Complexity of Barrier Coverage with Relocatable Sensors in the Plane*

S. Dobrev S. Durocher K. Georgiou M. E. Hesari E. Kranakis
D. Krizanc, L. Narayanan J. Opatrny S. Shende J. Urrutia

November 6, 2012

Abstract

We consider several variations of the problems of covering a set of barriers using sensors so that sensors 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 centered at the sensor. Given a set of barriers and a set of sensors located in the plane, we study three problems: the feasibility of barrier coverage, the problem of minimizing the largest relocation distance of a sensor (MinMax), and 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 problems are shown to be NP-hard. We also study the case when sensors use *perpendicular* movement to one of the barriers, thereby moving to the closest point on the barrier. We show that when the barriers are parallel, both the MinMax and MinSum problems can be solved in polynomial time. In contrast, we show that even the feasibility problem is NP-complete if two perpendicular barriers are to be covered, even if the sensors are located at integer positions, and have only two possible sizes. On the other hand, we give an $O(n^{3/2})$ algorithm for the case when the sensors form a non-overlapping arrangement.

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 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 [8, 10, 11], and the presence of an intruder can be detected by a sensor anywhere in the region, either immediately after an 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 or barrier [1, 2, 4, 5, 9], and an intruder is detected when crossing the barrier. Clearly, the second approach is less costly 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. It has been shown that when the barrier is sufficiently long, one random dispersal leaves gaps in the coverage, and thus several rounds of dispersal are needed to assure complete barrier coverage [13]. Since random sensor dispersal causes incomplete coverage, in the second approach, several papers

*This research was started at the Routing in Merida workshop in 2009.

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 [4, 5]. Clearly, when a sufficient number of sensors is used, this approach always guarantees complete coverage of the barrier. The problem therefore is assigning final positions to the sensors in order to minimize some aspect of the relocation cost. The variations studied so far include minimizing the *maximum* relocation distance (*MinMax*), the *sum* of relocation distances of sensors (*MinSum*), or minimizing the *number* of sensors that relocate (*MinNum*).

Most of the previous work 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 [4] , 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 if there are two barriers. In [3] a polynomial time algorithm for the MinMax problem is given 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 [5], 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. Minimizing the number of sensors moved (*MinNum problem*) was considered in [12]. 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.

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 [3, 4, 5, 12] in two significant ways. First, we assume that the initial locations of sensors are points in the two-dimensional plane and are not necessarily collinear. Second, we consider multiple barriers that are parallel or perpendicular to each other, rather than being required to be on the same line. We consider two types of sensor movements. In the first part of the paper, we assume that sensors can move to arbitrary final positions on any of the barriers. We use standard cost measures such as Euclidean or rectilinear distance between initial and final positions of sensors. In the second part of the paper, we assume that sensors use *perpendicular movement*, that is, once having been assigned a barrier to relocate to, a sensor will take the shortest path to the barrier, and relocate to the *closest* point on the barrier.

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 $r_1, r_2, \ldots, r_n$, respectively. A sensor s_i can detect any intruder in the closed circular area around x_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 in positions $p_1, p_2, \ldots, p_n$ in the plane, of sensing ranges $r_1, r_2, \ldots, r_n$, the *barrier coverage* problem is to to determine for each s_i its final position p'_i on one of the barriers, so that all barriers are 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. Sometimes we are also interested in optimizing some measure of the movement of sensors involved to achieve coverage.

We are interested in the algorithmic complexity of three problems:

Feasibility problem: Given a set of sensors S located at positions $p_1, p_2, \ldots, p_n$, and a set of

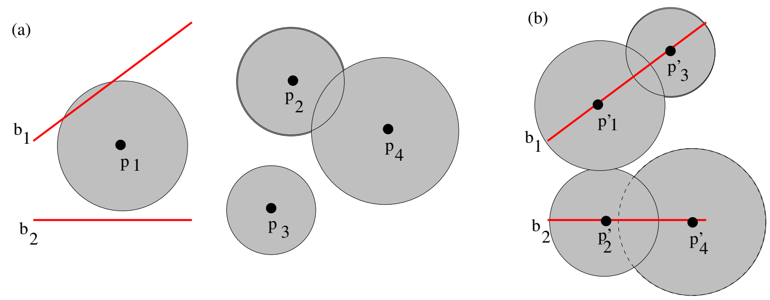

Figure 1: (a) A given barrier coverage problem (b) a possible covering assignment

barriers $\mathcal{B}$, determine if there exists a valid covering assignment, i.e. determine whether there exist final positions $p'_1, p'_2, \dots p'_n$ such that all barriers in $\mathcal{B}$ are covered.

