac{2}{3}$, for every X.

Furthermore, γ_k^* goes to $\ln 2$ as k goes to infinity.

Lemma 8. $\lim_{k\to\infty} \gamma_k^* = \ln 2$.

k	γ_k^*	Approx
1	1	1
2	$\frac{2}{3}$	1.5000
3	$\frac{11}{15}$	1.3636
4	$\frac{24}{35}$	1.4583
5	$\frac{223}{315}$	1.4126
6	$\frac{478}{693}$	1.4498
7	$\frac{6313}{9009}$	1.4271
8	$\frac{4448}{6435}$	1.4467
$\vdots$	$\vdots$	$\vdots$
∞	$\ln 2$	1.4427

(a) Exact and approximate values of γ_k^*.

(b) γ_k^* is an alternating sequence that converges to $\ln 2$

Figure 4: Tabular and graphical representation of small values of γ_k^*.

Proof. Observe that for both even and odd k's we have that

$$\gamma_k^* \geq \sum_{i=k+1}^{2k} \frac{1}{i} = H_{2k} - H_k \qquad \text{and} \qquad \gamma_k^* \leq \sum_{i=k}^{2k-1} \frac{1}{i} = H_{2k-1} - H_{k-1},$$

where H_k the kth Harmonic number. It follows that $\lim_{k\to\infty} \gamma_k^* = \lim_{k\to\infty}(H_{2k} - H_k) = \ln 2$. $\quad\square$

The table in Figure 4(a) contains several values of γ_k^*, whose convergence is also depicted graphically in Figure 4(b).

5 Strip Cover with Shifts of Size k

In this section we analyze the performance of MATCH in k-DUTYSC for $k \geq 3$. Recall that Algorithm MATCH finds the best solution among those using shifts of size at most 2. Since MATCH is more powerful than ROUNDROBIN, its approximation ratio is at most $\frac{3}{2} = 1.5$ (by Theorem 1). We show that the approximation ratio of MATCH is at most $\frac{35}{24} \approx 1.458$. We also provide lower bounds: $\frac{15}{11} \approx 1.364$ for $k \geq 4$, and $\frac{6}{5} = 1.2$, for $k = 3$. At the end of the section we discuss ways to improve the analysis of MATCH.

5.1 Upper Bound

We use our analysis of ROUNDROBIN vs. ALL to obtain an upper bound on the performance of MATCH for k-DUTYSC.

Theorem 5. *Algorithm* MATCH *is a $\frac{35}{24}$-approximation algorithm for k-DUTYSC, for every $k \geq 3$.*

Proof. Let X be a k-DUTYSC instance and let $C_1, \ldots, C_m$ be an optimal solution for X. Consider the 2-DUTYSC solution that is obtained by splitting each shift C_j with $|C_j| > 2$ into singleton shifts. For any such C_j, by Theorem 3 we know that $\mathrm{RR}(C_j) \geq \gamma^*_{|C_j|}\mathrm{ALL}(C_j)$, and by Lemma 7 we know that $\gamma^*_{|C_j|} \geq \min_{k \geq 3} \gamma^*_k = \gamma^*_4 = \frac{24}{35}$. Hence there exists a 2-DUTYSC solution whose lifetime is at least $\frac{24}{35} \cdot \mathrm{OPT}_k$. The lemma follows since MATCH finds the best 2-DUTYSC solution. $\square$

Due to Lemma 7 we can generalize this approach to find upper bounds on the performance of OPT_k in n-DUTYSC, for $k \leq n$.

Lemma 9. $\mathrm{OPT}_k(X) \geq \gamma^*_\ell \mathrm{OPT}_n(X)$, *where $\ell = 2\left\lceil\frac{k+1}{2}\right\rceil$ is the smallest even integer larger than k.*

Hence, if we use an algorithm that solves k-DUTYSC to solve n-DUTYSC and the approach of Theorem 5 the resulting approximation ratio would be at most $1/\gamma^*_\ell$. Lemma 8 implies that the smallest upper bound achievable via this technique is $1/\ln 2 \approx 1.4427$. Note that it is possible that a better ratio may be achieved using a different approach.

