An Efficient Exact Algorithm for the Natural Wireless Localization Problem*

Bruno E. Crepaldi† Pedro J. de Rezende† Cid C. de Souza†

Abstract

Considered a variation of the art gallery problem, the wireless localization problem deals with the placement of the smallest number of broadcasting antennas required to satisfy some property within a given polygon. The case dealt with here consists of antennas that propagate a unique key within a certain antenna-specific angle of broadcast, so that the set of keys received at any given point is sufficient to determine whether that point is inside or outside the polygon. To ascertain this localization property, a Boolean formula must be produced along with the placement of the antennas.

In this paper, we propose an exact algorithm based on integer linear programming for solving the $\mathbb{NP}$-hard natural wireless localization problem. The efficiency of our algorithm is certified by experimental results which include the solution of instances of up to 600 vertices in less than five minutes on a standard desktop computer.

1 Introduction

The Art Gallery Problem (AGP) [9, 10, 8] is a longstanding research topic in Computational Geometry. New problems of this type arose upon the introduction of a novel concept of visibility in which guards are able to see through the gallery boundary [7]. The motivation for this formulation originated from applications to wireless networks, where signals from antennas are not blocked by walls.

To illustrate this situation consider the following folkloric example, which captures the essence of the problem [1]. The owner of a café would like to provide wireless internet access to her customers while preventing those outside her shop to access the network infrastructure. To accomplish this, antennas may be installed, each of which broadcasting a unique (secret) key within an arbitrary but fixed angular range. The goal is to place these antennas and to adjust their angles of broadcast so that customers within the area of the café could be distinguished from those outside simply by having them name the keys received at their location. In a more formal way, one seeks to characterize the polygon corresponding to the area of the shop by means of a monotone Boolean formula whose variables are the keys transmitted by the antennas. Since installation and maintenance of the antennas carry a cost, a natural optimization problem amounts to finding a solution with the minimum number of such devices.

Similarities between this problem and the traditional art gallery problem are self-evident, e.g., guards of the latter correspond to antennas in the former. Notwithstanding that the notions of visibility differ, henceforth we will use the term guard and antenna indistinctly.

As in the classical AGP, the *wireless localization problem* (WLP) has several variants depending on the choice of potential locations for guards, their angular range and maximum visibility distance. In this paper, we assume visibility to be unbounded.

Now, assume that the gallery floor plan is described by a simple polygon P. In the most general situation, guards may be placed anywhere inside P and can broadcast in any direction, in which case they are called *internal guards*.In a more restricted version, guard placement is limited to the vertices of P, and they are referred to as *vertex guards*. Moreover, another situation often found in the literature is the one known as *natural guarding*. Here, the guards are limited to lie on vertices or edges of P and to transmit their signals within the range corresponding to the interior angle of the polygon at that point.

The corresponding *Natural Wireless Localization Problem* (NWLP) is known to be $\mathbb{NP}$-hard [2].

In [1] an alternative $\mathbb{NP}$-hardness proof is given, which can be extended to more general types of guards, such as vertex and internal guards. There are also results [7, 6, 3] that lead to upper bounds on the number of guards sufficient for coverage, but these bounds are not always tight.

To the best of our knowledge, no exact algorithm has been proposed to this date to solve the NWLP. Furthermore, we are also unaware of any computational experiments reported in the literature for this problem.

Contribution This paper aims at filling these two gaps. To this end, in Sections 3 to 6, we model the NWLP problem as an integer program and in Section 7 we describe ingenious ways to use this formulation algorithmically. Computational results are presented in Section 8 validating this technique as a viable method for computing optimal solutions for instances comprised of hole-free polygons with up to 600 vertices. Conclusions

*This work was partially supported by grants from CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo) and FAEPEX/UNICAMP.

