Complexity of Barrier Coverage with Relocatable Sensors in the Plane ☆

Stefan Dobrev[a], Stephane Durocher[b], Mohsen Eftekhari[c,*], Konstantinos Georgiou[d], Evangelos Kranakis[e], Danny Krizanc[f], Lata Narayanan[c], Jaroslav Opatrny[c], Sunil Shende[g], Jorge Urrutia[h]

[a] *Institute of Mathematics, Slovak Academy of Sciences, Slovakia*
[b] *Dept. of Comp. Science, University of Manitoba, Canada*
[c] *Dept. of Comp. Science and Soft. Eng., Concordia University, Canada*
[d] *Dept. of Comb. and Opt., University of Waterloo, Canada*
[e] *School of Comp. Science, Carleton University, Canada*
[f] *Dept. of Math. and Comp. Science, Wesleyan University, USA*
[g] *Dept. of Comp. Science, Rutgers University, USA*
[h] *Institute of Mathematics, UNAM, Mexico*

Abstract

We consider several variations of the problems of covering a set of barriers (modeled as line segments) using sensors that can detect any intruder crossing any of the barriers. Sensors are initially located in the plane and they can *relocate* to the barriers. We assume that each sensor can detect any intruder in a circular area of fixed range centered at the sensor. Given a set of barriers and a set of sensors located in the plane, we study three problems: (i) the feasibility of barrier coverage, (ii) the problem of minimizing the largest relocation distance of a sensor (MinMax), and (iii) the problem of minimizing the sum of relocation distances of sensors (MinSum). When sensors are permitted to move to arbitrary positions on the barrier, the MinMax problem is shown to be strongly NP-complete for sensors with arbitrary ranges. We also study the case when sensors are restricted to use *perpendicular* movement to one of the barriers. We show that when the barriers are parallel, both the

☆Supported by VEGA 2/0136/12, NSERC Discovery, and SEP-CONACYT research grants. A short version of this paper was presented at the CIAC conference, Barcelona, 2013 [8].

*Corresponding author
Email address: m_eftek@cse.concordia.ca (Mohsen Eftekhari)

MinMax and MinSum problems can be solved in polynomial time. In contrast, we show that even the feasibility problem is strongly NP-complete if two perpendicular barriers are to be covered, even if the sensors are located at integer positions, and have only two possible sensing ranges. On the other hand, we give an $O(n^{3/2})$ algorithm for a natural special case of this last problem.

Keywords: Sensor network, mobile sensors, barrier coverage, algorithmic complexity, NP-completeness.

---

1. Introduction

The protection of a region by sensors against intruders is an important application of sensor networks that has been previously studied in several papers. Each sensor is typically considered to be able to sense an intruder in a circular region of fixed range around the sensor. Previous work on region protection using sensors can be classified into two major classes. In the first body of work, called *area coverage*, the monitoring of an entire region is studied [13, 16], and the presence of an intruder can be detected by a sensor anywhere in the region, either immediately after the appearance of an intruder, or within a fixed time delay. In the second body of work, called *barrier coverage*, a region is assumed to be protected by monitoring its perimeter, called the *barrier*, [1, 3, 6, 7, 15], and an intruder is detected when crossing the barrier. Clearly, the second approach is less expensive in terms of the number of sensors required, and it is sufficient in many applications.

There are two different approaches to barrier coverage in the literature. In the first approach, a barrier is considered to be a narrow strip of fixed width. Sensors are dispersed randomly on the barrier, and the probability of barrier coverage is studied based on the density of dispersal. Since random dispersal may leave gaps in the coverage, some authors propose using several rounds of random dispersal for complete barrier coverage [10, 20]. In the second approach, several papers assume that sensors, once dispersed, are mobile, and can be instructed to relocate from the initial position to a final position on the barrier in order to achieve complete coverage [2, 6, 7, 9, 14, 17, 18]. Clearly, when a sufficient number of sensors is used, this approach always guarantees complete coverage of the barrier. In order to minimize energy consumption by the mobile sensors, researchers have studied the problem of assigning final positions to the sensors that minimize some aspect of the relocation cost. The variations studied so far include centralized algorithms for minimizing the *maximum* relocation distance (*MinMax*) [6], the *sum* of relocation distances (*MinSum*) [7], or minimizing the *number* of sensors that relocate (*MinNum*) [17]; distributed algorithms for barrier coverage are considered in [9, 18].

Circular barriers are studied in [4, 19]. In [4], the authors considered moving n sensors to the perimeter of a given circular barrier to form a regular n-gon and provided an algorithm for MinMax problem that runs in $O(n^{3.5} \log n)$. In [19] the results of [4] for the MinMax problem are improved and an $O(n^{2.5} \log n)$ algorithm is given. Also, the authors introduced an algorithm that solves the MinSum problem when initial positions of sensors are on the perimeter of the circular barrier and runs in $O(n^4)$ time.

Most of the previous work on linear barriers is set in the one-dimensional setting: the barriers are assumed to be one or more line segments that are part of a line $\mathcal{L}$, and furthermore, the sensors are initially located on the same line $\mathcal{L}$. In [6] it was shown that there is an $O(n^2)$ algorithm for the MinMax problem in the case when the sensor ranges are identical. The authors also showed that the problem becomes NP-complete for sensors with arbitrary ranges if there are two barriers on $\mathcal{L}$. A polynomial time algorithm for the MinMax problem is given in [5] for arbitrary sensor ranges for the case of a single barrier, and an improved algorithm is given for the case when all sensor ranges are identical. In [7], it was shown that the MinSum problem is NP-complete when arbitrary sensor ranges are allowed, and an $O(n^2)$ algorithm is given when all sensing ranges are the same. Similarly as in the MinSum problem, the MinNum problem is NP-complete when arbitrary sensor ranges are allowed, and an $O(n^2)$ algorithm is given when all sensing ranges are the same [17].

In this paper we consider the algorithmic complexity of several natural generalizations of the barrier coverage problem with sensors of arbitrary ranges. We generalize the work in [5, 6, 7, 17] in two significant ways. First, we assume that the initial positions of sensors are arbitrary points in the two-dimensional plane, not necessarily on the line containing the barrier. This assumption is justified since in many situations, initial dispersal of sensors on the line containing the barrier might not be possible. Second, we consider multiple barriers that are parallel or perpendicular to each other. This generalization is motivated by barrier coverage of the perimeter of an area.