MinMax problem: Given a set of sensors S located at positions $p_1, p_2, \dots, p_n$, and a set of barriers $\mathcal{B}$, find final positions $p'_1, p'_2, \dots p'_n$ 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 at positions $p_1, p_2, \dots, p_n$, and a set of barriers $\mathcal{B}$, find final positions $p'_1, p'_2, \dots p'_n$ 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

Our results are summarized in Table 1.

Barriers	Movement	Feasibility	MinMax	MinSum
1 barrier	Arbitrary final positions	$O(n)$	NP-hard	NP-hard [5]
2 barriers	Arbitrary final positions	NP-hard	NP-hard	NP-hard
1 barrier	Perpendicular	$O(n)$	$O(n \log n)$	$O(n^2)$
k parallel barriers	Perpendicular	P	P	P
2 perpendicular barriers	Perpendicular	NP-hard	NP-hard	NP-hard

Table 1: Barrier coverage problems: Initial positions on the plane, final positions on barriers

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-hard for even two barriers. The NP-hardness of the MinSum problem for one barrier follows trivially from the result in [5]. In this paper, we show that the MinMax problem is NP-hard 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. We assume that once a sensor has been ordered to relocate to a particular barrier, it moves to the closest point on the barrier. We call this *perpendicular movement.* Section 3 considers the case of one barrier and perpendicular movement, while Section 4 considers the case of perpendicular movement and multiple *parallel* barriers. We show that all three of our problems are solvable in polynomial time. Finally, in Section 5, we consider the case of perpendicular movement and two barriers perpendicular to each other. We show that even the feasibility problem is NP-complete in this case. The NP-hardness result holds even in the case when the given positions of the sensors have integer values and the sizes of sensors are limited to two different sizes. In contrast, we give an $O(n^{1.5})$ algorithm for finding a covering assignment in the case when the sensors form a non-overlapping arrangement. This is the case or example, when all sensors are located in integer positions and the sensing ranges of all sensors are of diameter 1.

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 standard measures for the cost of relocation, such as Euclidean distance or rectilinear distance.

2.1 Single Barrier

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 if $\Sigma_{i=0}^{n} 2r_i \geq L$. For the MinSum problem, it was shown in [5] that even if the initial positions of sensors are on the line containing the barrier, the problem is NP-hard; therefore the more general version of the problem studied here is clearly NP-hard. Recently, it was shown in [3] that if the initial positions of sensors are on the line containing the barrier, the MinMax problem is solvable in polynomial time. The complexity of the MinMax problem for general initial positions in the plane has not yet been studied.

MinMax

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.

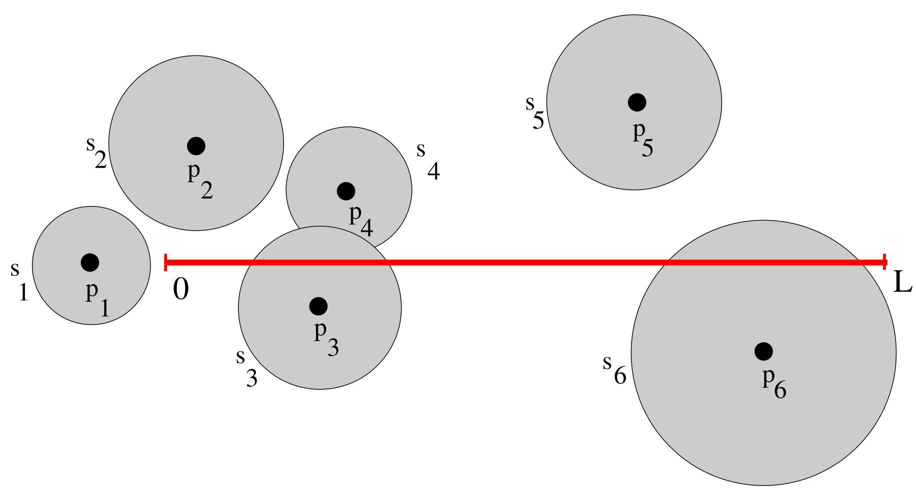

Figure 2: (a) An example of a 2-dimensional one barrier problem.

