On the barrier-resilience of arrangements of ray-sensors

David Kirkpatrick*[1], Boting Yang†[2], and Sandra Zilles‡[2]

[1]Department of Computer Science, University of British Columbia, Vancouver, Canada.
[2]Department of Computer Science, University of Regina, Regina, Canada.

Abstract

Given an arrangement $\mathcal{A}$ of n sensors and two points s and t in the plane, the barrier resilience of $\mathcal{A}$ with respect to s and t is the minimum number of sensors whose removal permits a path from s to t such that the path does not intersect the coverage region of any sensor in $\mathcal{A}$. When the surveillance domain is the entire plane and sensor coverage regions are unit line segments, even with restricted orientations, the problem of determining the barrier resilience is known to be NP-hard [11, 12]. On the other hand, if sensor coverage regions are arbitrary lines, the problem has a trivial linear time solution. In this paper, we give an $O(n^2m)$ time algorithm for computing the barrier resilience when each sensor coverage region is an arbitrary ray, where m is the number of sensor intersections.

1 Introduction

Barrier coverage is an important coverage concept that arises in the analysis of wireless sensor networks [3]. Other notions of coverage try to quantify the effectiveness of a collection of sensors in detecting the presence of objects in a particular surveillance region. Barrier coverage, motivated by applications such as border control, measures the effectiveness of detecting the movement of objects *between*, but not necessarily *within*, critical regions. Given a sensor network, specified as an arrangement $\mathcal{A}$ of sensors with associated coverage regions in the plane, and two points s and t, we say that the sensor network provides *barrier coverage* if it guarantees that any object moving from point s to point t must be detected by at least one sensor.

If sensor regions are connected then determining barrier coverage amounts to asking if s and t belong to the same face of the arrangement, which is straightforward to check provided the sensor region boundaries are sufficiently simple. In order to measure robustness of barrier coverage, Kumar et al. [10] introduced *k-barrier coverage* that guarantees that any path from a point s to a point t must intersect at least k distinct sensor regions. They showed that for unit disk sensors (i.e., sensors whose coverage regions are unit disks) distributed in a strip separating s and t, k-barrier coverage can be determined efficiently by reduction to a maximum flow problem in the intersection graph of the disks.

Bereg and Kirkpatrick [2] introduced and studied the associated optimization problem, which they called *barrier resilience*. This specifies the minimum number of sensors whose removal permits a path from point s to point t such that the path does not penetrate any of the remaining sensor coverage regions. Bereg and Kirkpatrick showed that there is a 3-approximation (or, under mild restrictions concerning the separation of s and t, a 2-approximation) algorithm for computing the barrier resilience of unit disk sensors. When the sensor coverage regions are arbitrary line segments, Alt et al. [1] proved that the problem of determining barrier resilience is NP-hard and APX-hard. In fact, even if all sensor coverage regions are unit line segments in one of at most two orientations, the barrier resilience problem remains NP-hard [11, 12]. The reader is referred to the papers [3, 5, 8] for more information on barrier coverage and related problems.

It is straightforward to see that if sensor coverage regions are arbitrary lines, the barrier resilience problem has a linear time solution, since the resilience of a given arrangement of lines is just the number of lines that separate s and t.

*Email: kirk@cs.ubc.ca.
†Email: boting@cs.uregina.ca.
‡Email: zilles@cs.uregina.ca.
Research support for all three authors is provided by the Natural Sciences and Engineering Research Council of Canada.

In this paper, we address the case where sensor coverage regions are half-lines (or rays). We describe an $O(n^2 m)$ time algorithm for computing the resilience of an arbitrary arrangement of n rays with m intersections. (Due to space constraints, proofs are omitted; full proofs will appear in an expanded version of the paper.)

2 Ray barriers and barrier graphs

Let $\vec{V}$ be a set of n rays, specified by an endpoint and a direction, in the plane. Suppose that we are given a sensor network consisting of n sensors, where the coverage regions of sensors correspond to the rays in $\vec{V}$, and two points s and t which are not intersected by any of the rays in $\vec{V}$. We consider the problem of finding a subset $\vec{U}$ of rays in $\vec{V}$ with the minimum cardinality such that there is a path from s to t which does not intersect any rays in $\vec{V} \backslash \vec{U}$. The cardinality of $\vec{U}$ is referred to as the *resilience* of the sensor configuration $(s, t, \vec{V})$, and $\vec{U}$ is a *resilience set* of $(s, t, \vec{V})$.[1]

We say that a set $\vec{V'} \subseteq \vec{V}$ forms a *barrier* for $(s, t, \vec{V})$ if any path from s to t intersects at least one of the rays in $\vec{V'}$. Thus a set $\vec{U} \subseteq \vec{V}$ is a resilience set of $(s, t, \vec{V})$ if and only if $\vec{U}$ is a smallest subset of $\vec{V}$ with the property that $\vec{V} \backslash \vec{U}$ does not form a barrier. Our algorithm for computing a resilience set uses a reformulation of the problem as a graph problem. This reformulation is based on the observation that if a set of rays forms a barrier it must contain a subset consisting of two rays that forms a barrier; we refer to such a subset as a 2-*barrier*.