1.1. Preliminaries and Notation

Throughout the paper, we assume that we are given a set of sensors $S = \{s_1, s_2, \ldots, s_n\}$ located in the plane in positions $p_1, p_2, \ldots, p_n$, where $p_i = (x_i, y_i)$ for some real values x_i, y_i. The sensing ranges of the sensors are positive real values $r_1, r_2, \ldots, r_n$, respectively. A sensor s_i can detect any intruder in the closed circular area around p_i of radius r_i. We assume that sensor s_i is mobile and thus can relocate itself from its initial location p_i to another specified location p'_i. A *barrier* b is a closed line segment in the plane. Given a set of barriers $\mathcal{B} = \{b_1, b_2, \ldots, b_k\}$, and a set of sensors S of sensing ranges $r_1, r_2, \ldots, r_n$ with $\sum_{i=1}^{n} 2r_i \geq \sum_{i=1}^{k} |b_i|$, initially located at positions $p_1, p_2, \ldots, p_n$ in the plane, the *barrier coverage* problem is to determine for each i with $1 \leq i \leq n$, the final position of sensor s_i on one of the barriers denoted by p'_i, so that all barriers are collectively covered by the sensing ranges of the sensors. We call such an assignment of final positions a *covering assignment*. Figure 1 shows an example of a barrier coverage problem and a possible covering assignment. Motivated by reducing the energy consumption of sensors, we are interested in optimizing some measure of the movement of sensors involved to achieve coverage, in particular MinMax and MinSum. We use standard cost measures such as Euclidean or rectilinear distance. The distance between the initial position p and a final position p' of a sensor is denoted by $d(p, p')$.

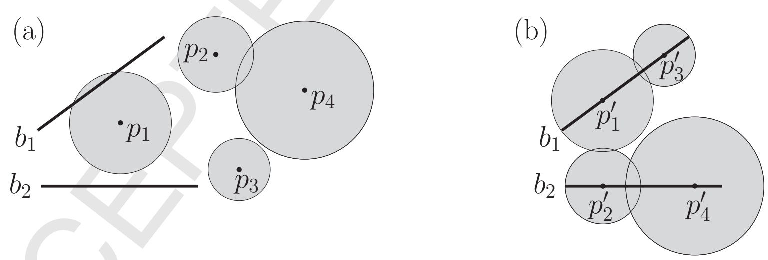

Figure 1: (a) A given barrier coverage problem for two barriers (b) a possible covering assignment

We are interested in the algorithmic complexity of three problems:

Feasibility problem: Given a set of sensors S located in the plane at positions $p_1, p_2, \ldots, p_n$, and a set of barriers $\mathcal{B}$, determine if there exists a valid covering assignment, i.e. determine whether there exist final node positions $p'_1, p'_2, \ldots, p'_n$ on the barriers such that all barriers in $\mathcal{B}$ are covered.

MinMax problem: Given a set of sensors S located in the plane at positions $p_1, p_2, \ldots, p_n$, and a set of barriers $\mathcal{B}$, find final node positions $p'_1, p'_2, \ldots, p'_n$ on the barriers so that all barriers in $\mathcal{B}$ are covered and $\max_{1\le i\le n}\{d(p_i, p'_i)\}$ is minimized.

MinSum problem: Given a set of sensors S located in the plane at positions $p_1, p_2, \ldots, p_n$, and a set of barriers $\mathcal{B}$, find final node positions $p'_1, p'_2, \ldots, p'_n$ on the barriers so that all barriers in $\mathcal{B}$ are covered, and $\sum_{i=1}^{n} d(p_i, p'_i)$ is minimized.

1.2. Our Results

Throughout the paper, we consider the barrier coverage problem with sensors of arbitrary ranges, initially located at arbitrary locations in the plane. In Section 2, we assume that sensors can move to arbitrary positions on any of the barriers. While feasibility is trivial in the case of one barrier, it is straightforward to show that it is NP-complete for even two barriers. The NP-completeness of the MinSum problem for one barrier follows trivially from the result in [7]. In this paper, we show that the MinMax problem is strongly NP-complete even for a single barrier. We show that this holds both when the cost measure is Euclidean distance and when it is rectilinear distance.

In light of these hardness results, in the rest of the paper, we consider a more restricted but natural movement of sensors. We assume that once a sensor has been ordered to relocate to a particular barrier, it moves to the *closest point* on a line containing a barrier. We call this *perpendicular movement*. Note that it is possible for a sensor that is not located on the barrier to cover part of the barrier. However, we require final positions of sensors to be on the line containing the barrier. Section 3.1 considers the case of one barrier and perpendicular movement, while Section 3.2 considers the case of perpendicular movement and multiple *parallel* barriers. We show that all three of our problems are solvable in polynomial time for any fixed number of barriers. Finally, in Section 4, we consider the case of perpendicular movement and two barriers perpendicular to each other. We show that even the feasibility problem is strongly NP-complete in this case. The NP-completeness result holds even in the case when the given positions of the sensors have integer values and the sensing ranges of sensors are limited to two different integer sensing ranges. In contrast, we give an $O(n^{1.5})$ algorithm for finding a covering assignment for a natural restriction of the problem that includes the case when all sensors are located in integer positions and the sensing ranges of all sensors are of diameter 1. Our results are summarized in Table 1 below.

Barriers	Movement	Feasibility	MinMax	MinSum
one	Arbitrary	$O(n)$	NPC	NPC [7]
two	Arbitrary	NPC	NPC	NPC
one	Perpendicular	$O(n)$	$O(n \log n)$	$O(n^2)$
k parallel	Perpendicular	$O(kn)$	$O(kn^{k+1})$	$O(kn^{k+1})$
2 perpendicular	Perpendicular	NPC	NPC	NPC

Table 1: Summary of our results.

2. Arbitrary Final Positions

In this section, we assume that sensors are allowed to relocate to any final positions on the barrier(s). We consider first the case of a single barrier b. Without loss of generality, we assume that b is located on the x-axis between $(0,0)$ and $(L,0)$ for some L. The feasibility of barrier coverage in this case is simply a matter of checking whether $\Sigma_{i=1}^{n} 2r_i \geq L$. For the MinSum problem, it was shown in [7] that even if the initial positions of sensors are on the line containing the barrier, the problem is NP-complete; therefore the more general version of the problem studied here is clearly NP-complete. Recently, it was shown in [5] that if the initial positions of sensors are on the line containing the barrier, the MinMax problem is solvable in polynomial time. We proceed to study the complexity of the MinMax problem when initial positions of sensors can be anywhere on the plane, and the final positions can be anywhere on the barrier. See Figure 2 for an example of the initial placement of sensors.