Theorem 1 *Let $S = \{s_1, s_2, \ldots, s_n\}$ be a set of n sensors with ranges $r_1, r_2, \ldots, r_n$ located in the plane in initial positions $p_1, p_2, \ldots, p_n$. Let the barrier b be a line segment on the x-axis 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 NP-hard.*

Proof. Let $R = \sum_{i=1}^{n} 2r_i$. Clearly if $R < L$ the problem is infeasible, so we assume $R \geq L$. We first give the proof for the case $R = L$. We prove it by reducing the 3-partition problem (see [6]) to the problem of covering the barrier b with sensors such that the maximum movement of the sensors is at most k. The 3-partition problem is defined as follows: we are given a multiset $A = \{a_1 \geq a_2 \geq \cdots \geq a_n\}$ of $n = 3m$ positive integers such that $B/4 < a_i < B/2$ for $1 \leq i \leq n$ and $\sum_{i=1}^{n} a_i = mB$ 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 $L = mB + m - 1$ so that the barrier b is a line segment from $(0,0)$ to $(L,0)$, and let $k = L+1$ Create a sensor s_i of range a_i for every $1 \leq i \leq 3m$ positioned at $-a_i$. In addition, create $m-1$ sensors $s_{3m+1}, s_{3m+2}, \ldots s_{4m-1}$ of range $1/2$ located 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 $R = L$, all sensors must move to the barrier. Observe that the distance from any of the $m-1$ sensors located above the barrier to the barrier is k, and even if all of them move this distance, there would be gaps of length B between these sensors on the barrier.

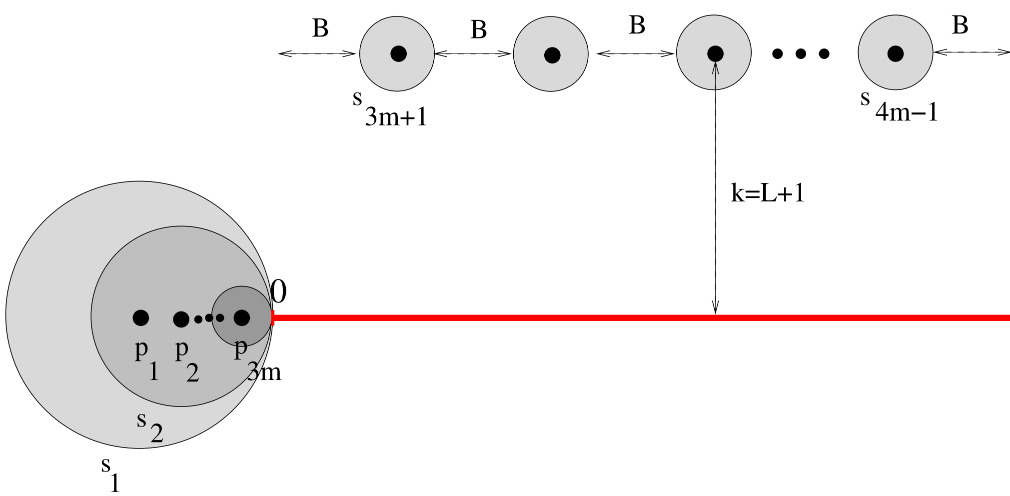

Figure 3: Reduction from 3-partition to the MinMax problem

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 $k+1$ (rectlinear 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 [6], we may assume that the values $a_1, a_2, \ldots, a_n$ are bounded by a polynomial cn^j for some constants c and j. Therefore, $B \leq 3cn^j$ and $k \leq cn^{j+1}$. Our reduction uses $n+m-1$ sensors and $n+m-1 \leq 2n$. The 3-partition problem can be represented using $O(n \log n)$ bits. 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.