In order to substantiate this observation, we introduce some helpful notation. For two points a and b in the plane, we use ab to denote the line segment with endpoints a and b, and use $\vec{a}$ to denote a ray with endpoint a.

For the remainder of this paper, suppose, without loss of generality, that the line containing st is horizontal. (Accordingly, we will say "above (resp., below) st" as an abbreviation for "above (resp., below) the horizontal line supporting st".) For any ray $\vec{a} \in \vec{V}$, we assign a unique color as follows: (i) if $\vec{a}$ intersects st and goes down we assign it the color *red* (drawn as a solid ray in figures); (ii) if $\vec{a}$ intersects st and goes up we assign it the color *blue* (drawn as a dashed ray in figures); and (iii) if $\vec{a}$ does not intersect st we assign it the color *black* (drawn as a dotted ray in figures).

[1] Note that a configuration $(s, t, \vec{V})$ does not necessarily have a unique resilience set.

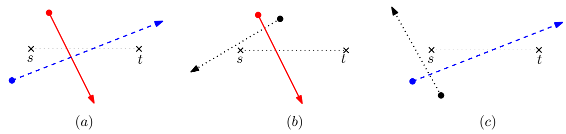

Figure 1: (a): a red-blue 2-barrier; (b): a red-black 2-barrier; (c): a blue-black 2-barrier

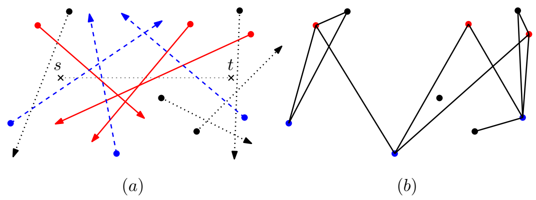

Figure 2: (a): a sensor configuration $(s, t, \vec{V})$; (b): its associated barrier graph

We first observe that certain pairs of intersecting rays are guaranteed to form a 2-barrier (see Figure 1).

Proposition 1 *Let $(s, t, \vec{V})$ be a sensor configuration. A pair of intersecting rays $\{\vec{a}, \vec{b}\} \subseteq \vec{V}$ forms a 2-barrier if (i) one is red and the other is blue, (ii) one is red and the other is black and they intersect above st, or (iii) one is blue and the other is black and they intersect below st.*

Lemma 2 *Let $(s, t, \vec{V})$ be a sensor configuration and $\vec{V'} \subseteq \vec{V}$. The set $\vec{V'}$ forms a barrier for $(s, t, \vec{V})$ if and only if there are two rays $\vec{a}, \vec{b} \in \vec{V'}$ such that $\{\vec{a}, \vec{b}\}$ forms a 2-barrier of one of the types described in Proposition 1.*

Lemma 2 motivates the reformulation of the resilience problem as a graph problem (see Lemma 3 below.) We say that a graph $G = (V, E)$ is a *barrier graph* of $(s, t, \vec{V})$, denoted by $G(\vec{V})$, if V is the set of all endpoints in $\vec{V}$, and $\{a, b\} \in E$ iff the corresponding pair of rays $\{\vec{a}, \vec{b}\}$ forms a 2-barrier (see Figure 2). It follows immediately from Lemma 2 that barrier graphs are tripartite.

Note that we use the same notation for a vertex in $G(\vec{V})$ and an endpoint in $\vec{V}$. When there is no ambiguity, a vertex of $G(\vec{V})$ is also referred to as an endpoint of a ray. We exploit the vertex-endpoint duality to view $G(\vec{V})$ as a vertex-coloured embedded graph. This allows us to say, for example, that any vertex that lies above st must be red or black, and any vertex that lies below st must be blue or black.

Lemma 3 *For any sensor configuration $(s, t, \vec{V})$, a vertex set V_c is a minimum size vertex cover of $G(\vec{V})$ if and only if the corresponding set of rays $\vec{V}_c$ is a resilience set of $(s, t, \vec{V})$.*

From Lemma 3, it is clear that we can find a resilience set efficiently if there is an efficient algorithm to compute a minimum size vertex cover of the graph $G(\vec{V})$. Unfortunately, the vertex cover problem on general tripartite graphs is NP-complete (there is a straightforward reduction from the vertex cover problem for cubic planar graphs, which is known to be NP-complete [6]). In fact, Clementi et al. [4] have shown that the minimum size vertex cover for tripartite graphs is not even approximable to within a factor of 34/33, unless P=NP.