5.2 Lower Bounds

We show that the approximation ratio of MATCH is at least $\frac{15}{11}$, for $k \geq 4$, and at least $\frac{6}{5}$, for $k = 3$.

Lemma 10. $\mathrm{MATCH}(X^*_4) = \frac{11}{15}\mathrm{OPT}_k(X^*_4)$, *for every $k \geq 4$.*

Proof. Consider the instance $X^*_4 = \{\frac{1}{8}, \frac{3}{8}, \frac{5}{8}, \frac{7}{8}\}$. Observe that the ROUNDROBIN lifetime of the sensors on the outside is $\mathrm{RR}(\frac{1}{8}) = \mathrm{RR}(\frac{7}{8}) = \frac{8}{7}$, while the lifetime is $\mathrm{RR}(\frac{3}{8}) = \mathrm{RR}(\frac{5}{8}) = \frac{8}{5}$ for the sensors in the middle. Perhaps surprisingly, any sensible pairing of the sensors achieves a lifetime of $8/3$. Thus, one MATCH solution is to pair the outside sensors for a lifetime of $8/3$, and then run ROUNDROBIN on the middle sensors successively, for an additional lifetime of $2 \cdot \frac{8}{5}$, as depicted in Figure 5. Thus, the total lifetime is $\mathrm{MATCH}(X^*_4) = \frac{8}{3} + 2 \cdot \frac{8}{5} = \frac{88}{15}$. The lemma follows, since $\mathrm{OPT}_k(X^*_4) = \mathrm{ALL}(X^*_4) = 8$, for every $k \geq 4$. $\square$

We have a weaker lower bound for $k = 3$.

Lemma 11. $\mathrm{MATCH}(X^*_3) = \frac{5}{6}\mathrm{OPT}_3(X^*_3)$.

Proof. MATCH has 2 duty cycles: $\{\frac{1}{6}, \frac{5}{6}\}, \{\frac{1}{2}\}$. Hence, $\mathrm{MATCH}(X^*_3) = 5$, while $\mathrm{ALL}(X^*_3) = 6$. $\square$

We conjecture that both Lemma 10 and Lemma 11 are tight due to the fact that computer-based searches for counterexamples have failed.

Conjecture 1. $\mathrm{MATCH}(X) \geq \frac{11}{15}\mathrm{OPT}_k(X)$, *for every $k \geq 4$, and* $\mathrm{MATCH}(X) \geq \frac{5}{6}\mathrm{OPT}_3(X)$.

For some positive integer d, let $D_d = \{\frac{i}{d} : i \in \{0, 1, \ldots, d\}\}$ be a discretization of $[0, 1]$. Clearly, as $d \to \infty$, D becomes a close approximation of $[0, 1]$. Using brute force, we checked all 680 possible 3-sensor instances and all 2380 possible 4-sensor instances in D_{16}, and found no instance X for which $\mathrm{MATCH}(X) < \frac{5}{6}\mathrm{OPT}_3(X)$ in the first case, nor any for which $\mathrm{MATCH}(X) < \frac{11}{15}\mathrm{OPT}_4(X)$ in the second case.

Again, we can extend the ideas embedded in Lemma 10 to obtain bounds on the performance of OPT_k in n-DUTYSC.
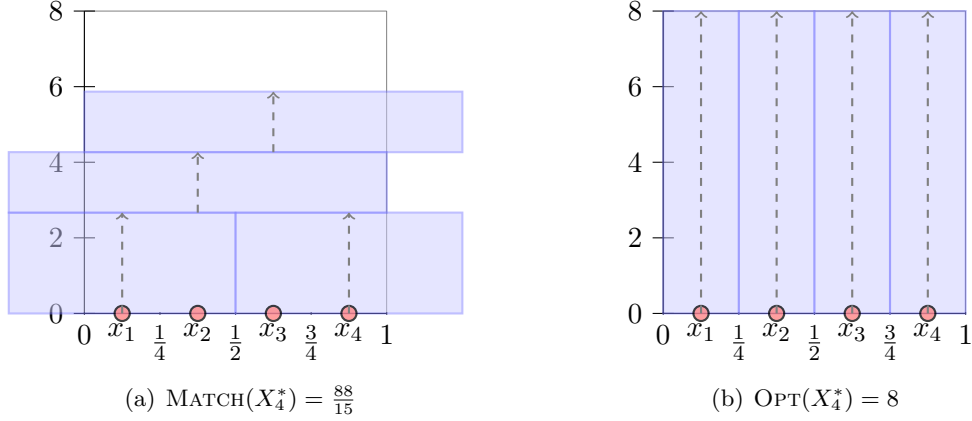
(a) $\textsc{Match}(X_4^*) = \frac{88}{15}$ (b) $\textsc{Opt}(X_4^*) = 8$