Finally, by adding one additional sensor at distance $> k$ above the barrier, we can create an instance of the problem where $R > L$, and the proof remains exactly the same as that sensor cannot be involved in a covering assignment that has maximum relocation distance k. ■

2.2 Multiple Barriers

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 to an appropriate 2-barrier coverage problem. Given a partition instance with n items $a_1, a_2, \ldots, a_n$, we create n sensors such that sensor s_i has range $a_i/2$, and is initially placed at location $(i, 0)$. Create any two non-intersecting barriers b_1 and b_2, each of length $\Sigma_{i=1}^{n} a_i/2$. Clearly this is a polytime reduction, and there is a solution to the partition problem if and only if there is a solution to the barrier coverage problem. It follows that k-barrier coverage is also NP-hard.

3 Perpendicular Movement: One Barrier

In this section, we assume that sensors use perpendicular movement, and can only move to the closest point in the barrier. Without loss of generality, let the barrier b be the line segment between $(0, 0)$ and $(L, 0)$ and 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 $p_i = (x_i, y_i)$ and $x_1 - r_1 \leq x_2 - r_2 \leq \ldots \leq x_n - r_n$. Observe that sensors are ordered by the leftmost x-coordinate they can cover, and they can only move in a vertical direction. For simplicity we assume all points of interest (sensor locations, left and right endpoints of sensor ranges, barrier left and right endpoint) are distinct. 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* $[i, j)$.

3.1 Feasibility of coverage

We first show a necessary and sufficient condition on the sensors for the barrier to be covered. Since only vertical movements are allowed, given a point $p' = (x', 0)$ on the barrier, a sensor s in position $p = (x, y)$ with sensing range r can be assigned to cover p' only if $|x - x'| \leq r$. Once again, for technical reasons, we consider the interval on the barrier that a sensor can cover to be a half-open interval. More precisely, we say that the sensor s at position $p = (x, y)$ is a *candidate sensor* for $p' = (x', y')$ on the barrier if $x - r \leq x' < x + r$. Alternatively we say s potentially covers the point p'. Clearly, the barrier b can be covered only if every point on the barrier has a candidate sensor. Conversely, if every point has a candidate sensor, the problem can be solved in linear time by simply repeatedly covering the leftmost uncovered point on the barrier by moving the smallest numbered candidate sensor for the point down to the barrier.

3.2 MinSum

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 the first sensor is not 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 down 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:

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

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 $cost(S_i, [a, L))$ given the solutions to the sub-problems. Thus, by using either a tabular method or memoization, the problem can be solved in quadratic time.

Theorem 2 *Let $s_1, s_2, \dots s_n$ be n sensors initially located at positions $p_1, p_2, \dots, p_n$ 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.*

3.3 MinMax

The same dynamic programming formulation works for minimizing the maximum movement, except that in the case when the i-th sensor moves down in the optimal solution, the cost is the maximum of y_i and $cost(S_{i+1}, [x_i + r_i, L))$ instead of their sum. Thus, the MinMax problem can also be solved in $O(n^2)$ time. However, an alternative approach is more efficient. Consider the subset of sensors that are at distance at most d from the barrier. Clearly, we can check the feasibility of covering the barrier with such a subset in $O(n)$ time. The minimum value of d for which the problem remains feasible gives the solution to the minmax problem. This optimal value of d can be found using binary search on the set of distances of all sensors to the barrier. This gives the following result:

Theorem 3 *Let $s_1, s_2, \dots s_n$ be n sensors initially located at positions $p_1, p_2, \dots, p_n$ respectively, and let b be a barrier between between $(0, 0)$ and $(L, 0)$. The MinMax problem using only perpendicular movement can be solved in $O(n \log n)$ time.*

4 Perpendicular Movement: Multiple Parallel Barriers

In this section, we study the problem of covering multiple *parallel* barriers. We assume that sensors can relocate to any of the barriers, but will use perpendicular movement to move to the closest point of the chosen barrier. Without loss of generality, we assume all barriers are parallel to the x-axis. Since there are k barriers, there are 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 $x - r \leq x' < x + r$. Clearly, such a sensor is a candidate for an point p' on a border with x-coordinate x'. We say an interval $I = [a, b)$ of x-coordinates is *k-coverable* if every x-coordinate in the interval has k candidate sensors. Observe that such an interval of x-coordinates could exist on multiple barriers.

For simplicity, we explain the case of two barriers; the results for the feasibility and MinSum problems generalize to k barriers. 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 $(0,W)$ and (L,W) and the set of n sensors $s_1, s_2, \ldots s_n$ to be initially located at positions $p_1, p_2, \ldots, p_n$ respectively, and is ordered by the $x_i - r_i$ values as in Section 3. Thus, sensors may only move in a vertical direction. We assume that the ranges of sensors are smaller than 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.