Theorem 1. *Let $S = \{s_1, s_2, \ldots, s_n\}$ be a set of sensors of ranges $r_1, r_2, \ldots, r_n$ initially located in the plane at positions $p_1, p_2, \ldots, p_n$ and $\sum_{i=1}^{n} 2r_i \geq L$. Let the barrier b be a line segment between $(0,0)$ and $(L,0)$. Given an integer k, the problem of determining if there is a covering assignment such that the maximum relocation distance (Euclidean/rectilinear) of the sensors is at most k is strongly NP-complete.*

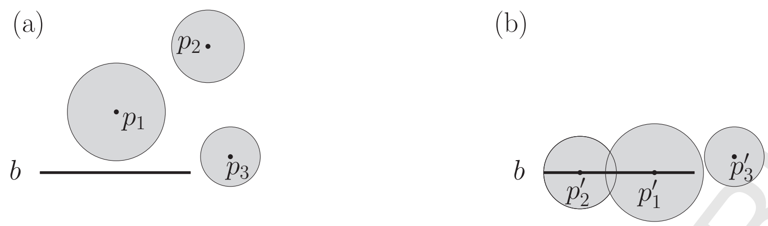

Figure 2: (a) A given barrier coverage problem for a single barrier (b) a possible covering assignment

Proof. The problem is trivially in NP; we give here a reduction from the 3-partition problem [11]. We are given a multiset $A = \{a_1 \geq a_2 \geq \cdots \geq a_{3m}\}$ of $3m$ positive integers such that $B/4 < a_i < B/2$ for $1 \leq i \leq 3m$ and $\sum_{i=1}^{3m} a_i = mB$ for some B. The problem is to decide whether A can be partitioned into m triples $T_1, T_2, \ldots, T_m$ such that the sum of the numbers in each triple is equal to B. We create an instance of the barrier coverage problem as follows: Let $L = mB + m - 1$, the barrier b be a line segment from $(0,0)$ to $(L,0)$, and let $k = L$. For every i with $1 \leq i \leq 3m$, we create a sensor s_i of range $a_i/2$ and place it at $(-a_i/2, 0)$. In addition, place $m-1$ sensors $s_{3m+1}, s_{3m+2}, \ldots, s_{4m-1}$ of range $1/2$ at positions $(B+1/2, k), (2B+3/2, k), (3B+5/2, k), \ldots, ((m-1)B + (2m-3)/2, k)$. See Figure 3 for an example. Since $L = \sum_{i=1}^{4m-1} 2r_i$, all sensors must move to the barrier in any covering assignment. Observe that the distance from any of the $m-1$ sensors located above the barrier to the barrier is k, and when all of them move this distance, there are gaps of length B between these sensors on the barrier.

If there is a partition of S into m triples $T_1, T_2, \ldots, T_m$, the sum of each triple being B, then there is a solution to the movement of the sensors such that the three sensors corresponding to triple T_i are moved to fill the ith gap in the barrier b. The maximal move of the three sensors corresponding to T_i into ith gap is at most L, and the maximum of the moves of all sensors is k in this case. If such a partition does not exist, then any covering assignment for the barrier b corresponds to moving at least one of the sensors above the x-axis by $k+1$ (rectilinear distance), and by $\sqrt{k^2+1} > k$ (Euclidean distance).

It remains to show that the transformation from the 3-partition problem to the sensor movement problem is polynomial. Since 3-partition is strongly NP-complete [11], we may assume that the values $a_1, a_2, \ldots, a_{3m}$ are bounded by a polynomial $c(3m)^j$ for some constants c and j. The 3-partition problem can be represented using $O(m \log m)$ bits. Therefore, $B \leq c_1 m^j$ and $k \leq c_2 m^{j+1}$ for some constants c_1 and c_2. Our reduction uses $n = 4m - 1$ sensors. In the corresponding barrier coverage problem we need $O(n \log n)$ bits for the positions and sizes of sensors $s_1, s_2, \ldots, s_n$ and we need $O(n \log n)$ bits to represent the position and size of each sensor of size 1. Thus we need $O(n \log n)$ bits to represent the corresponding barrier coverage problem, which shows that the transformation is polynomial.

By adding one additional sensor at distance $> k$ above the barrier, we can create an instance of the problem where $\sum_{i=1}^{4m-1} 2r_i > L$, and the proof remains exactly the same as that sensor cannot be involved in a covering assignment that has maximum relocation distance k.

Finally, it is straightforward to see that any given covering assignment can be verified in polynomial time, completing the proof of strong NP-completeness. □

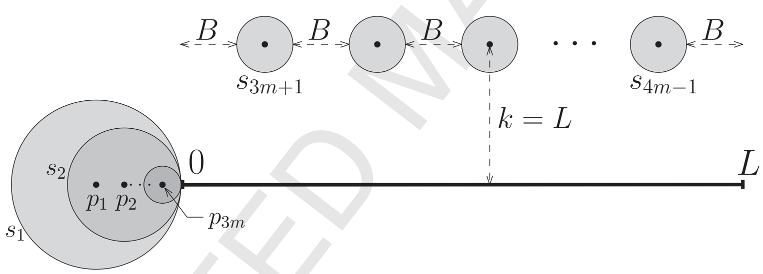

Figure 3: Reduction from 3-partition to the MinMax problem

It is easy to see that when there are two barriers to be covered, even feasibility of coverage is NP-complete. This can be shown by reducing the Partition problem [11] to an appropriate 2-barrier coverage problem, as in [6]. It follows that k-barrier coverage is also NP-complete.

3. Perpendicular Movement: Parallel Barriers

In the previous section, we showed that if sensor movements are not limited even the feasibility problem is NP-complete. In this section, we assume that all sensors use only perpendicular movement to a barrier. The motivation for considering this limited mobility comes from the fact that in perpendicular movement each sensor if assigned to a barrier will move minimum distance to reach it. In the case of several barriers, the barriers are *parallel* to each other. Figure 4 illustrates an example of such a problem. Without loss of generality, we assume barriers $b_1, b_2, \ldots, b_k$, $k \geq 1$ are parallel to the x-axis. Thus, sensors may only move in a vertical direction. Let the set of n sensors $s_1, s_2, \ldots, s_n$ be initially located at positions $p_1, p_2, \ldots, p_n$ respectively, where $p_i = (x_i, y_i)$. We assume that the sensors are listed in the order of the leftmost x-coordinates they can cover, i.e., $x_1 - r_1 \leq x_2 - r_2 \leq \cdots \leq x_n - r_n$. For simplicity we assume all points of interest (sensor locations, left/right endpoints of sensor ranges and barriers) are distinct.

Since there are k barriers, there are up to k points on barriers with the same x-coordinate. We therefore speak of sensors being *candidates for x-coordinates*: a sensor s in position $p = (x, y)$ with sensing range r is a candidate sensor for x-coordinate x' if $x - r \leq x' < x + r$. Alternatively we say *s potentially covers* the x-coordinate x'. Notice that according to our definition, the sensor s potentially covers a *half-open interval* of x-coordinates; this definition simplifies our algorithms. We first consider the simpler case of $k = 1$.