Figure 5: Illustration of the $\frac{15}{11}$ upper bound for the approximation ratio of $\textsc{Match}$ for $k \geq 4$. For the instance $X_4^* = \{\frac{1}{8}, \frac{3}{8}, \frac{5}{8}, \frac{7}{8}\}$, $\textsc{Match}$ achieves a lifetime of $\frac{88}{15}$ while $\textsc{Opt}$ achieves 8.

Lemma 12. $\min_{X:|X|=n} \frac{\textsc{Opt}_k(X)}{\textsc{Opt}_n(X)} \leq \frac{\textsc{Opt}_k(X_\ell^*)}{2\ell}$, *for every* $\ell \leq n$.

Note that the bound of $\frac{2}{3}$ for RR was obtained for $k = 1$ and $\ell = 2$, while the bounds of $\frac{5}{6}$ and $\frac{11}{15}$ for $\textsc{Match}$ were obtained for $k = 2$ and $\ell = 3, 4$.

5.3 Asymptotics

One way to improve the analysis of Algorithm $\textsc{Match}$ would be to first prove that perfect deployments are worst with respect to $\textsc{Match}$ (as they are with respect to $\textsc{RoundRobin}$), and then to analyze $\gamma_k^2 = \textsc{Match}(X_k^*)/\textsc{All}(X_k^*)$.

Our experiments show that γ_k^2 seems to converge to approximately 0.816, which is significantly higher than $\lim_{k \to \infty} \gamma_k^* = \ln 2 \approx 0.693$. See Figure 7.

We would like to evaluate $\lim_{k \to \infty} \gamma_k^2$. However, since $\textsc{Match}$ explicitly evaluates each pair of sensors, it is not trivial to extrapolate its behavior for large k. Nevertheless, we obtain lower bounds on the limit by analyzing simple heuristics. For a given instance X, let $L = X \cap [0, \frac{1}{3}]$, $M = X \cap (\frac{1}{3}, \frac{2}{3})$, and $R = X \cap [\frac{2}{3}, 1]$. Note that $|L| = |R|$ for the perfect deployment instance X_k^* of any size. Consider the following heuristics:

- $\textsc{Sweep}$: Sensors in L and R are paired to create shifts of size two, starting with the leftmost sensor in L and the *leftmost* sensor in R. Any remaining sensors are put in size one shifts.

- $\textsc{Nest}$: Sensors in L and R are paired to create shifts of size two, starting with the leftmost sensor in L and the *rightmost* sensor in R. Any remaining sensors are put in size one shifts.