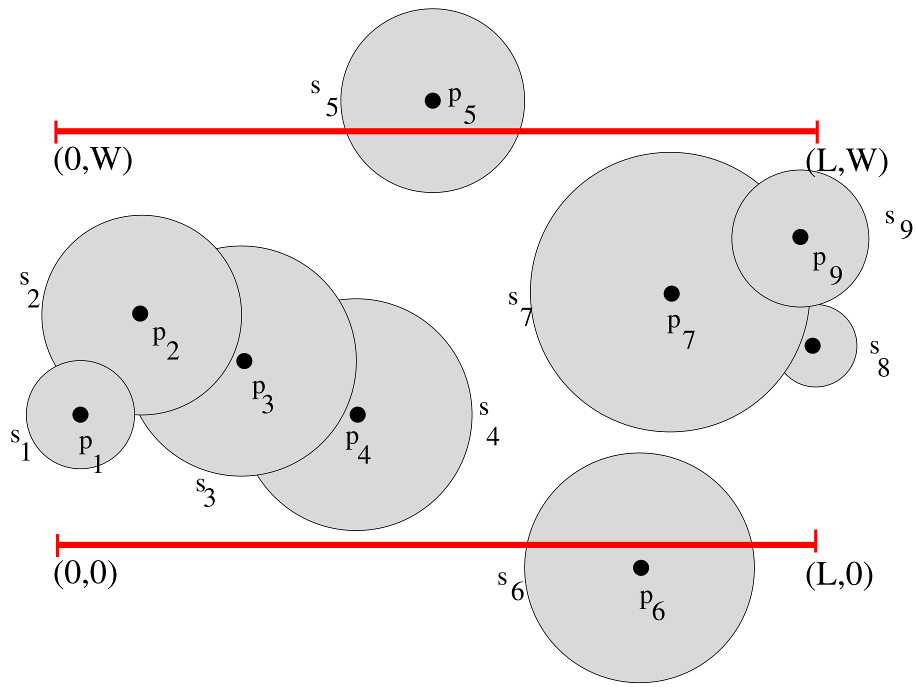

Figure 4: An example of a barrier coverage problem with two parallel barriers

4.1 Feasibility

We first show a necessary and sufficient condition on the sensors for the two barriers to be covered. Clearly, since the ranges of sensors are smaller than 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 $[0, L)$ is 2-coverable by S. We proceed to show that 2-coverability is also a sufficient condition, and give a $O(n)$ algorithm for finding a covering assignment for two parallel barriers. To simplify the proof of the main theorem, we first prove a lemma that considers a slightly more general version of the two parallel barrier problem.

Lemma 1 *Let $s_1, s_2, \ldots s_n$ be n sensors initially located at positions $p_1, p_2, \ldots, p_n$ respectively where $p_i = (x_i, y_i)$ and $x_1 - r_1 \leq x_2 - r_2 \leq \ldots x_n - r_n$. Let b_1 between $(0,0)$ and $(L,0)$ and b_2 between (P,W) and (Q,W), where $0 \leq P < L \leq Q$, be two parallel barriers to be covered. If intervals $[0,P)$ and $[L,Q)$ are 1-coverable, and interval $[P,L)$ 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. First we assign sensors to cover b_1 between $(0,0)$ and $(P,0)$ by repeatedly assigning an arbitrary candidate sensor to cover the leftmost uncovered point of this interval. Clearly this is possible, since the interval of x-coordinates $[0, P)$ is 1-coverable. Let s be the last 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 $x + r \geq P$.

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

$P < x + r < L$ 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. We now have a sub-problem of the same type as the original problem and proceed to solve it recursively.

$x + r = P$ Then there must be two other sensors that are candidates for the x-coordinate P. We arbitrarily pick one of these two candidate sensors and assign it to barrier b_1. It follows that the point (P, W) on barrier b_2 must be 1-covered, and in fact the initial interval of b_2 is 1-covered. Once again, the remaining sub-problem can be solved 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. ∎

The above lemma establishes that complete coverage of two parallel barriers b_1 between $(0, 0)$ and $(L, 0)$ and b_2 between $(0, W)$ and (L, W) can be achieved if and only if the interval of x-coordinates $[0, L]$ is 2-covered, and a covering assignment can be found in linear 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.