†Institute of Computing, University of Campinas, Campinas, Brazil, brunoecrepaldi@gmail.com,{rezende|cid}@ic.unicamp.br

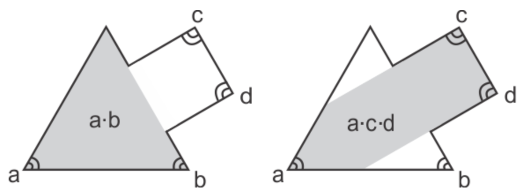

Figure 1: Polygon with guards on vertices a, b, c and d and Boolean formula $a \cdot b + a \cdot c \cdot d$.

and future directions follow.

2 Problem Definition and Terminology

A guard can be viewed as a wireless station positioned at a given location, which broadcasts a signal in a predefined *angle* and *direction*. The region, $Vis(g)$, covered by a guard g positioned at a point p is the cone with apex at p defined by two rays emanating from it. The bounding rays establish the *angle* and the *direction* of transmission of the corresponding guard in a natural way. Hence, from this point on, a *guard* will be identified to its cone of broadcast: the position of its apex and its angle of transmission.

We may now associate to a guard g a Boolean variable that, for every point p in the plane, takes a true value if and only if p belongs to $Vis(g)$. Given a polygon P and set of guards G, one may ask whether there exists a Boolean formula B on these variables that is satisfied uniquely on the points in P. In the affirmative case, G is said to form a *guarding* of P. Figure 1 illustrates this idea. For simplicity, in the remainder of the text, Boolean formulas are assumed to be in disjunctive normal form.

In the context of WLP, one is given a guard candidate set G known to contain a guarding of P. When a unitary cost is assigned to each guard in a guarding subset of G, the optimization problem seeks a guarding subset with minimum total cost. Variants of the problem depending on how the set of candidate guards G is defined can be formulated. Usually, G consist of a predefined finite set of locations, broadcasting angles and directions. Common locations for guards are the vertices and edges of P. In this work, we focus on the so-called *natural* guardings and on the resulting optimization problem, NWLP. A guard placed on a vertex of the polygon P is a *natural vertex guard* if its angle is the interior angle at that vertex, relative to P. A guard placed anywhere on an edge of P and broadcasting within an angle of π directed to the interior of P is called a *natural edge guard*. Since any two of these guards on a single edge would cover the same region, we can restrict the placement of natural edge guards to midpoints of edges. Accordingly, we will refer to a guarding consisting only of natural vertex and edge guards as a *natural guarding* [7].

3 Discretization

Viewing the NWLP as a continuous problem, for any point in the plane, the resulting Boolean formula should correctly identify whether it is inside or outside the polygon. In this section, we show that it actually suffices to ensure the validity of the formula for a finite set of points in the plane.

The rays on the boundary of the visibility regions of all natural guards define a planar arrangement. Notice that this arrangement coincides with the one obtained from the lines that support the edges of P. Moreover, since P has n edges, the planar subdivision induced by this arrangement has $O(n^2)$ faces. From here on, we use the term *face* to refer to a face of this subdivision. The next result shows that the correctness of a Boolean formula that solves the NWLP follows from its validity on any single point in each of these faces.

Lemma 1 *Given a simple polygon, let G be the set of its natural guards. Denote by $Sub(G)$ the planar subdivision induced by the visibility regions of all guards in G. A guard $g \in G$ covers one point in the interior of a face f of $Sub(G)$ if and only if g covers all points in f.*

Proof. Let $g \in G$ and let p be a point in the interior of a face f of $Sub(G)$ so that p is guarded by g. Suppose, by contradiction, that there exists a point q in f that is not guarded by g. Then, one of the rays that form the boundary of $Vis(g)$ must separate p from q. This contradicts the fact that f is a face of $Sub(G)$. The converse is immediate. □

Recall that a Boolean formula that solves NWLP must be satisfied at all points in the closure of P but not at the external ones. Lemma 1 establishes that it suffices to verify this property at a single point per face of the resulting subdivision and hence on $O(n^2)$ points.