3.1. One Barrier

Without loss of generality, let the barrier $b = b_1$ be the line segment between $(0, 0)$ and $(L, 0)$. Since the y-coordinate of all points on the barrier are the same, we sometimes represent the barrier or a segment of the barrier by an interval of x-coordinates. For technical reasons, we denote the segment of the barrier between the points $(i, 0)$ and $(j, 0)$ by the *half-open interval* $[i, j)$.

We first show a necessary and sufficient condition on the sensors for the barrier to be covered. We give a dynamic programming formulation for the MinSum problem. We denote the set of sensors $\{s_i, s_{i+1}, \ldots, s_n\}$ by S_i. If the barrier is an empty interval, then the cost is 0. If no sensor is a candidate for the left endpoint of the barrier, or if the sensor set is empty while the barrier is a non-empty interval, then clearly the problem is infeasible and the cost is infinity. If not, observe that the optimal solution to the MinSum problem either involves moving sensor s_1 to the barrier or it doesn't. In the first case, the cost of the optimal solution is the sum of $|y_1|$, the cost of moving the first sensor to the barrier, and the optimal cost of the subproblem of covering the interval $[x_1 + r_1, L)$ with the remaining sensors $S_2 = S - \{s_1\}$. In the second case, the optimal solution is the optimal cost of covering the original interval

$[0, L)$ with S_2. The recursive formulation is given below:

$$cost(S_i, a) = \begin{cases} 0 & \text{if } L < a \\ \infty & \text{if } x_i - r_i > a \text{ or } (S_i = \emptyset \text{ and } L > a) \\ min \begin{cases} |y_i| + cost(S_{i+1}, x_i + r_i), \\ cost(S_{i+1}, a) \end{cases} & \text{otherwise} \end{cases}$$

Observe that a subproblem is always defined by a set S_i and a left endpoint to the barrier which is given by the rightmost x-coordinate covered by a sensor. Thus the number of possible subproblems is $O(n^2)$, and it takes constant time to compute $cost(S_i, a)$ given the solutions to the sub-problems. Using either a tabular method or memoization, the problem can be solved in quadratic time. The same dynamic programming formulation works for minimizing the maximum movement, except that in the case when the i-th sensor moves to the barrier in the optimal solution, the cost is the maximum of $|y_i|$ and $cost(S_{i+1}, x_i + r_i)$ instead of their sum. A better approach is to check the feasibility of covering the barrier with the subset of sensors at distance at most d from the barrier in $O(n)$ time, and find the minimum value of d using binary search on the set of distances of all sensors to the barrier. This yields an $O(n \log n)$ algorithm for MinMax.

Theorem 2. *Let $s_1, s_2, \ldots, s_n$ be n sensors initially located in the plane at positions $p_1, p_2, \ldots, p_n$ respectively, and let b be a barrier between $(0, 0)$ and $(L, 0)$. The MinSum problem using only perpendicular movement can be solved in $O(n^2)$ time, and the MinMax problem can be solved in $O(n \log n)$ time.*

3.2. Multiple Parallel Barriers

In this section we study the case of k equi-length barriers parallel to x axis with all left endpoints sharing same x-coordinate. For simplicity, we explain the case of two barriers; the results generalize to k barriers as stated in Theorem 3. Assume without loss of generality that the two barriers to be covered are b_1 between $(0, 0)$ and $(L, 0)$ and b_2 between (P, W) and (L, W), $0 \le P$. Note that we are proving a more general version of the problem where barriers may have different starting points because it is used in the recursive step of our proof. We assume that the sensing ranges of sensors are smaller than half the distance W between the two barriers, and thus it is impossible for a sensor to simultaneously cover two barriers. See Figure 4 for an example of such a problem.

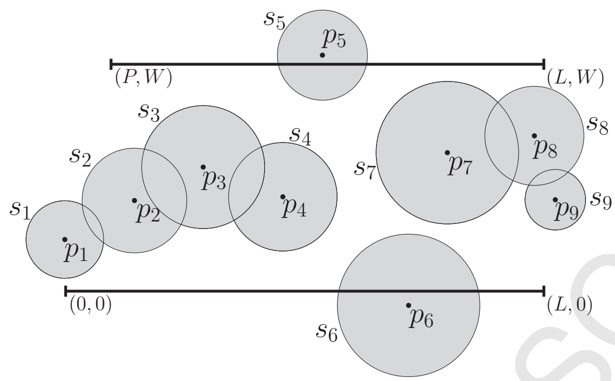

Figure 4: An example of a barrier coverage problem with two parallel barriers

Since there are two barriers, there are two points on barriers with the same x-coordinate. We say an interval $I = [a, b)$ of x-coordinates is *k-coverable* if every x-coordinate in the interval has k candidate sensors; such an interval of x-coordinates can exist on both barriers. We first show a necessary and sufficient condition on the sensors for the two barriers to be covered. Clearly, since the sensing range of every sensor is smaller than half of the distance between the two barriers, the barrier coverage problem for the two parallel barriers b_1 and b_2 above is solvable by a set of sensors S *only if* the interval $[0, P)$ is 1-coverable, and $[P, L)$ is 2-coverable by sensors in S. We proceed to show that this is also a sufficient condition, and give an $O(n)$ algorithm for finding a covering assignment for two parallel barriers.

Lemma 1. *Let $s_1, s_2, \ldots, s_n$ be sensors located at positions $p_1, p_2, \ldots, p_n$ respectively where $p_i = (x_i, y_i)$ and $x_1 - r_1 \leq x_2 - r_2 \leq \cdots x_n - r_n$. Let b_1 between $(0, 0)$ and $(L, 0)$ and b_2 between (P, W) and (L, W), where $0 \leq P < L$, be two parallel barriers to be covered. If interval $[0, P)$ is 1-coverable and $[P, L)$ is 2-coverable, then a covering assignment that uses only perpendicular movement of the sensors can be obtained in $O(n)$ time.*

Proof. We give an algorithm to find such a covering assignment. We assign a sensor from S to cover $(0, 0)$ on b_1. Clearly this is possible, since the interval of x-coordinates $[0, P)$ is 1-coverable. Let s be the sensor that was used in this assignment, of range r, and initially in position (x, y), so that its final position is $(x, 0)$ where $x - r \leq 0$. We remove s from S and consider the following cases:

$x + r \leq P$**:** It is easy to see that the interval of x-coordinates $[x + r, P)$ is 1-coverable and $[P, L)$ is 2-coverable.

$P < x + r < L$**:** Then since $[P, L)$ was initially 2-coverable, and s is the only unavailable sensor among all candidate sensors for this interval, it follows that the interval of x-coordinates $[P, x+r)$ is now 1-coverable and $[x + r, L)$ is 2-coverable.