Note that these heuristics return valid (albeit suboptimal) 2-$\textsc{DutySC}$ solutions. Examples of these solutions are shown in Figure 6.
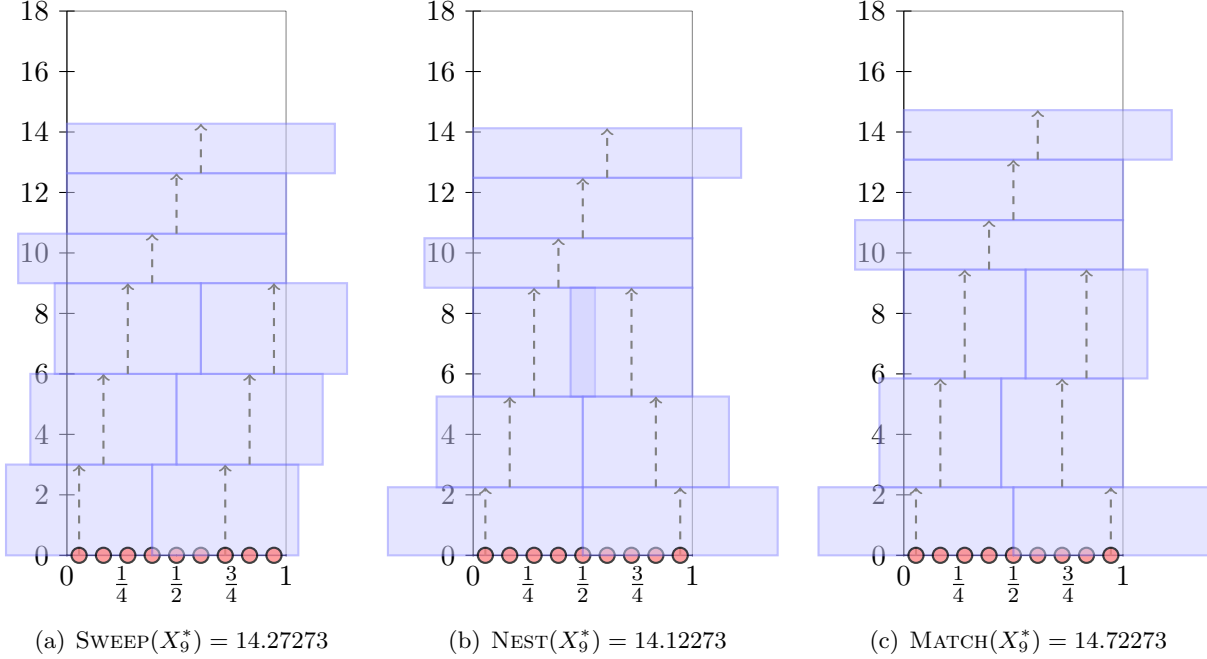
14

(a) $\textsc{Sweep}(X_9^*) = 14.27273$ (b) $\textsc{Nest}(X_9^*) = 14.12273$ (c) $\textsc{Match}(X_9^*) = 14.72273$

Figure 6: Illustration of the performance of the two heuristics, $\textsc{Sweep}$ and $\textsc{Nest}$, and Algorithm $\textsc{Match}$. For the instance X_9^*, each procedure returns a different solution. The lifetimes achieved by $\textsc{Sweep}$ and $\textsc{Nest}$ represent lower bounds on the performance of $\textsc{Match}$.

Using similar arguments as in Section 4.3 we can show that:

$$\frac{\textsc{Sweep}(X_n^*)}{\textsc{All}(X_n^*)} \approx 1/2 + H_{4n/3} - H_n$$

$$\frac{\textsc{Nest}(X_n^*)}{\textsc{All}(X_n^*)} \approx (H_n - H_{n/2})/2 + (H_{2k/3} - H_{k/2})/2 + (H_{4n/3} - H_n)$$

which means that

$$\lim_{n\to\infty} \frac{\textsc{Sweep}(X_n^*)}{\textsc{All}(X_n^*)} = \frac{1}{2} + \ln(4/3) \approx 0.788$$

$$\lim_{n\to\infty} \frac{\textsc{Nest}(X_n^*)}{\textsc{All}(X_n^*)} = \frac{1}{2}\ln 2 + \frac{1}{2}\ln(4/3) + \ln(4/3) \approx 0.778$$

A comparison of the performance of $\textsc{RoundRobin}$, $\textsc{Match}$, $\textsc{Sweep}$ and $\textsc{Nest}$ on perfect deployments to the performance of $\textsc{Opt}$ is given in Figure 7.