4 An Integer Programming Model

We now turn our attention to the algorithm we propose for solving the NWLP to optimality. It is divided into two phases: a preprocessing phase, where the discretization described in Section 3 is computed and a solution phase, where we create and solve an Integer Linear Programming (ILP) model. In this section, we describe this model.

Consider an instance of the NWLP in which a polygon P is given. Recall that a solution consists of a Boolean formula, in disjunctive normal form, that discriminates the points in P from the points in the exterior of P. We say that a Boolean variable accepts (rejects) a point if it is true (false) for that point. Similarly, it accepts (rejects) a face if it accepts (rejects) all points of that face. Therefore, it suffices to create a clause that accepts the points (a single point actually will do) of *each* internal face while rejecting the points of *all* external faces.

Clearly, redundant clauses (covering the same internal faces) may be eliminated in a post-processing phase.

Let G be the set of all natural guards of P and F be the set of faces of the corresponding planar subdivision. Denote by $F_P \subset F$ ($F_{\overline{P}} \subset F$) the subset of faces internal (external) to P. Furthermore, we denote by $C_f \subset G$ the set of guards which cover face f, and by $N_{fh} \subset C_f$ the subset of its guards that, while covering face f, do not cover face h.

To each $g \in G$, we associate a binary variable x_g, which is 1 whenever guard g is used in the solution and 0 otherwise. Moreover, to each guard g and interior face $f \in F_P$, we relate a binary variable y_{gf}, which is 1 if and only if the Boolean variable corresponding to guard g is part of the clause built to ensure that face f is satisfied by the Boolean formula. We now formulate the Integer Linear Program:

$$\min \quad \sum_{g \in G} x_g,$$

$$\text{s.t.} \quad \sum_{g \in C_f} y_{gf} \geq 1, \forall f \in F_P, \tag{1}$$

$$\sum_{g \in N_{fh}} y_{gf} \geq 1, \forall f \in F_P, \forall h \in F_{\overline{P}}, \tag{2}$$

$$y_{gf} \leq x_g, \forall f \in F_P, \forall g \in C_f, \tag{3}$$

$$x_g \in \{0,1\}, y_{gf} \in \{0,1\}, \forall g \in G, \forall f \in F_P.$$

The objective function seeks to minimize the number of natural guards used. It is easy to see that the required Boolean formula may be built from the y_{gf} variables in the following fashion. A clause is associated to each $f \in F_P$ and the Boolean variable corresponding to a guard g will be part of this clause if, and only if, $y_{gf} = 1$. Constraints (1) guarantee that the internal faces are accepted, since at least one guard covers each face in F_P. Constraints (2) assure that exterior faces are not accepted by the formula, since for every pair of an internal face f and an external face h there is at least one guard accepting f and rejecting h. Constraints (3) prevent a clause from containing Boolean variables associated with a unused guard.

It is easy to modify this model so that the resulting Boolean formula is minimized along the process.

5 Strengthening the Model

Usual techniques to increase the computational efficiency of an ILP model amount to making it stronger in relation to dual bounds and more compact by reducing the number of constraints and variables in the formulation. In this section, we describe how these techniques can be applied to the model given in the previous section.

Firstly, notice that any single guard always covers external faces of the polygon, so, there is no point allowing for a clause consisting of a single Boolean variable.Therefore, we can tighten constraints (1) to require at least two guards to cover any given internal face. This already leads to a slightly more restricted linear relaxation. However, we can strengthen the model even further as a consequence of the following lemma:

Lemma 2 *For every edge e of a polygon P, any feasible solution of P includes a guard whose visibility cone contains e on its boundary.*

Proof. Since e is an edge of P, there is a pair of faces $f \in F_P$ and $h \in F_{\overline{P}}$ adjacent to e on the subdivision induced by the (natural) guard candidates. If p and q are interior points of f and h, respectively, any Boolean formula that accepts p and rejects q must contain a Boolean variable that corresponds to a guard g whose cone contains p and excludes q. This is only possible if $Vis(g)$ is bounded by a ray that contains e, otherwise, both f and h would not be faces. □