$x + r \geq L$**:** Then we have a single barrier left and the interval of x-coordinates $[x + r, L)$ is 1-coverable, so we can use the algorithm of the previous section.

We now have a sub-problem of the same type as the original problem and proceed to solve it recursively. Since at every step of the algorithm, one of the sensors is assigned to cover one of the barriers in increasing order of the values $x_i - r_i$, the algorithm takes $O(n)$ time. □

It is easy to see that the lemma can be generalized for k barriers to show that the feasibility problem can be solved in $O(kn)$ time. We proceed to study the problem of minimizing the sum of movements required to perform barrier coverage.

The dynamic programming formulation given in Section 3.1 can be generalized for the case of two barriers. The key difference is that in an optimal solution, sensor s_i may be used to cover a part of barrier b_1 or barrier b_2 or neither. Let $xcost(S_i, a_1, a_2)$ denote the cost of covering the interval $[a_1, L)$ of the barrier b_1 and the interval $[a_2, L)$ of the second barrier with the sensor set $S_i = \{s_i, s_{i+1}, \ldots, s_n\}$. The optimal cost is given by the formulation below:

$$xcost(S_i, a_1, a_2) =$$
$$= \begin{cases} cost(S_i, a_2) & \text{if } L < a_1 \\ cost(S_i, a_1) & \text{if } L < a_2 \\ \infty & \text{if } x_i - r_i > min\{a_1, a_2\} \text{ or } (S_i = \emptyset \text{ and } L > min\{a_1, a_2\}) \\ min \begin{cases} |y_i| + xcost(S_{i+1}, x_i + r_i, a_2), \\ |W - y_i| + xcost(S_{i+1}, a_1, x_i + r_i), \\ xcost(S_{i+1}, a_1, a_2) \end{cases} & \text{otherwise} \end{cases}$$

It is not hard to see that the formulation can be generalized to k barriers with possibly different lengths; a sensor s_i may move to any of the k barriers with the corresponding cost being added to the solution. Observe that a subproblem is now given by a set S_i, and a left endpoint of each of the barriers to be covered. The total number of subproblems is $O(n^{k+1})$ and the time needed to compute the cost of a problem given the costs of the subproblems is $O(k)$. Thus, the time needed to solve the problem is $O(kn^{k+1})$. Clearly a very similar formulation as above can be used to solve the MinMax problem in $O(kn^{k+1})$ time as well. We have proved the following theorem.

Theorem 3. *Let $s_1, s_2, \ldots, s_n$ be n sensors initially located in the plane at positions $p_1, p_2, \ldots, p_n$ respectively, where $p_i = (x_i, y_i)$ and $x_1 - r_1 \leq x_2 - r_2 \leq \cdots x_n - r_n$. Both the MinSum problem and the MinMax problem for k parallel barriers using only perpendicular movement can be solved in $O(kn^{k+1})$ time.*

4. Perpendicular Movement: Two Perpendicular Barriers

In this section we consider the problem of covering two perpendicular barriers. Once again, we assume that sensors can relocate to either of the two barriers, using only perpendicular movement. Figure 5 illustrates an example of such a problem. In contrast to the case of parallel barriers, we show here that even the feasibility problem in this case is NP-complete. For simplicity we assume that b_1 is a segment on the x-axis between $(0,0), (L_1, 0)$ and b_2 is a segment on the y-axis between $(0,0), (0, L_2)$. Since the sensors can only employ perpendicular movement, the only possible final positions on the barriers for a sensor s_i in position $p_i = (x_i, y_i)$ are $p'_i = (0, y_i)$ or $p'_i = (x_i, 0)$.

We first show that the feasibility problem for this case is NP-complete by giving a reduction from the monotone 3-SAT problem [12]. Recall that a Boolean 3-CNF formula $f = c_1 \wedge c_2 \wedge ... \wedge c_m$ of m clauses is called *monotone* if and only if every clause c_i in f either contains only unnegated literals or only negated literals. In order to obtain a reduction to a barrier coverage problem with two perpendicular barriers, we first put a monotone 3-SAT formula in a special form as given in the lemma below.

Lemma 2. *Let $f = f_1 \wedge f_2$ be a monotone 3-CNF Boolean formula with n clauses where f_1 and f_2 only contain unnegated and negated literals respectively, and every literal appears in at most m clauses. Then f can be transformed into a monotone formula $f' = f'_1 \wedge f'_2$ such that f'_1 and f'_2 have only unnegated and negated literals respectively, and f' has the following properties:*

1. *f and f' are equisatisfiable, i.e. f' is satisfiable if and only if f is satisfiable.*
2. *All clauses are of size two or three.*
3. *Clauses of size two contain exactly one variable from f and one new variable.*
4. *Clauses of size three contain only new variables.*
5. *Each new literal appears exactly once: either in a clause of size two or in a clause of size three.*
6. *Each variable x_i of f appears exactly in m clauses of f'_1, and exactly in m clauses of f'_2.*
7. *f' contains at most $3mn$ clauses.*
8. *The clauses in f'_1 (respectively f'_2) can be ordered so that all clauses containing the literal x_i ($\overline{x_i}$) appear before clauses containing the literal x_j (respectively $\overline{x_j}$) for $i < j$, and all clauses of size three are placed last.*

Proof. Let $f = f_1 \wedge f_2$ be a monotone 3-CNF Boolean formula, where f_1 only contains unnegated literals and f_2 only contains negated literals. Assume the clauses are numbered from 1 to n, and let m be the maximum number of occurrences of any literal in f. For each unnegated literal x_p that appears in the clause numbered i, we introduce a new variable $x_{p,i}$; suppose there are k such variables where $1 \leq k \leq m$. If $k < m$, we also introduce $m - k$ new variables $y_{p,1}, y_{p,2}, \ldots, y_{p,m-k}$. Similarly, for each negated literal $\overline{x_p}$ that appears in the clause numbered j in f_2, we introduce a new variable $x_{p,j}$; suppose there are k such variables where $1 \leq k \leq m$. If $k < m$, we also introduce $m - k$ new variables $z_{p,1}, z_{p,2}, \ldots, z_{p,m-k}$.

For each clause $c_i \in f_1$ of the form $(x_p \vee x_q \vee x_r)$, we put the collection of clauses $(x_p \vee x_{p,i}), (x_q \vee x_{q,i}), (x_r \vee x_{r,i})$ into f'_1 and the clause $(\overline{x_{p,i}} \vee \overline{x_{q,i}} \vee \overline{x_{r,i}})$ into f'_2. Similarly for each clause $c_j \in f_2$ of the form $(\overline{x_p} \vee \overline{x_q} \vee \overline{x_r})$, we put the collection of clauses $(\overline{x_p} \vee \overline{x_{p,j}}), (\overline{x_q} \vee \overline{x_{q,j}}), (\overline{x_r} \vee \overline{x_{r,j}})$ into f'_2 and the clause $(x_{p,j} \vee x_{q,j} \vee x_{r,j})$ into f'_1.