4.2 MinSum

The dynamic programming formulation given in Section 3.2 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 $xcost(S_i, [a_1, L), [a_2, L))$ 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:

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

It is not hard to see that the formulation can be generalized to k barriers; 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 to each of the barriers. 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 $O(kn^{k+1})$.

Theorem 4 *Let $s_1, s_2, \dots s_n$ be n sensors initially located at positions $p_1, p_2, \dots, p_n$ respectively where $p_i = (x_i, y_i)$ and $x_1 - r_1 \leq x_2 - r_2 \leq \dots x_n - r_n$. The MinSum problem for k parallel barriers using only perpendicular movement can be solved in $O(kn^{k+1})$ time.*

4.3 MinMax

Clearly a very similar formulation as above can be used to solve the MinMax problem in $O(kn^{k+1})$ time as well. However, the approach used in Section 3.3 can be used for multiple barriers, as shown in the theorem below:

Theorem 5 *Let $s_1, s_2, \dots s_n$ be n sensors initially located at positions $p_1, p_2, \dots, p_n$ respectively, and let b_1 between $(0, 0)$ and $(L, 0)$ and b_2 between $(0, W)$ and (L, W) be the two parallel barriers to be covered. The MinMax problem for the 2 parallel barriers using only perpendicular movement can be solved in $O(n \log n)$ time.*

Proof. We first show that given a maximum distance d, we can decide in linear time whether a covering assignment exists so that every sensor relocates at most distance d to its final position. If $d < W/2$, the sets of candidate sensors for each of the two barriers are disjoint. We can verify independently the feasibility of covering each barrier with its set of candidate sensors, as shown in Lemma 1.

If $d \geq W/2$, we partition S into the sets A, B, and C where A consists of sensors that are only candidates for barrier b_1 (that is, they are at distance $> d$ from barrier b_2), B consists of sensors that are only candidates for barrier b_2, and C consists of candidates for both barriers. We assign all sensors in set A to barrier B_1 and all sensors in set B to barrier B_2. This now leaves a set of uncovered intervals on each barrier. If there is a point x that is uncovered on either barrier and has no candidate sensors, then barrier coverage is impossible. If there is a point x that is only uncovered on one barrier and has a candidate sensor, then we assign the candidate sensor to the barrier. After this process is completed, we have a set of intervals that have non-empty parts that are 2-coverable. We now appeal to Lemma 1 to complete the proof.

The optimal value of d can be found using binary search on the set of distances of all sensors from each of the two barriers, and an application of Lemma 1 is done in $O(n)$ time. This completes the proof. ∎

5 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, but will use perpendicular movement to move to the closest point of the chosen barrier. In contrast to the case of parallel barriers, we show here that even the feasibility problem in this case is NP-complete. Figure 5 illustrates an example of such a problem. 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 [6]. 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 [7]. In order to obtain a reduction into a barrier coverage problem with two perpendicular barriers, we first put a monotone 3-SAT formula in a special form as shown in the lemma below:

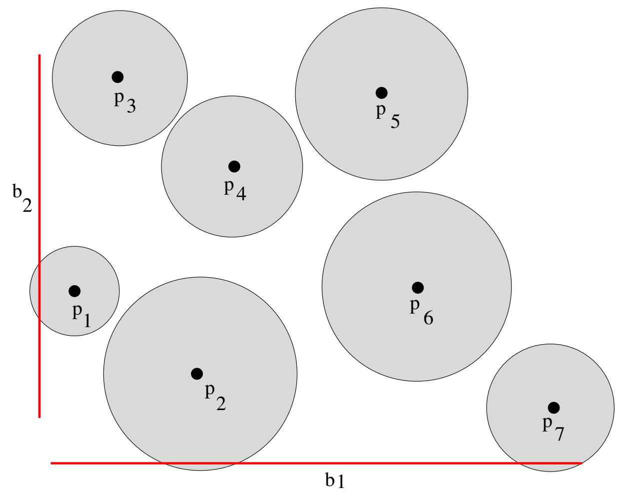

Figure 5: An example of a barrier coverage problem with two perpendicular barriers.