Let E denote the set of edges of P and G_e the set of natural guards g such that one of the rays that define $Vis(g)$ contains e. By Lemma 2, we can add the following constraints to the model:

$$\sum_{g \in G_e} x_g \geq 1, \forall e \in E \tag{4}$$

6 Shadow and Light Faces

Solving the ILP model proposed in Section 4 using all faces can be very costly. However, we can significantly reduce the number of faces considered in constraint (2) and still guarantee that the algorithm finds a valid formula using the minimum number of guards. To accomplish this, we can extend to the NWLP the notion of *shadow* and *light* faces, presented in [4].

Firstly, define a partial order $\prec$ both on F_P and on $F_{\overline{P}}$ as follows. If $f, f' \in F_P$ ($\in F_{\overline{P}}$) then $f \prec f'$ if and only if $C_{f'} \subset C_f$. We call $f \in F_P$ ($\in F_{\overline{P}}$) an internal *shadow face* (external *light face*) if f is minimal (maximal) with respect to $\prec$.

Lemma 3 *If a Boolean formula accepts all internal shadow faces, then it accepts all internal faces.*

Proof. Let B be a Boolean formula that accepts all internal shadow faces. Let f be any internal face. If f is a shadow face, we are done. Suppose f is not a shadow face. Then, there must exist an internal shadow face f' such that $C_{f'} \subset C_f$. Since B accepts f', there is at least one clause of B whose Boolean variables represent guards that cover f'. Since $C_{f'} \subset C_f$, this clause also accepts f. □

Lemma 4 *If a Boolean formula rejects all external light faces, then it rejects all external faces.*

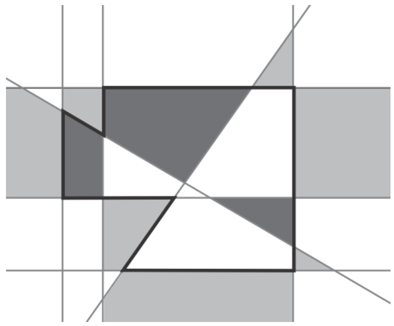

Figure 2: Example of internal shadow and external light faces of a polygon.

Proof. Analogous to the previous proof. □

From Lemmas 3 and 4, the following theorem follows.

Theorem 5 *A Boolean formula is a solution to an instance of the NWLP if and only if it accepts all internal shadow faces and rejects all external light faces.*

Let S_P be the set of internal shadow faces and $L_{\overline{P}}$ be the set of external light faces. Theorem 5 implies that we can replace the sets F_P and $F_{\overline{P}}$ by the sets S_P and $L_{\overline{P}}$, respectively, hence reducing the size of the ILP model. Based on our experimental results, this reduction has a significant impact on the efficiency of our algorithm.

7 An Efficient Iterative Algorithm

In this section, we further enhance the model proposed in Section 4, by incorporating the strengthening and compression refinements proposed in Sections 5 and 6. Furthermore, we propose a more effective way for solving the model in order to arrive at a more efficient algorithm able to quickly handle instances of considerable size.

From constraints (3), the Boolean variable associated to a used guard might not be present in the clause liable for accepting a face covered by that guard. However, in order to make the model more compact, we may tighten the constraints (3) to $y_{gf} = x_g$, effectively requiring the clause responsible for accepting face f to contain *all* variables associated to used guards that cover f. Therefore, we can remove all variables y_{gf}, obtaining following ILP model:

$$\min \quad \sum_{g \in G} x_g,$$

$$\text{s.t.} \quad \sum_{g \in C_f} x_g \geq 2, \forall f \in L_P, \tag{5}$$

$$\sum_{g \in N_{fh}} x_g \geq 1, \forall f \in L_P, \forall h \in S_{\overline{P}}, \tag{6}$$

$$\sum_{g \in G_e} x_g \geq 1, \forall e \in E \tag{7}$$

$$x_g \in \{0, 1\}, \forall g \in G.$$

This model finds a solution that minimizes the number of guards, but the resulting formula may be much larger than necessary, since the clause responsible for accepting a face f will have all variables that represent used guards that cover f. However, this model can be solved much more efficiently than the initial model and, for now, we are not particularly concerned with the length of the Boolean formula.