In Figure 8, we show the density of the per sensor network lifetime $(\textsc{Alg}(X)/n)$ for $\textsc{RoundRobin}$, $\textsc{All}$, $\textsc{Match}$, and $\textsc{Opt}$ (computed via brute force), for all 2380 possible instances with 4 sensors over a grid of size 16. X_4^* was the only instance found for which $\textsc{Match}(X) \leq \frac{11}{15}\textsc{Opt}_4(X)$. Moreover, for about 82% of the instances, $\textsc{Match}(X) = \textsc{Opt}(X)$, and the mean approximation ratio was 0.9923. Meanwhile, the average per sensor lifetime for $\textsc{Match}$ was 1.483, which easily surpasses the corresponding figure for $\textsc{RoundRobin}$ of 1.365.
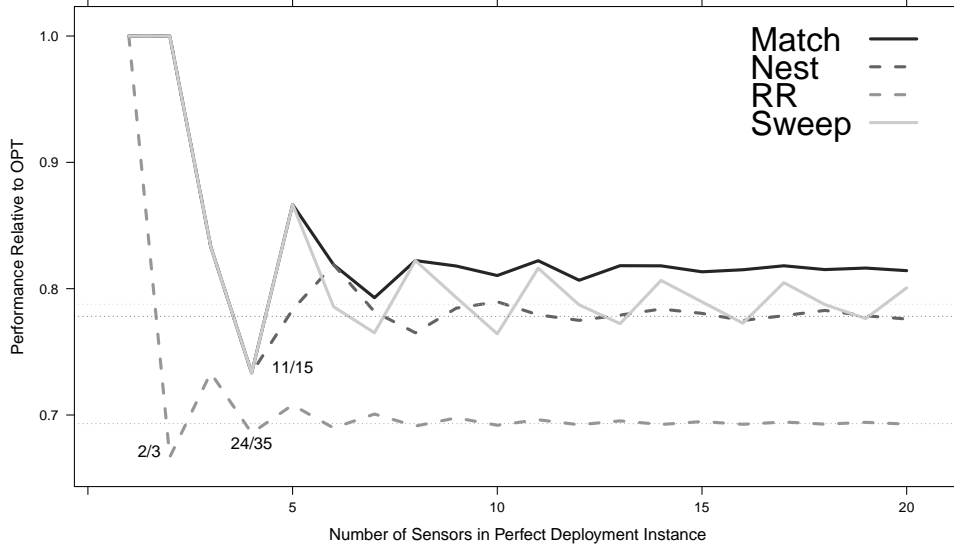
Figure 7: Performance of ROUNDROBIN, MATCH, SWEEP, and NEST compared to OPT on perfect deployments.

6 Fault Tolerance

In this section we extend our analysis to incorporate a fault tolerance model, in which each sensor may fail to activate with probability $p \in [0, 1]$. We assume that failures occur randomly and independently. If any sensor in a shift fails to activate, then the entire coverage lifetime of that shift is lost. Under these assumptions, we can make the following observation about the expected lifetime of an algorithm.

Observation 13. *For any shift $C \subseteq X$, the expected lifetime of the shift is $(1-p)^{|C|} \cdot \text{ALL}(C)$.*

The expected lifetime of a solution $C_1, \ldots, C_m \subseteq X$ is $\sum_i (1-p)^{|C_i|} \cdot \text{ALL}(C_i)$. In the fault tolerant version of k-DUTYSC our goal is to find a solution $C_1, \ldots, C_m \subseteq X$ such that $|C_i| \leq k$ with maximum expected lifetime. Let $\text{OPT}_k^p(X)$ denote the expected lifetime of an optimal k-DUTYSC solution of X.

Theorem 6. *Fault tolerant* 2-DUTYSC *can be solved in polynomial time.*

Proof. The proof is similar to the proof of Theorem 2. The only difference is that the weight of an edge $e = (v_i, v_j')$ is defined as follows:

$$w(e) = \begin{cases} (1-p) \cdot \text{ALL}\{x_i\} & i = j, \\ (1-p)^2 \cdot \text{ALL}(\{x_i, x_j\}) & i < j. \end{cases}$$