Thus, we are motivated to look for additional structure in instances of the tripartite vertex cover problems that arise from barrier graphs. In some settings, for example if all rays are parallel to one of two different lines, the graph $G(\vec{V})$ is easily seen to be bipartite. It is well-known, by König's theorem [9], that, for bipartite graphs, constructing a minimum size vertex cover is equivalent to constructing a maximum size matching. Thus, we can use the Hopcroft-Karp algorithm [7] to find a minimum size vertex cover in such instances in $O(m\sqrt{n})$ time, where m is the number of edges in $G(\vec{V})$.

To exploit the structure inherent in less restricted instances, we first introduce some additional notation (see Figure 3). We denote by ℓ_{sv} the line passing through points s and v. Similarly ℓ_{tw} denotes the line passing through points t and w. A generic line through s (respectively, t) is denoted ℓ_{s-} (respectively, ℓ_{t-}). Similarly, a distinguished line through s (respectively, t) will be denoted ℓ_{s*} (respectively, ℓ_{t*}). With any line ℓ_{s-} (respectively, ℓ_{t-}) we associate the half-space, denoted ℓ^-_{s-} (respectively, ℓ^+_{t-}) consisting of all points to the left of ℓ_{s-}, or above ℓ_{s-} in case ℓ_{s-} is horizontal (respectively, all points to the right of ℓ_{t-}, or above ℓ_{t-} in case ℓ_{t-} is horizontal).

Armed with this notation, we can capture two additional properties of barrier graphs that can be exploited in the efficient construction of optimal vertex covers:

Lemma 4 *Let $G(\vec{V})$ be a barrier graph and suppose that V_c is any vertex cover of $G(\vec{V})$. Then there must exist lines ℓ_{s*} and ℓ_{t*}, through s and t respectively, such that V_c contains all of the red and blue vertices that lie in $\ell^-_{s*} \cup \ell^+_{t*}$.*

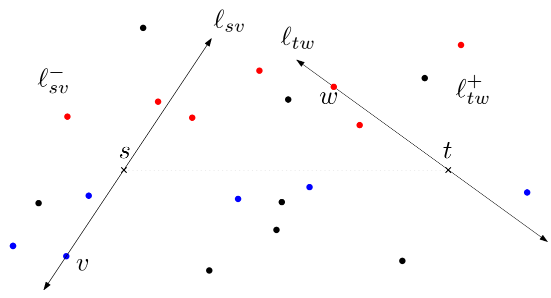

Figure 3: Lines (and associated half-spaces) through s and t.

Lemma 5 *Let ℓ_{s-} and ℓ_{t-} be arbitrary lines through s and t respectively, and let $\widetilde{RB}$ denote the set of red and blue vertices that lie in $\ell^-_{s-} \cup \ell^+_{t-}$. Then the subgraph of the barrier graph $G(\vec{V})$ that is induced by the vertices in $V \backslash \widetilde{RB}$, is bipartite.*

3 Algorithm

In this section, we present an algorithm for computing resilience sets. Its correctness follows immediately from Lemma 3.

Algorithm 1: RESILIENCE.

Input: Sensor configuration $(s, t, \vec{V})$.
Output: A resilience set for $(s, t, \vec{V})$.
build the vertex-coloured barrier graph $G(\vec{V})$ (described in Section 2)
return MIN-VERTEX-COVER($G(\vec{V})$)

As we have already noted, a minimum size vertex cover of a bipartite graph can be found in polynomial time [7]. So the basic idea of our algorithm is to exploit this by forming a sequence of subsets $U_1, U_2, \ldots$ of V such that (i) for all i, $G|(V \backslash U_i)$, the subgraph of $G(\vec{V})$ induced on the vertex set $V \backslash U_i$, is bipartite, and (ii) for some i, the minimum size vertex cover of $G|(V \backslash U_i)$, together with the vertices in U_i, forms a minimum size vertex cover of $G(\vec{V})$.

We know, by Lemma 4, that for any minimum size vertex cover V_c of $G(\vec{V})$ there must exist lines ℓ_{s*} and ℓ_{t*}, through s and t respectively, such that V_c contains all of the vertices in $\widetilde{RB}$, the set of red and blue vertices that lie in $\ell^-_{s*} \cup \ell^+_{t*}$. Furthermore, the vertices of $V_c \backslash \widetilde{RB}$ must be a minimum size vertex cover of $G|(V \backslash \widetilde{RB})$; otherwise

V_c would not have minimum size. So our algorithm simply tries all possibilities for ℓ_{s*} and ℓ_{t*}, determines the associated set $\widetilde{RB}$, finds a minimum size vertex cover of $G|(V\backslash\widetilde{RB})$ (which, by Lemma 5, is bipartite), and chooses, among all of these possibilities, one that minimizes the size of this vertex cover together with the set $\widetilde{RB}$.