Let us look into the growth of the number of constraints (6) compared to the increase in the size of the instances (i.e., the number of edges of the input polygons). While the model contains only n constraints (7), the number of constraints (5) is $O(n^2)$ – proportional to the number of internal shadow faces. However, there is one constraint (6) for each pair of internal shadow and external light faces, leading to $O(n^4)$ of these constraints. Hence, if we found constraints (6) that we could avert checking, we might end up with a much smaller and more efficient model.

We observed, experimentally, that if a small set of constraints (6) are satisfied by the guards and clauses used, many other constraints (6) are automatically satisfied as well. Building upon this observation, we devised the following iterative algorithm.

Preprocessing phase. Two procedures are executed: the first one computes the visibility regions of the guards (cones) while the second one creates the planar subdivision and identifies the light and shadow faces.

Solution phase. The model is built without the constraints (6). Iteratively, the restricted model is solved to optimality and any violated constraints (6) are added to the model prior to the next iteration, until a viable (and optimal) solution is found.

8 Computational Experiments

In this section, we discuss the experimental investigation we carried out to evaluate the algorithm proposed in Section 7.

Our programs were coded in `C++`, compiled with `GNU g++ 4.6`, and made use of `CGAL 4.1` (Computational Geometry Algorithms Library). The solver used to compute the ILP models was `IBM ILOG CPLEX 12.2`. As for

Table 1: Average number of faces.

Vertices	Internal Faces	Shadow Int Faces	External Faces	Light Ext Faces
20	49	12	161	22
40	211	38	610	69
60	493	73	1338	134
80	852	116	2389	233
100	1331	175	3720	355
200	5541	604	14560	1268
300	12566	1278	32585	2762
400	22549	2191	57652	4829
500	35124	3336	90127	7386
600	51968	4815	128333	10477

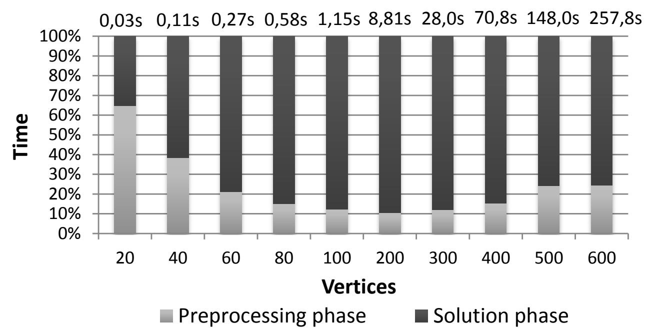

Figure 3: Average computation times, total and per phase.

hardware, a desktop PC featuring an AMD Phenom II X6 1055T @ 2.80GHz and 8GB RAM was employed.

The instances tested correspond to simple polygons randomly generated by a procedure present in CGAL. This procedure starts off by randomly distributing the vertices of the polygon uniformly on a given rectangle and then applies the method of elimination of self-intersections using 2-opt moves. The instances that comprise our benchmark may be downloaded from *www.ic.unicamp.br/~cid/Problem-instances/Wireless-Localization*.

The number of vertices of the polygons associated to these instances was chosen in the ranges: $[20, 100]$ with step size 20 and $(100, 600]$ with step size of 100. For each polygon size, 30 instances were created.