For every literal $x_p \in f_1$ that occurs $k < m$ times in f_1, we add clauses $(x_p \vee y_{p,1}) \wedge (x_p \vee y_{p,2}) \wedge \cdots (x_p \vee y_{p,m-k})$. Similarly, for every literal $\overline{x_p}$ that occurs $k < m$ times in f_2, we add clauses $(\overline{x_q} \vee \overline{z_{q,1}}) \wedge \cdots \wedge (\overline{x_q} \vee \overline{z_{q,m-k}})$. Finally, let $f' = f'_1 \wedge f'_2$. From the construction of f' it is easy to verify that it has Properties 2 to 7 stated in the lemma. Property 8 follows from Property 2, 3, and 4.

Now we show that f and f' are equisatisfiable. First assume f is satisfiable, and let A be a satisfying assignment for f. We show how to obtain a satisfying assignment A' for f'. For every variable x_p in f, A' uses

(a) the same truth assignment for x_p as in A,

(b) the opposite truth value for all new variables $x_{p,i}$,

(c) the truth value **true** for every new variable of the type $y_{p,i}$, and

(d) the truth value **false** for every new variable of the type $z_{p,i}$.

To see that A' satisfies f', observe that all clauses of size two in f'_1 are of the form $(x_p \vee x_{p,i})$ or $(x_p \vee y_{p,i})$ and are clearly satisfied. The only clauses of size three in f'_1 are of type $(x_{p,i} \vee x_{q,i} \vee x_{r,i})$ and correspond to a clause $c_i = (\overline{x_p} \vee \overline{x_q} \vee \overline{x_r})$ in f_2. Since A satisfies c_i, one of x_p, x_q, x_r must be false. But then one of $x_{p,i}, x_{q,i}, x_{r,i}$ must be true in A', and hence the clause $(x_{p,i} \vee x_{q,i} \vee x_{r,i})$ is satisfied. A similar argument can be made about the clauses in f'_2.

Next assume that f' is satisfiable, and let A' be a satisfying assignment for f'. We claim that taking the assignment for the original variables x_p in A' will also satisfy f. To see this, consider the clause $c_i = (x_p \vee x_q \vee x_r)$ in f_1. In f'_2 there is a corresponding clause $(\overline{x_{p,i}} \vee \overline{x_{q,i}} \vee \overline{x_{r,i}})$. Since A' satisfies this clause, at least one of $x_{p,i}, x_{q,i}, x_{r,i}$ must be false. Suppose $x_{p,i}$ is false. To satisfy the clause $(x_p \vee x_{p,i})$ in f'_1, the truth value of x_p in A' must be true. Thus the clause $c_i = (x_p \vee x_q \vee x_r)$ is satisfied in f_1. A similar argument can be made about the clauses in f_2. $\square$

We give an example that illustrates the reduction and the ordering specified in Property 8.

Example 1: Consider 3-CNF formula

$$f = (x_1 \vee x_3 \vee x_4) \wedge (x_2 \vee x_3 \vee x_4) \wedge (x_1 \vee x_2 \vee x_3) \wedge (\overline{x}_1 \vee \overline{x}_2 \vee \overline{x}_4) \wedge (\overline{x}_2 \vee \overline{x}_3 \vee \overline{x}_4)$$

An equisatisfiable formula f' satisfying the properties of Lemma 2 is:

$$\begin{aligned}
f' \ &= (x_1 \vee x_{1,1}) \wedge (x_1 \vee x_{1,3}) \wedge (x_1 \vee y_{1,1}) \wedge (x_2 \vee x_{2,2}) \wedge (x_2 \vee x_{2,3}) \\
\wedge \ & (x_2 \vee y_{2,1}) \wedge (x_3 \vee x_{3,1}) \wedge (x_3 \vee x_{3,2}) \wedge (x_3 \vee x_{3,3}) \wedge (x_4 \vee x_{4,1}) \\
\wedge \ & (x_4 \vee x_{4,2}) \wedge (x_4 \vee y_{4,1}) \wedge (x_{1,4} \vee x_{2,4} \vee x_{4,4}) \wedge (x_{2,5} \vee x_{3,5} \vee x_{4,5}) \\
\wedge \ & (\overline{x_1} \vee \overline{x_{1,4}}) \wedge (\overline{x_1} \vee \overline{z_{1,1}}) \wedge (\overline{x_1} \vee \overline{z_{1,2}}) \wedge (\overline{x_2} \vee \overline{x_{2,4}}) \wedge (\overline{x_2} \vee \overline{x_{2,5}}) \wedge (\overline{x_2} \vee \overline{z_{2,1}}) \\
\wedge \ & (\overline{x_3} \vee \overline{x_{3,5}}) \wedge (\overline{x_3} \vee \overline{z_{3,1}}) \wedge (\overline{x_3} \vee \overline{z_{3,2}}) \wedge (\overline{x_4} \vee \overline{x_{4,4}}) \wedge (\overline{x_4} \vee \overline{x_{4,5}}) \wedge (\overline{x_4} \vee \overline{z_{4,1}}) \\
\wedge \ & (\overline{x_{1,1}} \vee \overline{x_{3,1}} \vee \overline{x_{4,1}}) \wedge (\overline{x_{2,2}} \vee \overline{x_{3,2}} \vee \overline{x_{4,2}}) \wedge (\overline{x_{1,3}} \vee \overline{x_{2,3}} \vee \overline{x_{3,3}})
\end{aligned}$$

Theorem 4. *Let $s_1, s_2, \ldots, s_n$ be n sensors of sensing ranges $r_1, r_2, \ldots, r_n$, initially located in the plane at positions $p_1, p_2, \ldots, p_n$ respectively, and let b_1 between $(0, 0)$ and $(L_1, 0)$ and b_2 between $(0, 0)$ and $(0, L_2)$ be the two perpendicular barriers to be covered. Then the problem of finding a covering assignment using perpendicular movement for the two barriers is strongly NP-complete.*

Proof. It is easy to see that any given covering assignment can be verified in polynomial time. Given a monotone 3-SAT formula f, we use the construction described in Lemma 2 to obtain a formula $f' = f'_1 \wedge f'_2$ satisfying the properties stated in Lemma 2 with clauses ordered as described in Property 8. Let f_1 have i_1 clauses, and f_2 have i_2 clauses, and assume the clauses in each are numbered from $1, \ldots, i_1$ and $1, \ldots, i_2$ respectively. We create an instance P of the barrier coverage problem with two barriers b_1, the line segment between $(0, 0)$ and $(2i_1, 0)$ and b_2, the line segment between $(0, 0)$, and $(0, 2i_2)$.