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 an equisatisfiable 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. *All clauses are of size two or three.*
2. *Clauses of size two contain exactly one variable from f and one new variable.*
3. *Clauses of size three contain only new variables.*
4. *Each new literal appears exactly once: either in a clause of size two or in a clause of size three.*
5. *Each x_i appears exactly in m clauses of f'_1, and exactly in m clauses of f'_2.*
6. *f' contains at most $4n + mn$ clauses.*
7. *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 x_p that appears in the clause numbered j in f_1, 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 $z_{p,1}, z_{p,2}, \ldots, z_{p,m-k}$. where x_p appears in clause i.

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 $(x_p \vee x_{p,i}), (x_p \vee x_{q,i}), (x_p \vee x_{r,i})$ 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 $(x_p \vee y_{p,1}) \wedge (x_p \vee y_{p,2}) \wedge \ldots (x_p \vee y_{p,m-k})$. Similarly, for every literal $\overline{x_p}$ that occurs $k < m$ times in f_2, we add clauses $(\overline{x_q} \vee \overline{z_{q,1}}) \wedge \ldots \wedge (\overline{x_q} \vee \overline{z_{q,m-k}})$. Finally, let $f' = f'_1 \wedge f'_2$. From the construction of f' it is easy to verify that it has Properties 1 to 6 mentioned in the statement of the lemma. Property 7 follows from Properties 1 to 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. ■

We give an example that illustrates the reduction and the ordering specified in Property 7:
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 shown bellow:

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

Theorem 6 *Let $s_1, s_2, \ldots s_n$ be n sensors initially located 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 NP-hard.*

Proof. Given an instance f of monotone 3-SAT we show how to obtain a corresponding instance P of the barrier coverage problem with two perpendicular barriers so that f is satisfiable iff there exist a covering assignment using perpendicular movement for P. 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 7. 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. Figure 6 illustrates the instance of barrier coverage corresponding to the 3-SAT formula from Example 5 above. 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 $(2p-1, 2q-1)$. For example 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 $(25, 7)$. 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 $(-1, 2\ell - 1)$.

Observe that in this assignment of positions to sensors, for any i, 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.

Assume f' is satisfiable, and consider a satisfying assignment for f'. We move every sensor s_i in P to the barrier b_1 if the corresponding variable is assigned $True$ and we move it to the barrier b_2 if the corresponding variable is assigned $False$. Since every variable in f' is set to either true or false, every sensor in P will either move horizontally or vertically. Since every clause in f'_1 is satisfied in the assignment, the corresponding sensor will cover the line segment from $(2i-2, 0)$ to $(2i, 0)$ of b_1. With a similar argument it can be shown that every segment from $(0, 2i-2)$ to $(0, 2i)$ on b_2 is also covered, and the assignment forms a covering assignment.

Now assume there is a covering assignment for P. For every variable in f' we set it to true if the corresponding sensor moves to the barrier b_1 in the covering assignment, and false otherwise. From the construction of the arrangement we know that if a sensor covers the line segment from $(2i-2, 0)$ to $(2i, 0)$ in the covering assignment then the corresponding variable in f' satisfies the clause i in f'_1. Since every such segment on b_1 is covered, all clauses in f_1 are satisfied. A similar argument shows that all clauses of f'_2 are satisfied. Therefore f' evaluates to true and is satisfiable.
■

It follows from the proof that the problem is NP-complete even when the sensors are in integer positions and the ranges are limited to two different sizes 1 and m. The proof works for $m \geq 4$ since it is easy to see that any instance of monotone 3-SAT problem can be transformed into one in which no variable occurs more than 4 times. 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 S a set of sensors, and 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