The first aspect to be considered in our analysis relates to the reduction on the size of the original ILP model described in Section 4 as a consequence of the application of Theorem 5. Recall that the number of faces on the planar subdivisions is the main determining factor of the number of constraints in the model. Table 8 show the average number of *internal*, *external*, *shadow internal* and *light external* faces per polygon size. Using only the internal shadow and external light faces, we reduced the number of internal and external faces to be considered on average by $86.7\% \pm 4.7\%$ and $90.4\% \pm 1.7\%$, respectively. Taking into account that in the original model there is one constraint of type (2) for each pair of internal and external faces, when we limited these pairs to the internal shadow and external light faces, the number of constraints in the ILP formulation dropped by $98.6\% \pm 0.8\%$ on average. This huge decrease in the model size evoked by the results presented in Section 6 was one of the key ideas that made solutions of instances of hundreds of vertices possible.

As our algorithm has two phases, the next analysis focus on how the computation time breaks up between them. Figure 3 summarizes the average percentage of the time spent by the iterative algorithm in the preprocessing and solution phases. The average total time to solve an instance is displayed over each bar. Notice that solutions we found for all instances in less than five minutes and the preprocessing phase took on average $23.7\% \pm 16.6\%$ of that time. These results show that our approach is a very good choice for calculating optimal solutions for the NWLP on polygons of several hundreds of vertices as they can be obtained in only a few minutes. The fact that preprocessing requires about one-third of the time spent by the solution phase may seem surprising at first. After all, the former is a polynomial time procedure while the latter involves multiple solutions of an NP-hard problem. However, as observed earlier in experiments on the classical AGP (see [5]), the current technology of ILP solvers is extremely advanced and allows for handling difficult problems very efficiently in practice.

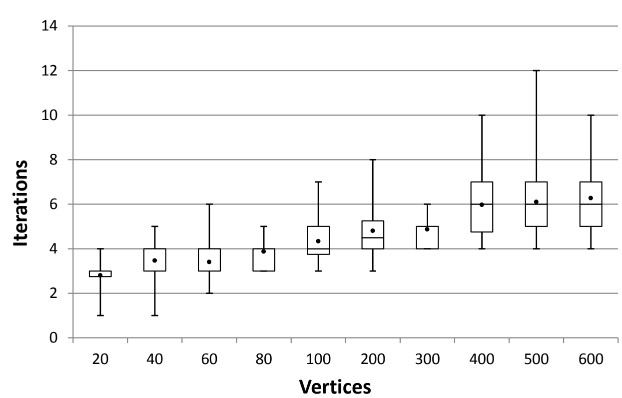

Figure 4: Number of iterations by polygon size.

An important point on the analysis of our algorithm relates to how the number of iterations increases with the size of the instances. This can be assessed by analyzing the data displayed in Figure 4. We see that, on average, 4.5 ± 1.2 iterations were sufficient to reach the optimum and that no instance in our benchmark required more than 12 iterations. Preliminary tests, where the *entire* initial ILP model was given as input to the solver, failed to attain optimal solutions on polygons of 50+ vertices within acceptable times. On the other hand, the data in Figure 4 show that the number of iterations until convergence is reached is small. Bringing to mind that each iteration requires the solution of a much lighter ILP, we conclude that the iterative computation is indeed crucial in achieving the small computation times shown in Figure 3.

To perceive how much smaller the ILP models solved at each iteration are compared to the full model given in Section 7, we measured the number of constraints (6) added along the iterations and compared it to the total of constraints of this type. On average, in the last iteration of the algorithm, the model has only $0.6\% \pm 1.1\%$ of all constraints (6).