Algorithm 2: MIN-VERTEX-COVER of a barrier graph.

Input: A barrier graph $G(\vec{V})$.
Output: A minimum vertex cover of $G(\vec{V})$.

$\widetilde{RB} \leftarrow$ {red vertices in V}
$VC_{temp} \leftarrow$ BIPARTITE-VERTEX-COVER$(G|(V\backslash\widetilde{RB}))$
$VC_{best} \leftarrow VC_{temp} \cup \widetilde{RB}$
for every red vertex v **do**
  for every red vertex w **do**
    $\widetilde{RB} \leftarrow$ {red and blue vertices in $\ell^-_{sv} \cup \ell^+_{tw}$}
    $VC_{temp} \leftarrow$ BIPARTITE-VERTEX-COVER$(G|(V\backslash\widetilde{RB}))$
    if $|VC_{temp}| + |\widetilde{RB}| < |VC_{best}|$ **then**
      $VC_{best} \leftarrow VC_{temp} \cup \widetilde{RB}$
return VC_{best}

As we have already noted, the correctness of Algorithm RESILIENCE follows immediately from Lemma 3. We now turn our attention to the correctness of our VERTEX COVER algorithm for barrier graphs.

Theorem 6 *The output of Algorithm 2 is a minimum size vertex cover of $G(\vec{V})$.*

It will be clear from the description of Algorithm 2 that the problem of constructing a minimum size vertex cover of a barrier graph with n vertices and m edges can be reduced to $O(n^2)$ minimum size vertex cover subproblems on induced subgraphs, each of which, by Lemma 5, is bipartite. As previously noted, König's theorem [9] states that, for bipartite graphs, constructing a minimum size vertex cover is equivalent to constructing a maximum size matching. Thus, we can use the Hopcroft-Karp algorithm to find a minimum size vertex cover in each subproblem in $O(m\sqrt{n})$ time [7], or $O(n^2m\sqrt{n})$ time in total. We note, however, that it is possible to implement Algorithm 2 to run in $O(n^2m)$ time, by ordering the successive subproblems in a way that they do not require independent solution; we leave the details to an expanded version of this paper.

References

[1] H. Alt, S. Cabello, P. Giannopoulos and C. Knauer, Minimum cell connection and separation in line segment arrangements, arXiv:1104.4618v2 [cs.CG], 2011.

[2] S. Bereg and D. Kirkpatrick, Approximating barrier resilience in wireless sensor networks, *Proceedings of the 5th International Workshop on Algorithmic Aspects of Wireless Sensor Networks, Lecture Notes in Computer Science*, Vol. 5804, pages 29–40, 2009.

[3] M. Cardei and J. Wu, Coverage in wireless sensor networks, In M. Ilyas and I. Mahgoub, editors, *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems*, chapter 19, pages 432–446, CRC Press, 2005.

[4] A. Clementi, P. Crescenzi, G. Rossi, On the complexity of approximating colored-graph problems. *Proceedings of the 5th International Conference on Computing and Combinatorics, Lecture Notes in Computer Science*, Vol. 1627, pages 281–290, 1999.

[5] A. Chen, T. H. Lai and D. Xuan, Measuring and guaranteeing quality of barrier-coverage in wireless sensor networks, *Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*, pages 421–430, 2008.

[6] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, San Francisco, 1979.

[7] J. E. Hopcroft and R. M. Karp, An $n^{2.5}$ algorithm for maximum matchings in bipartite graphs, *SIAM Journal on Computing*, 2(4): 225–231, 1973.

[8] S. Kloder and S. Hutchinson, Barrier coverage for variable bounded-range line-ofsight guards, *Proceedings of IEEE International Conference on Robotics and Automation (ICRA'07)*, pages 391–396, 2007.

[9] D. König, Gráfok és mátrixok, *Matematikai és Fizikai Lapok*, 38:116–119, 1931.

[10] S. Kumar, T. H. Lai, and A. Arora, Barrier coverage with wireless sensors, *Wireless Networks*, 13(6): 817–834, 2007.

[11] K.-C. Tseng, Resilience of Wireless Sensor Networks, Master thesis, University of British Columbia, BC, Canada, 2011.

[12] K.-C. Tseng and D. Kirkpatrick, On Barrier Resilience of Sensor Networks, *Proceedings of the 7th International Workshop on Algorithmic Aspects of Wireless Sensor Networks, Lecture Notes in Computer Science*, Vol. 7111, pages 130-144, 2011.