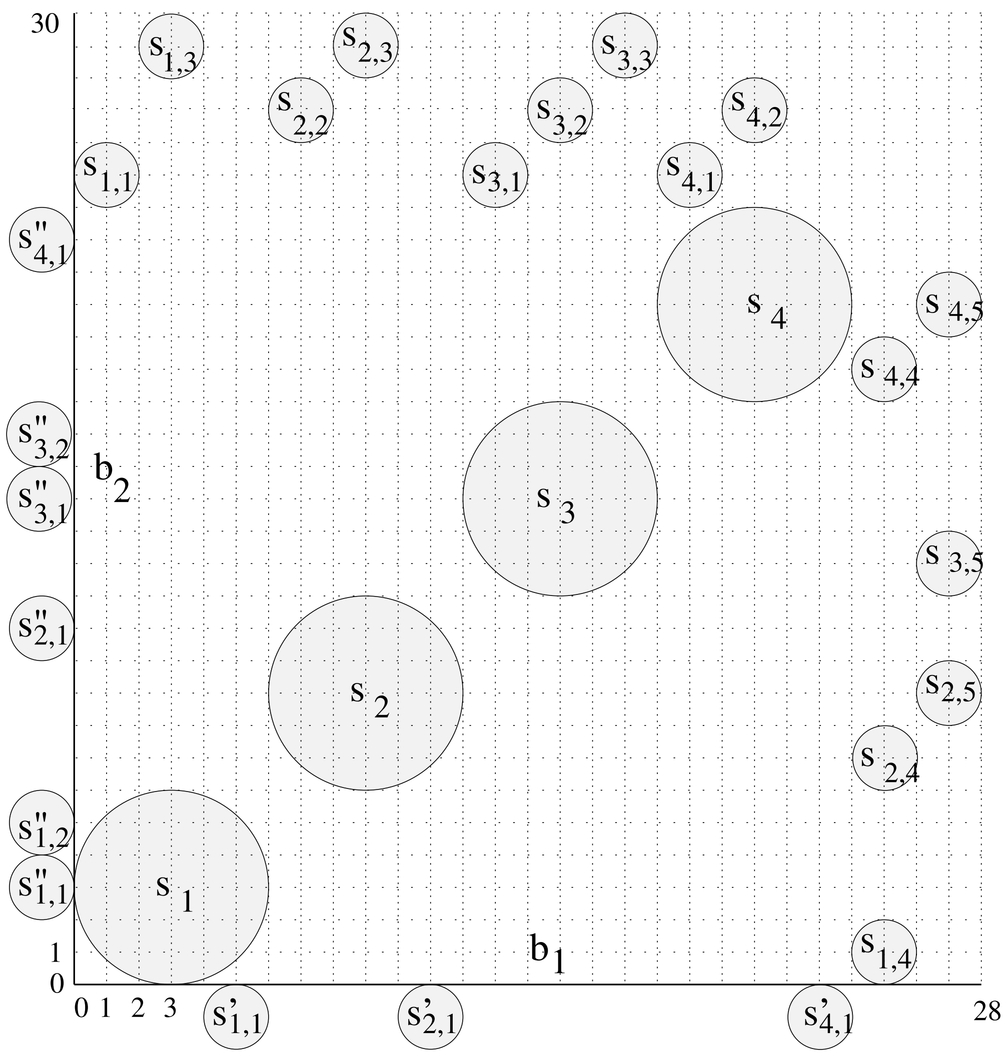

Figure 6: Barrier coverage instance corresponding to the monotone 3-SAT instance of Example 1

covered by s_1 and s_2 on the barrier b_1 (and b_2) are either the same or disjoint. This would be the case, for example, if all sensor ranges are of 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 7 *Let $S = \{s_1, s_2, \ldots, s_n\}$ be a set of sensors located in the plane in 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. 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. ■

6 Conclusions

It is known that the problem of minimizing the maximum movement to cover a line segment barrier when the sensors are initially located on the line containing the barrier is solvable in polynomial time [3]. In contrast, our results show that the MinMax barrier coverage problem becomes NP-hard 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 NP-hard, even when sensor ranges are restricted to two sizes. 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.

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 *MobiCom '07: Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, pages 75–86. ACM, 2007.

[2] 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. Combinatorial Optimization and Applications.

[3] 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*, 2012.

[4] 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.

[5] 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.

[6] 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.

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

[8] 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.

[9] S. Kumar, T. H. Lai, and A. Arora. Barrier coverage with wireless sensors. In *MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking*, pages 284–298. ACM, 2005.

[10] S. Kumar, T. H. Lai, and J. Balogh. On k-coverage in a mostly sleeping sensor network. In *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 144–158. ACM, 2004.

[11] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M.B. Srivastava. Coverage problems in wireless ad-hoc sensor networks. In *proceeddings of INFOCOM, the Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies, vol, 3*, pages 1380–1387, 2001.

[12] 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.

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