For each variable x_i of the original formula f we have a sensor s_i of sensing range m located in position $p_i = ((2i - 1)m, (2i - 1)m)$, i.e., on the diagonal. Each of the variables $x_{i,j}, y_{i,j}, z_{i,j}$ is represented by a sensor of sensing range 1, denoted $s_{i,j}$, $s'_{i,j}$, and $s''_{i,j}$ respectively, and is placed in such a manner that the sensors corresponding to variables associated with the same s_i collectively cover the same parts of the two barriers as covered by sensor s_i. Furthermore, sensors corresponding to variables that appear in the same clause of size three cover exactly the same segment of a barrier. A sensor corresponding to a new variable $x_{i,j}$ that occurs in the pth clause in f'_1 and in the qth clause in f'_2 is placed in position $(2p - 1, 2q - 1)$. Also a sensor corresponding to variable $y_{i,j}$ which occurs in the ℓth clause in f'_1 is placed in position $(2\ell - 1, -1)$ and sensor corresponding to variable $z_{i,j}$ which occurs in the ℓth clause of f'_2 is placed in position $(-1, 2\ell - 1)$.

Figure 5 illustrates the instance of barrier coverage corresponding to the monotone 3-SAT formula from Example 4 above. The sensor $s_{1,3}$ corresponding to the variable $x_{1,3}$ appears in the second clause of f'_1 and the fifteenth clause of f'_2, and hence is placed at position $(3, 29)$. Similarly, the sensor $s_{2,4}$ corresponding to the variable $x_{2,4}$ appears in the thirteenth clause of f'_1 and the fourth clause of f'_2, and hence is placed at position $(25, 7)$.

It is easy to see that the reduction is polynomial, and the sensor sizes and border length are linear in the length of the input to the barrier coverage problem.

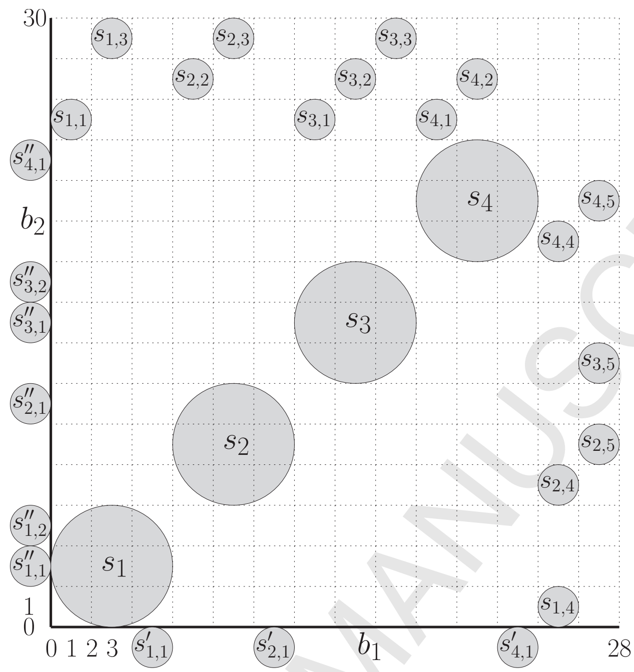

Figure 5: Barrier coverage instance corresponding to Example 1

Observe that in this assignment of positions to sensors, there is a one-to-one correspondence between the line segments of length 2 in b_1 and b_2 and clauses in f'_1 and f'_2 respectively. In particular, the sensors that potentially cover the line segment from $(2i-2, 0)$ to $(2i, 0)$ on the barrier b_1 correspond to variables in clause i of f'_1. Similarly, the sensors that potentially cover the line segment from $(0, 2i-2)$ to $(0, 2i)$ on the barrier b_2 correspond to variables in clause i of f'_2. Thus, by associating the vertical move of a sensor with an assignment of **true** to the corresponding variable of f', and the horizontal move of a sensor with an assignment of **false** to the corresponding variable of f', f' is satisfiable if and only if for the corresponding instance P there exists a covering assignment assuming perpendicular movement. □

Since any instance of monotone 3-SAT problem can be transformed into an instance of monotone SAT problem in which no literal occurs more than three times, it follows from the proof that the problem is NP-complete even when the sensors are in integer positions and the ranges of sensors are limited to two different sizes 1 and $m \geq 3$.

Corollary 5. *Let $s_1, s_2, \ldots, s_n$ be n sensors initially located in the plane* *at positions $p_1, p_2, \ldots, p_n$ respectively, and let b_1 between $(0, 0)$ and $(L_1, 0)$ and b_2 between $(0, 0)$ and $(0, L_2)$ be the two perpendicular barriers to be covered. Then the problem of finding a covering assignment using perpendicular movement for the two barriers is strongly NP-complete even if the ranges of sensors are limited to two different values.*

It is also clear from the proof that the perpendicularity of the barriers is not critical. The key issue is that the order of intervals covered by the sensors in one barrier has no relationship to those covered in the other barrier. In the case of parallel barriers, this property does not hold. The exact characterization of barriers for which a polytime algorithm is possible remains an open question.

We now turn our attention to restricted versions of barrier coverage of two perpendicular barriers where a polytime algorithm is possible. For a set of sensors S, and perpendicular barriers b_1, b_2, we call (S, b_1, b_2) a *non-overlapping arrangement* if for any two sensors $s_i, s_j \in S$, the intervals that are potentially covered by s_1 and s_2 on the barrier b_1 (and b_2) are either the same or disjoint. An example of a non-overlapping arrangement is shown in Figure 6. This would be the case, for example, if all sensor ranges are of

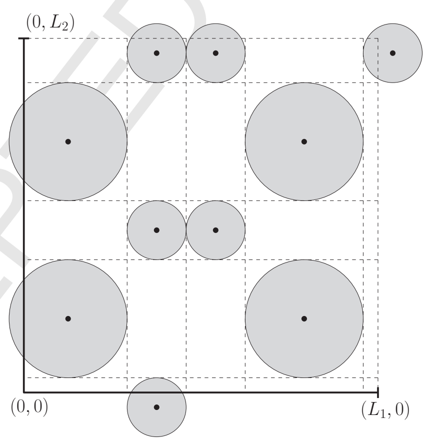

Figure 6: A non-overlapping arrangement of sensors. Each interval on the x-axis and y-axis delineated by dotted lines is represented by a node in the corresponding bipartite graph.

the same diameter equal to 1 and the sensors are in integer positions. We show below that for a non-overlapping arrangement, the problem of finding a covering assignment is polynomial.