$\square$

We show that if the probability of failure is high enough, MATCH, or even ROUNDROBIN, compute optimal solutions.
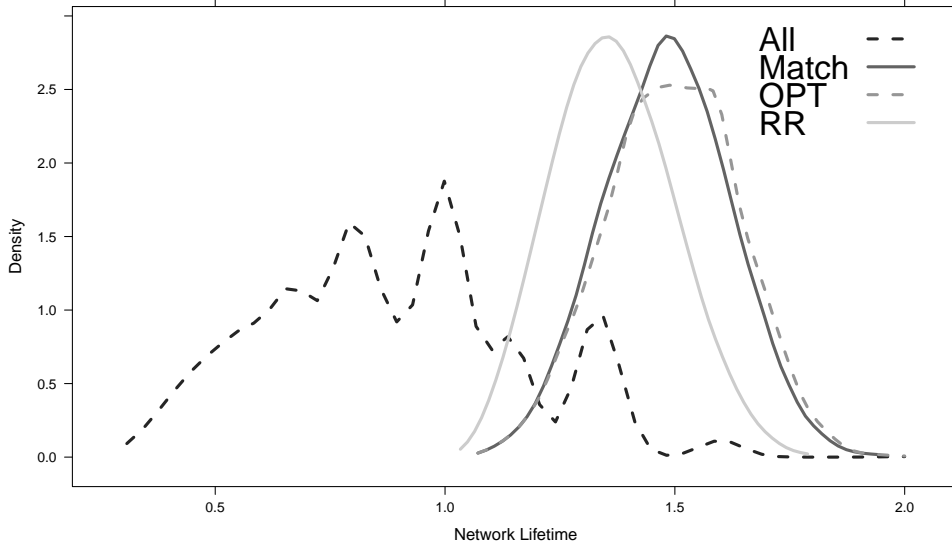
16

Figure 8: Comparison of the Distribution of Per Sensor Network Lifetime for various algorithms in 4-DUTYSC. The lifetime of each algorithm was computed for all 2380 possible instances of 4 sensors on a grid of size 16.

Lemma 14. *If $p \geq \frac{1}{3}$, then $\mathbb{E}[\mathrm{RR}(X)] = \mathrm{OPT}_k^p(X)$, for every X.*

Proof. Let $C_1, \ldots, C_m$ be an optimal schedule. Consider any shift C_j for which $|C_j| \geq 2$. By Observation 13, the expected lifetime of that shift is at most $(1-p)^2 \cdot \mathrm{ALL}(C_j)$. Since $\gamma(C_j) \geq \frac{2}{3}$ for every C_j, due to Corollary 4, we have that $\mathrm{RR}(C_j) \geq \frac{2}{3}\mathrm{ALL}(C_j)$. It follows that

$$(1-p)^2 \cdot \mathrm{ALL}(C_j) \leq \frac{2}{3}(1-p) \cdot \mathrm{ALL}(C_j) \leq (1-p) \cdot \mathrm{RR}(C_j) .$$

$\square$

Lemma 15. *If $p \geq 1 - \sqrt{\gamma_3^*} \approx 0.144$, then $\mathbb{E}[\mathrm{MATCH}(X)] = \mathrm{OPT}_k^p(X)$, for every X.*

Proof. Let $C_1, \ldots, C_m$ be an optimal schedule. Consider any shift C_j for which $|C_j| \geq 3$. By Observation 13, the expected lifetime of that shift is $(1-p)^{|C_j|} \cdot \mathrm{ALL}(C_j)$. Due to Theorem 3 we have that

$$(1-p)^{|C_j|}\mathrm{ALL}(C_j) \leq \frac{(1-p)^{|C_j|-1}}{\gamma_k^*} \cdot (1-p)\mathrm{RR}(C_j) .$$

If $|C_j| = 3$, we have that $(1-p)^{|C_j|-1}/\gamma_k^* \leq 1$, since $p \geq 1 - \sqrt{\gamma_3^*} \approx 0.144$. Also, for $|C_j| > 3$ we have that $(1-p)^{|C_j|-1}/\gamma_k^* \leq (1-p)^3/\gamma_4^* < 1$, since $p > 1 - \sqrt[3]{\gamma_4^*} \approx 0.118$. It follows that if $p \geq 1 - \sqrt{\gamma_3^*}$, then there exists an optimal schedule that does not use shifts of size larger than 2. Hence, Algorithm MATCH computes an optimal solution. $\square$

7 Discussion and Open Problems

While 1-DutySC can be solved trivially by RoundRobin, and we have shown that 2-DutySC can be solved in polynomial time using Algorithm Match, it remains open whether k-DutySC is NP-hard, for $k \geq 3$. It would also be interesting to close the gap between the upper and lower bounds on the approximation ratio of Match, for $k \geq 3$. We offered one possible direction to improving the upper bound in Section 5.3.