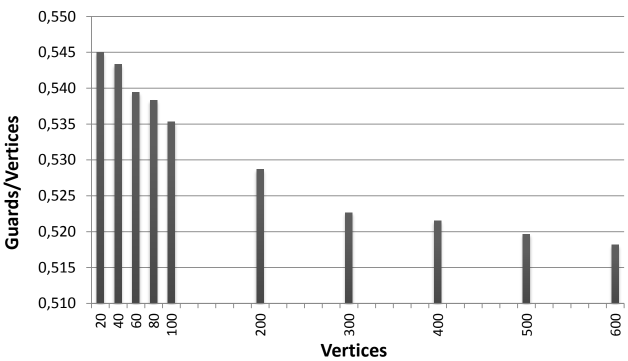

Figure 5: Ratio of guards to vertices by polygon size.

Lastly, an interesting insight on the guards per vertices ratio. It is known that, for a polygon of n edges, $n/2$ is a lower bound for the number of guards on an optimal solution. Figure 5 shows that for the random polygons in our benchmark, the number of guards used in the optimal solutions approaches $n/2$ as n increases.

9 Comments and Future Directions

To the best of our knowledge, this investigation on practical solutions to the natural wireless localization problem is unprecedented. Besides being known as an $\mathbb{NP}$-hard problem [2] only a few theoretical studies on the NWLP have been undertaken [7, 6, 3].

The algorithm we proposed in this paper is based on an integer linear programming model and derives its effectiveness from an elaborate reduction on the number of constraints. An iterative approach has lead to significant gains in efficiency, which yielded solutions to instances of up to 600 vertices in less than five minutes of computation.

Extensions to this approach that might solve instances where the polygons contain holes or when antennas are not restricted to polygon vertices are worth investigating.

Heuristics or alternative ILP models may be compared to the results we described here by accessing our set of benchmark instances made public together with the optimal solutions we found.

Lastly, we believe that the knowledge of exact solutions to a large collection of instances may lead to new theoretical developments on the NWLP, hence improving the understanding of the problem.

References

[1] Tobias Christ and Michael Hoffmann. Wireless localization with vertex guards is $\mathbb{NP}$-hard. In *Canadian Conference on Computational Geometry (CCCG)*, pages 149–152, 2009.

[2] Tobias Christ, Michael Hoffmann, and Yoshio Okamoto. Natural wireless localization is $\mathbb{NP}$-hard. In *Proceedings of the 25th European Workshop Comput. Geom.*, pages 175–178, 2009.

[3] Tobias Christ, Michael Hoffmann, Yoshio Okamoto, and Takeaki Uno. Improved bounds for wireless localization. In *Proceedings of the 11th Scandinavian Workshop on Algorithm Theory*, pages 77–89, Berlin, 2008. Springer-Verlag.

[4] Marcelo C. Couto, Pedro J. de Rezende, and Cid C. de Souza. Experimental evaluation of an exact algorithm for the orthogonal art gallery problem. In *7th International Workshop on Experimental Algorithms*, volume 5038 of *Lecture Notes in Computer Science*, pages 101–113, 2008.

[5] Marcelo C. Couto, Cid C. de Souza, and Pedro J. de Rezende. An exact algorithm for minimizing vertex guards on art galleries. *International Transactions in Operational Research*, 18:425–448, 2011.

[6] Mirela Damian, Robin Y. Flatland, Joseph O'Rourke, and Suneeta Ramaswami. A new lower bound on guard placement for wireless localization. *CoRR*, abs/0709.3554, 2007.

[7] David Eppstein, Michael T. Goodrich, and Nodari Sitchinava. Guard placement for efficient point-in-polygon proofs. In *Symposium on Computational Geometry*, pages 27–36, 2007.

[8] Alexander Kröller, Tobias Baumgartner, Sándor P. Fekete, and Christiane Schmidt. Exact solutions and bounds for general art gallery problems. *ACM Journal of Experimental Algorithmics*, 17(1), 2012.

[9] Joseph O'Rourke. *Art gallery theorems and algorithms.* Oxford University Press, Inc., New York, NY, USA, 1987.

[10] Jorge Urrutia. Art gallery and illumination problems. In *Handbook of Computational Geometry*, pages 973–1027. North-Holland, 2000.