Theorem 6. *Let $S = \{s_1, s_2, \ldots, s_n\}$ be a set of n sensors initially located in the plane at positions $p_1, p_2, \ldots, p_n$ and let b_1 and b_2 be two perpendicular barriers to be covered. If (S, b_1, b_2) form a non-overlapping arrangement, then there exists an $O(n^{1.5})$ algorithm that finds a covering assignment, using only perpendicular movement or reports that none exists.*

Proof. If there exists a segment of either of the barriers that is not covered by any of the sensors, then clearly there is no covering assignment. Otherwise, the problem of finding a covering assignment in this case can be reduced to the problem of maximum matching in a bipartite graph. Create one node for each sensor and one node for each segment of each barrier that is potentially covered by a sensor. Since (S, b_1, b_2) is a non-overlapping arrangement, the segments are disjoint and together they cover both barriers (see Figure 6). We put an edge between a node representing a barrier segment and a node representing a sensor if the sensor can cover the segment. Clearly, the problem of finding a covering assignment is equivalent to finding a matching in which each node representing a segment of the barrier is matched with a node representing a sensor. Since each node representing a sensor has degree two, this can be done in time $O(n^{1.5})$ using the Hopcroft-Karp algorithm. □

5. Conclusions

It was previously shown that the MinMax barrier coverage problem when the sensors are initially located on the line containing the barrier is solvable in polynomial time [5]. In contrast, our results show that the same problem becomes strongly NP-complete when sensors of arbitrary ranges are initially located in the plane, and are allowed to move to any final positions on the barrier. It remains open whether this problem is polynomial in the case when there is a fixed number of possible sensor ranges. If sensors are restricted to use perpendicular movement, the feasibility, MinMax, and MinSum problems are all polytime solvable for the case of k parallel barriers. However, when the barriers are not parallel, even the feasibility problem is strongly NP-complete, even when sensor ranges are restricted to two distinct values. It would be therefore interesting to study approximation algorithms for MinMax and MinSum for this case. Characterizing the problems for which barrier coverage is achievable in polytime remains an open question.

6. Acknowledgement

The authors would like to thank the anonymous reviewers for their helpful and constructive comments.

References

[1] P. Balister, B. Bollobas, A. Sarkar, and S. Kumar. Reliable density estimates for coverage and connectivity in thin strips of finite length. In *Proceedings of MobiCom '07*, pages 75–86. ACM, 2007.

[2] A. Bar-Noy, D. Rawitz, and P. Terlecky. Maximizing barrier coverage lifetime with mobile sensors. In *Proceedings of Algorithms – ESA '13, LNCS v. 8125*, pages 97–108, 2013.

[3] B. Bhattacharya, M. Burmester, Y. Hu, E. Kranakis, Q. Shi, and A. Wiese. Optimal movement of mobile sensors for barrier coverage of a planar region. *Theoretical Computer Science*, 410(52):5515–5528, 2009.

[4] B. K. Bhattacharya, B. Burmester, Y. Hu, E. Kranakis, Q. Shi, and A. Wiese. Optimal movement of mobile sensors for barrier coverage of a planar region. In *Proceedings of COCOA '08*, pages 103–115, 2008.

[5] D. Z. Chen, Y. Gu, J. Li, and H. Wang. Algorithms on minimizing the maximum sensor movement for barrier coverage of a linear domain. In *Proceedings of SWAT '12*, pages 177–188, 2012.

[6] J. Czyzowicz, E. Kranakis, D. Krizanc, I. Lambadaris, L. Narayanan, J. Opatrny, L. Stacho, J. Urrutia, and M. Yazdani. On minimizing the maximum sensor movement for barrier coverage of a line segment. In *Proceedings of ADHOC-NOW, LNCS v. 5793*, pages 194–212, 2009.

[7] J. Czyzowicz, E. Kranakis, D. Krizanc, I. Lambadaris, L. Narayanan, J. Opatrny, L. Stacho, J. Urrutia, and M. Yazdani. On minimizing the sum of sensor movements for barrier coverage of a line segment. In *Proceedings of ADHOC-NOW, LNCS v. 6288*, pages 29–42, 2010.

[8] S. Dobrev, S. Durocher, K. Georgiou, M. Eftekhari, E. Kranakis, D. Krizanc, L. Narayanan, J. Opatrny, S. Shende, and J. Urrutia. Complexity of barrier coverage with relocatable sensors in the plane. In

Proceedings of CIAC 2013 conference, LNCS v. 7878, pages 170–182, 2013.

[9] M. Eftekhari, E. Kranakis, D. Krizanc, O. Morales-Ponce, L. Narayanan, J. Opatrny, and S. Shende. Distributed local algorithms for barrier coverage using relocatable sensors. In *Proceeding of ACM PODC Symposium*, pages 383–392, 2013.

[10] M. Eftekhari, L. Narayanan, and J. Opatrny. On multi-round sensor deployment for barrier coverage. In *Proceedings of 10th IEEE MASS*, pages 310–318, 2013.

[11] M. R. Garey and D. S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1990.

[12] E. M. Gold. Complexity of automaton identification from given data. *Information and Control*, 37(3):302–320, 1987.

[13] C. F. Huang and Y. C. Tseng. The coverage problem in a wireless sensor network. In *WSNA '03: Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications*, pages 115–121. ACM, 2003.

[14] E. Kranakis, D. Krizanc, O. Morales-Ponce, L. Narayanan, J. Opatrny, and S. Shende. Expected sum and maximum of displacement of random sensors for coverage of a domain: Extended abstract. In *Proceedings of the Twenty-fifth Annual ACM Symposium on Parallelism in Algorithms and Architectures (SPAA '13)*, pages 73–82, 2013.

[15] S. Kumar, T. H. Lai, and A. Arora. Barrier coverage with wireless sensors. In *Proceedings of MobiCom '05*, pages 284–298. ACM, 2005.

[16] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M.B. Srivastava. Coverage problems in wireless ad-hoc sensor networks. In *Proceedings of INFOCOM, vol, 3*, pages 1380–1387, 2001.

[17] M. Mehrandish, L. Narayanan, and J. Opatrny. Minimizing the number of sensors moved on line barriers. In *Proceedings of IEEE WCNC '11*, pages 1464–1469, 2011.

[18] Ch. Shen, W. Cheng, X. Liao, and Sh. Peng. Barrier coverage with mobile sensors. In *Proceedings of International Symposium on Parallel Architectures, Algorithms, and Networks (ISPAN '08)*, pages 99–104, 2008.

[19] X. Tan and G. Wu. New algorithms for barrier coverage with mobile sensors. In *Proceedings of Frontiers of Algorithmics Workshop (FAW '10), LNCS v. 6213*, pages 327–338, 2010.

[20] G. Yan and D. Qiao. Multi-round sensor deployment for guaranteed barrier coverage. In *Proceedings of IEEE INFOCOM '10*, pages 2462–2470, 2010.