In this paper, we have assumed that: (i) the initial battery charge of each sensor is identical; and (ii) the battery charge in each sensor drains in inverse linear proportion to its assigned radius. Two natural extensions of our work would be to allow the initial battery charges to differ, and to allow the latter proportion to vary according to some exponent $\alpha \neq 1$.

Finally, while we have restricted our attention to a one-dimensional coverage region, one could consider a variety of similar problems in higher dimensions. For example, one might keep the sensor locations restricted to the line, but consider a two-dimensional coverage region. Conversely, the sensors could be located in the plane, while the coverage region remains one-dimensional. Of course, an even more general problem would allow both the sensor locations and the coverage region to be two-dimensional.

References

[1] Z. Abrams, A. Goel, and S. A. Plotkin. Set k-cover algorithms for energy efficient monitoring in wireless sensor networks. In *IPSN*, pages 424–432, 2004.

[2] G. Aloupis, J. Cardinal, S. Collette, S. Langerman, D. Orden, and P. Ramos. Decomposition of multiple coverings into more parts. *Discrete & Computational Geometry*, 44(3):706–723, 2010.

[3] A. Bar-Noy and B. Baumer. Maximizing network lifetime on the line with adjustable sensing ranges. In *ALGOSENSORS*, pages 28–41, 2011.

[4] A. Bar-Noy, B. Baumer, and D. Rawitz. Set It and Forget It: Approximating the Set Once Strip Cover Problem. *Arxiv preprint arXiv:1204.1082v1*, 2012.

[5] A. Buchsbaum, A. Efrat, S. Jain, S. Venkatasubramanian, and K. Yi. Restricted strip covering and the sensor cover problem. In *SODA*, pages 1056–1063, 2007.

[6] M. Cardei and D. Du. Improving wireless sensor network lifetime through power aware organization. *Wireless Networks*, 11(3):333–340, 2005.

[7] M. Cardei, J. Wu, and M. Lu. Improving network lifetime using sensors with adjustable sensing ranges. *Int. J. Sensor Networks*, 1(1/2):41–49, 2006.

[8] M. Cardei, J. Wu, M. Lu, and M. Pervaiz. Maximum network lifetime in wireless sensor networks with adjustable sensing ranges. In *WiMob (3)*, pages 438–445, 2005.

[9] E. Chambers, S. Fekete, H. Hoffmann, D. Marinakis, J. Mitchell, V. Srinivasan, U. Stege, and S. Whitesides. Connecting a set of circles with minimum sum of radii. In *WADS*, pages 183–194. Springer, 2011.

[10] W. J. Cook, W. H. Cunningham, and A. S. William R. Pulleyblank. *Combinatorial Optimization.* John Wiley & Sons, 1998.

[11] M. Gibson and K. Varadarajan. Decomposing coverings and the planar sensor cover problem. In *FOCS*, pages 159–168. IEEE, 2009.

[12] N. Lev-Tov and D. Peleg. Polynomial time approximation schemes for base station coverage with minimum total radii. *Computer Networks*, 47(4):489–501, 2005.

[13] J. Pach. Covering the plane with convex polygons. *Discrete & Computational Geometry*, 1(1):73–81, 1986.

[14] J. Pach and G. Tóth. Decomposition of multiple coverings into many parts. *Computational Geometry*, 42(2):127–133, 2009.

[15] M. Perillo and W. Heinzelman. Optimal sensor management under energy and reliability constraints. In *Wireless Communications and Networking*, volume 3, pages 1621–1626. IEEE, 2003.

[16] S. Slijepcevic and M. Potkonjak. Power efficient organization of wireless sensor networks. In *IEEE Intl. Conf. on Communications*, volume 2, pages 472–476, 2001.